A Fog Based Security Model For Electronic Medical Records In the Cloud Database

Elizabeth M J, Jobin Jose, Dona Jose

Abstract*: *Nowadays, a lot of emerging trends such as telemedicine, robotics in hospitals, computerized medical diagnosis, cybersecurity, Artificial intelligence, etc. are used to uplift the healthcare sector. Especially telemedicine services get more attention to diagnose and treat patients where health professionals use information and communication technologies for remote patient monitoring. The medical data such as CT scan, MRI reports, X-rays, Heart or Kidney transplantation videos and other health information should be available in digital format and such type of huge multimedia big data needs to be kept in the cloud and needs to be archived. Cloud storage provides better storage capability so that customers no need to worry about their limited resources. This paper proposes a new method for securing various electronic medical records in the cloud database. This proposal mainly concentrates on how we can get data with less latency for patient monitoring and how to secure the patient's private data to overcome data breaches in the cloud server using fog computing technology. A pairing based cryptography technique such as an Elliptic Curve Diffie-Hellman key agreement protocol and a decoy technique are used to access and store data more securely along with the help of some cryptographic algorithms.

Index Terms*: *Telemedicine, Fog Computing, Healthcare, Cybersecurity, Electronic Medical Records.

I. INTRODUCTION

Highlight Over the past fifteen years, the main focus was on the use of cloud computing service but the steep increase in the number of devices connected over the network will not provide better performance through the centralized server. Therefore, the new computing paradigm emerged is used to provide storage and computation facilities towards the edge of the network closer to end users [1]. The word fog computing (From cOre to edGe) is formulated by Cisco in 2012 and IBM termed it as edge computing [2]. However, these words mainly focus on how to decentralize the computing infrastructure and how to provide all the computing resources in between the data source and the cloud server where the internet ends.

Now we are at the beginning of a very transformative era of fog computing. Fogonomics is an emerging business model of fog computing called fog-as-a-service. The pros and cons of fogonomics are still un-studied but to develop business applications on the edge of the network, a platform is needed. In the field of IT, the tangible and intangible factors of a company always depend on data, is a financial and strategic asset of an organization. Therefore the cloud providers must take the responsibility to protect the data at rest and during transmission. In order to gain long-term value, entrepreneurs regularly map their cloud resource requirements closely to their key business goals [4]. The latest cyber-attacks, for example, WannaCry attack and Petya ransomware attack, etc., show that the existing cloud paradigm lacks enough security mechanisms. Therefore, Fog computing with different security mechanisms gets more attention in this era of computing. The market of fog computing is driven by the increasing IOT applications, machine to machine communication, the need for real-time processing, and the increasing demand for a large number of connected devices, etc. According to the MnM report, by 2022, the expected growth-rate of fog computing market is from USD 22.28 Million in 2017 to USD 203.48 Million[15]. Here, we need to pay for resources based on usage that is paying for resources according to use over time.

A lot of emerging trends used to uplift the healthcare sector. Telemedicine services get more attention to diagnose and treat patients where health professionals use ICT for remote patient monitoring. In telemedicine service, the emerging fog computing paradigm plays a major role in providing latency sensitive information as well as data privacy and security. Medical big data refers to a set of electronic health records, includes clinical data, sensor data, insurance, pharmacy, laboratory data, medical images, and other multimedia medical data. Nowadays, these are available in digital format; such records are called electronic medical records (EMR). These EMR records need to be kept in the secure storage and need to be downloaded for the emergency case with very less latency and need to be archived in the healthcare cloud.

A real-time two-way communication takes place between patient and healthcare provider through audiovisual media and integrated medical devices are used for the purpose of consulting. Telemedicine service providers are responsible for the transmission of EMR over the internet to provide better treatment and care of remote patients. Here, a lot of security issues may arise in the health cloud similar to cloud computing such as privacy and protection of data, legal and policy issues, lack of transparency and cybersecurity issues.

The rest of the paper is organized as follows: The background and motivation are explained in section II. The work done so far in this area is presented in section III. Section IV shows the details of the model. The implementation details are explained in part IV and the performance analysis is done in section V. The last section concludes the paper.

Revised Manuscript Received on May 06, 2019

Elizabeth M J, Computer Science and Engineering, Viswajyothi College of Eng. and Tech., Vazhakulam, Kerala , India.

Jobin Jose, Computer Science and Engineering, Viswajyothi College of Eng. and Tech., Vazhakulam, Kerala , India.

Dona Jose, Computer Science and Engineering, Viswajyothi College of Eng. and Tech., Vazhakulam, Kerala , India.

II. BACKGROUND AND MOTIVATION

A. Cloud Computing

In cloud computing, we can define three types of cloud service models: Iaas, Paas, and Saas. Iaas which provides customers get access to the service provider's infrastructure but they use their own applications and platforms. Paas offering provides a platform or cloud environment for developing their applications. Saas delivers software applications, customers no need to install software in their local system, pay for using the software without owning the underlying infrastructure [3].

B. Fog Computing

Fog computing is an extension of cloud computing and it is superior to edge computing. Fog computing paradigm inserts a new layer called a layer of fog nodes in the cloud paradigm. The main fog specific features are low latency, optimal resource allocation, fast data access, and interoperability, etc. can be achieved through the scalable edge microdata centers placed in between the centralized cloud server and the end users. The fog computing architecture is shown in Fig: 1.

The architecture contains three tiers: In tier 1, a user to fog server communication takes place through edge gateways. In tier 2, we have to consider the communication within the fog commuting devices. Every fog devices must support intra fog-tier communication because it must support mobility of the terminal nodes. The interaction between tier 1 and tier 2 is through edge gateways. In tier 3, the communication between the fog devices and the cloud servers takes place through fog and cloud gateways.

Telemedicine is used in different medical fields such as tele-stroke services, tele-rehabilitation services, behavioral health, telemedicine in dentistry, dermatology, cardiology, as a way to provide better healthcare and services. The existing system provides security by encryption but it fails to secure the cloud. The existing cybersecurity tools are not enough to protect an entrepreneur's assets such as electronic medical records when the telehealth center connects with remote doctors or hospitals and patients together. The existing single authentication method is not enough to prevent fraudulent activities in the cloud. So how can we protect telehealth center website details or EMR details safely in this current era? is a challenging question which is motivated me to think about it and to do my project in this context. This is an area where more security challenges are hidden. Therefore we have to take into account all the ways the attackers used so far for hacking systems. Here, hardware and software security play a major role during EMR transmission between both doctors/hospitals and patients.

The main objective of this paper is to present a better storage capability for healthcare cloud, efficient data access and mobility of data, the technology, and mechanisms used for providing enough security for telemedicine data, images or videos, etc. in the field of healthcare cloud using fog computing technology. This paper focuses mainly on how to mitigate the healthcare cloud storage security gap. In order to solve security problems and privacy issues, H. A. Al Hamid et. al [6] proposed a pairing-based encryption method along with decoy technique to resist insider attack as well as outsider attack, which motivated me to work in this area.

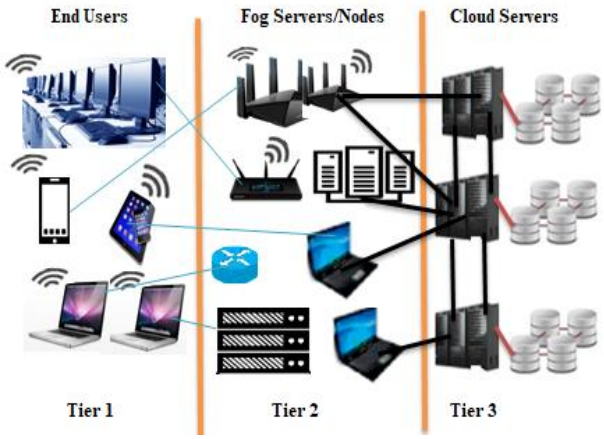

Figure 1: Architecture of Fog Technology

Another interesting step what I have taken here is to hide medical videos such as heart transplantation surgery video, Kidney transplantation videos or open surgeries of the heart, brain videos, etc. within their own image in an encrypted form. Therefore, it is possible to provide high security for their private information which can be in any format (text, images or videos) in the cloud server. If any masquerade activity takes place then the attacker can get decoy images only.

III. RELATED WORK

The authors, Wang et. al [7] explained a three-layer storage framework for fog computing to protect the privacy of data and they designed a Hash-Solomon code algorithm, is an enhanced version of Reed-Solomon encoding algorithm. They breakdown the file into three parts with encoding technology and then they placed these pieces of information, separately, into the different layers of the local machine, fog nodes, and cloud servers. Their work helps to secure data but it does not ensure data availability within the time period because to access a file we need to access all the nodes simultaneously, in which the pieces of information stored. In this three-layer architecture, the uppermost layer consists of cloud data centers. The next fog computing layer consists of different fog nodes with some kind of processing. They used the advantages of CI to do complex calculations of Hash-Solomon code algorithm in this fog layer. The lowermost layer is of a wireless sensor network. The main advantage is to protect the privacy of data by placing user's data in different layers so that if an incident occurs, an attacker cannot do anything with these piece of information and this method is not suitable for latency-sensitive applications because it takes time to access a file and Hash-Solomon algorithm needs complex calculations.

The two additional cloud-related insider risks are presented in the paper [8] by W. R. Claycomb et.al. These risks are related to insiders: the person who steal information from a data-center, and the person who makes an attack on the local resources of the employer. A detailed description is given about the hierarchical structure of the administrators and the threats of rogue administrators within the service providers. They also explained the future research on cloud-related vulnerabilities and what are the matters they

should focus on to identify and to address unique vulnerabilities, but not mentioned how to resist the insider or outsider attack.

In the Paper [9] Subhadeep Sarkar et.al, presents fog computing architecture and networking model is illustrated in the context of IoT by Sarkar et.al. Their work mainly focused to formulate the model of fog computing paradigm mathematically and based on their case study, they point out that fog computing paradigm outperforms traditional cloud computing paradigm when real-time applications increase. To assess the performance of the model they proposed different types of performance matrices such as transmission latency, processing latency, CO2 emission, power consumption, and operational cost. But they did not mention anything about the protection of data privacy and security. The detailed description of the fog architecture and its network model and its practicality also explained in the paper. They formulated all the performance metrics mathematically. But not mentioned anything about the protection of data privacy and security.

A novel approach is proposed by S. J. Stolfo et.al [10] to secure cloud data by profiling user behavior with decoy technique. The decoy files header contain keyed-HMAC (Hash Message Authentication Code), is calculated based on the file contents. These decoy files help us to detect masquerade activity, to confuse the attacker and prevent the attacker to do so by its deterrence effect. Here they mentioned how to handle data theft attack in the cloud or fog. Fog computing faces a lot of network security issues such as malicious re-directing and driver attacks, SQL injection attack, session and cookie hijacking, other web and cyber-attacks. So the proposed method in this paper is not enough to resist all the insider and outsider attack. It detects abnormal data access patterns by combining user profiling with decoy technique. They didn't mention the matrices needed to measure the performance.

A software-based system using a critical defense method called decoy technique to prevent data theft attack inside the cloud data is proposed by Park et.al [11]. They used a code obfuscation technique to generate decoy codes for the source program to avoid data ex-filtration. Bogus programs are syntactically different but semantically identical versions. The proposed system in this paper synthesizes, automatically, a series of bogus software programs using different code transformations, traps and beacons to deceive and confuse insiders. The provided information is not enough to provide complete security mechanisms in fog computing as well as in the cloud computing paradigm because an attack can occur anywhere during data transmission.

H. K. Patil et.al reviewed few healthcare security issues and explained the needs for the improvements especially technological aspects in the field of securing healthcare data [12]. The steep increase in the security breaches signalizes that traditional security solutions are not enough to provide security and we cannot apply such solutions to the big data-sets which are more diverse. They proposed a real-time healthcare model to show the importance of real-time processing on various parameters and values from different fields affecting the patient's health status. For example, a patient's vital signs (pulse rate, temperature, etc.) can be normal, but their psychological or behavioral factors can have awful problems. Moreover, they indicate that safety does not guarantee through HIPAA certification. Therefore it is necessary to give more importance to keep electronic medical records safely. The main drawback is that they do not propose any model for a security issue in healthcare IT.

A basic steganography algorithm is used to modify the least significant bit (LSB) of each pixel from an image to hide one character of information from the plain text. Therefore in 2016, the author Shreyank N Gowda proposed an algorithm [13] to hide more characters, is stronger than the LSB algorithm. It can be broken down easily. The purpose of this proposed famous cryptography algorithm is to provide enough security for the information being hidden. This proposed approach is stronger than the LSB algorithm and provides more security.

The elliptic curve crypto-system is faster than RSA crypto-system. For example, the key length of 1024 bits of RSA is equivalent to the key length of 163 bits of elliptic curve crypto-system and less memory is enough to store this key. The merits and demerits of ECC are explained in the paper [14]. The time taken to generate the key is also very less when compared to other crypto-system, which increases the level of security. This paper includes the complex mathematical background of ECC and its practical use cases in the industry, common implementation mistakes, performance comparison of ECC with RSA crypto-systems, etc. The merits and demerits of ECC are explained. This comparative study is not focused on the large group of weak elliptic curves.

The purpose of paper "Using image steganography for providing enhanced medical data security" [15] is to provide an introduction to pairing-based cryptography. It should be bi-linear, non-degenerate and efficiently computable. They also present some of the important developments in protocol design, Tate pairing computation, and elliptic curve selection. The bi-linear Diffie-Hellman problem (BDHP) and its hardness, the three fundamental pairing-based protocols such as identity-based encryption, short signatures, and three-party one-round key agreement protocol, etc. are also explained in this paper. It fulfills the three requirements of a good steganography approach. It takes more time for processing. Dustin Moody et. al in their paper [16], used Blowfish algorithm and watermarking technique for image encryption. We can apply this symmetric key algorithm to any color image or black-white image of any size with any extension (.png, .jpg, .bmp, .tif, etc.). Normally 16 rounds are used in Blowfish; if the number of rounds is increased then the algorithm becomes stronger. It cannot be broken until an attacker tries [(28 * rounds) + 1] combinations. It is an excellent standard encryption algorithm without any weak points. But a secure channel is needed to send symmetric key to the other side for decryption. The main advantage is it provides enough security and is more robust. Different approaches used so far, for securing electronic medical images to provide privacy and security are shown in table 1.

A. Survey Implications

Nowadays healthcare cybersecurity is one of the top challenges for 2019 to maintain consumer trust in sharing of patient data based on the annual HCEG (Healthcare Executive Group) Poll [17]. Thirty percent of Big Data security attacks reported during 2009-2016 according to AJMC (American Journal of Managed Care) [18]. The

majority of security professionals believe that either malicious or unintentional data access from an insider breach is an organization's greatest security threat. Based on this study we can say that Healthcare cybersecurity entails the following:

1. Protecting privacy: solving the patient unique identification issue is must.
2. Improving data security: for sharing of patient data across different caregiving organizations to provide better care.
3. Preventing ransomware attacks: from infecting the edge network.

Table-1: Pros and cons of various approaches used

Sl. No.	Approach used	Pros	Cons
1	Hash-Solomon Encoding Algorithm	- Pieces of information are stored in different layers - Provides enough privacy	- Takes more time for retrieving data from different layers -Needs complex calculations
2	Software Decoys	-Synthesizes, automatically, a Series of bogus software programs to confuse the attacker	- Not good for outsider attack
3	Image Steganography	- Provides security	- Requires more time for processing
4	Pairing-Based Cryptography using Elliptic Curve Cryptography(ECC)	- Faster than RSA cryptosystem - Less memory is enough to store key value	- The study is not focused on the large group of weak elliptic curves
5	Blowfish algorithm and Watermarking	- More Robust - Faster than other algorithms	- The need for a secure channel

This literature review helps to identify IT security safeguards. These are:

1. Access control Providing access permissions or privileges.
2. It controls Mechanisms used to record and examine information system activities
3. Integrity controls Policies and procedures used to alteration.
4. Transmission security prevents unauthorized access during transmission.

Cisco reports [19] that by 2021, the storage capacity of the data center will increase to 2.6 ZB, up from 663 EB in 2016. Furthermore, globally, the data stored in the data centers will nearly quintuple by 2021 to reach 1.3 ZB by 2021, up 4.6-fold from 286 EB in 2016 [20]. Therefore the security gets more attention in this era of computing. Based on this literature study we can conclude that by using the Blowfish algorithm along with the Diffie-Hellman key exchange protocol, we can resist all type of outsider attack. An insider attack can be resisted by providing double authentication mechanism along with decoy document generation by using invisible digital watermarking method or various image processing techniques.

IV. PROPOSED MODEL

The proposed system is a variation of the model proposed in paper [6]. It takes a medical text, a medical image or a surgery video as input and provides a high level of security with the help of DMD in the fog server. A user, fog server and cloud server are different three parties involved in the model. The design of the proposed method is shown in Fig: 2 and the block diagram of data-flow and communicational details are also shown in Fig: 3.

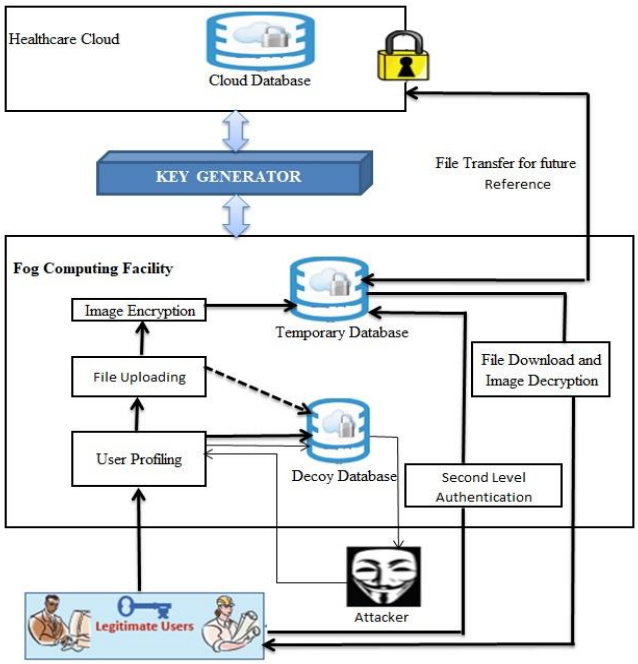

Figure 2: Proposed Model

A. Centralized Cloud server

An Amazon Elastic Compute Cloud (Amazon EC2) server is used as a centralized cloud server. It uses public–key cryptography to encrypt data with a public key such as a password. At the receiver side, it uses a private key to decrypt the data. We must create SSH-2 RSA keys when we launch an instance. In this project we launched a Linux instance with the public key content is placed in an entry within ".ssh/authorized_keys" to securely access our instance. The key size we used here is of 2048-bit and saved it as a private key in a secure place. Otherwise, anyone who possesses our private key can decrypt our data. The public key is stored within Amazon EC2.

B. Decentralized Fog Node

Based on different regions, different devices such as laptops, server machines, access points can be set up as fog node/server, which is a decentralized mechanism to reduce time delay for data transmission. Fog devices can be owned by users or service providers and can be physically observed, imposed or verified by their organizations. Many existing computing networks rely on central authorities, which can be expensive, inefficient, and time-consuming processing. This decentralized system helps to eliminate the need for central authority in computing networks.

These fog nodes are capable of handling the following actions:

1. Send or receive requests with arbitrary operations to/from other fog devices/fog servers.
2. Processes data obtained from the end users.
3. Send data to the cloud server which requires big data processing.

C. Network Setup

To implement computing environment we use two or more systems such as a laptop, mobile devices, etc., are connected with a fog node/server using IP address addresses over the internet. This fog node/server is connected with a cloud server through SFTP (Secure File Transfer Protocol) supports secure key-based authentication. FTP (File Transfer Protocol) is not so secure to transfer commercial files on all modern OS. A third party tool "Filezilla" is used SFTP client to connect with other computers. Key-based Authentication is done with SFTP using passwords and SSH keys. In this work, we used the SSH-2 Key-based authentication mechanism to save SSH keys. FileZilla has a built-in key management page to save keys securely and which allows a connection to a remote server automatically. The developed project codes are transferred to the remote server using this third-party tool.

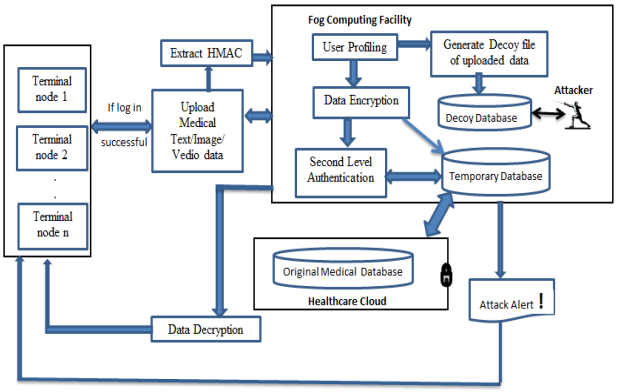

Figure 3: Block Diagram

D. Different Modules:

1. User authentication and HMAC extraction in the fog server.
2. Secure Communication for mutual agreement: Private Key Generator server generates private and public keys using Diffie-Hellman elliptic curve cryptographic method.
3. Image Encryption and Decryption: Blowfish algorithm is used.
4. Decoy Image Creation in the fog while sending an encrypted image to the cloud.
5. V2I Binding Algorithm for Secure Video Files
6. The performance evaluation can be carried out in this module.

E. User Authentication and HMAC Extraction

In the first module hybrid user behavior profiling method is used for user profiling, which helps to detect any unusual data access pattern and to find whether a user is legitimate or not. This module also includes extraction of HMAC information. Then attach this information in the header part of the data.

The users are facilitated here to authenticate and thus, ensure that only valid users can access the application. The cloud users have some clear-cut idea of their own when they are accessing services. That is their preferences, search pattern, duration of access time, types of operations, etc. Moreover, they know the file name that is uploaded and its contents. So we need to keep a log record using a hybrid user behavior profiling algorithm. So that accordingly we can redirect the users to the decoy application to avoid unauthorized access. This can be done based on the log record details under the following assumptions:

1. Only a narrow search performed by a legitimate user while looking for a particular file.
2. An Intruder exhibits a wide search pattern to steal data because he does not have complete awareness about the files and their contents. He has to open every file to find interesting data.

The second parameter to collect user's real behavior is their mouse activities. That is mouse clicks per user session, average distance traveled by mouse per session, session time, login time, mac address information for creating user profile, etc., are collected by the system to track user's real-time behavior and this information is used to find legitimacy by comparing with the earlier patterns whenever each session begins. If there exists deviation more than a particular threshold then it is taken as a potential attack and we have to take some kind of attention to reduce false positives [24] (System suspects an attack when it is not).

F. 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. This is the resultant HMAC code. The entire steps are illustrated in Fig: 4.

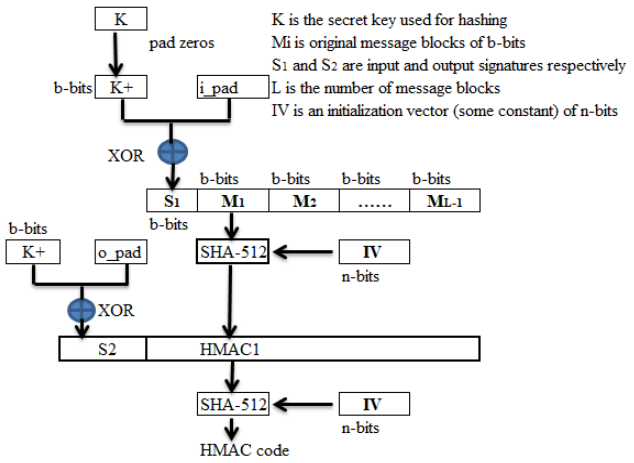

Figure 4: Extraction of HMAC

G. 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.

H. Image Encryption and Decryption

In the third module, 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.

I. Blowfish Algorithm

Blowfish algorithm [22] is used for encryption and decryption of data. Its key size varies from 32 bits to 448 bits. It is a fast and free alternative symmetric algorithm in which block ciphers of size 64 bits are used. Input data is divided into fixed-length blocks during encryption and decryption. If the size is not multiple of eight then it must be padded. The first part of the algorithm is a key-expansion part converts a key of at most 448 bits into several subkey arrays totaling 4168 bytes. The second part is a data encryption part that occurs via a 16-round Feistel (symmetric structure) network.

Key Expansion

Now we have to break the original key which ranges from 32 bits to 448 bits [21], into a set of sub-keys. These keys are stored in an array, as follows:

K1, K2,…..., K14 where each key block contains 32 bits.

There is a p-array consisting of eighteen 32-bit sub-keys:p1, p2, p3, p4,…., p18 and four 32-bit S-boxes contain 256 entries each.

That is:

S1_0, S1_1, S1_2, S1_3, ….. ,S1_256
S2_0, S2_1, S2_2, S2_3, ….. ,S2_256
S3_0, S3_1, S3_2, S3_3, ….. ,S3_256
S4_0, S4_1, S4_2, S4_3, ….. ,S4_256

The following steps are used to calculate the sub-keys are shown in algorithm 1. In this process, totally 521 iterations are needed to calculate new sub-keys for all the P-array and the four S-boxes. Basically, S-boxes are substitution boxes.

Blowfish Key Expansion Algorithm

1. Initialize the p-array and S-boxes with a fixed string of hexadecimal digits.
2. XOR p-array with the key bits that is the first 32 bits of the key is XORed with P1 and the second 32 bits of the key are XORed with P2 and so on, as follows:
   p1=p1 XOR k1
   p2=p2 XOR k2
   …
   p14= p14 XOR k14
   p15= p15 XOR k1
   …
   p18=p1 XOR k4
3. To get 64 block cipher, encrypt the all-zero string using Blowfish algorithm using sub-keys(p-arrays and S-boxes) obtained from the above two steps.
4. This new output is a 64-bit ciphertext. Divide this into two 32-bit blocks, P1 and P2 i.e., replace p1 and p2 with the output of step3.
5. Encrypt the new P1 and P2 with the modified sub-keys. The resulting output is of a 64-bit block and repeats step 4. The new output is now P3 and P4.
6. Continue the process, to replace all entries of the p-array, and then all four S-boxes in order.

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 operations are XORs and additions on 32-bit words [22]. The steps are shown in algorithm2.

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 file2

The function F is as follows:

1. Divide 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}

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.

J. Decoy Image Creation

The fourth module includes decoy document creation in

the fog computing layer using image flipping and alpha blending methods of image processing. When a user uploads an image, OMD needs to interact with DMD to inform to add decoy file to the decoy medical database (DMD). It is an illusion technique to confuse the attacker. This process carried out in a fog nodes layer to overcome the masquerade attack. A decoy database of decoy files is placed in a fog node to confuse attackers or we can generate decoy files in real-time when user's activity seems doubtful. The steps used are shown below:

1. Read an image as img1
2. Create a flipped image of img1 as img2 and ensure that the size is the same for both images.
3. Then apply alpha blending to create an alpha bitmap with an alpha value.
4. Convert the alpha blended image into a grayscale image, which is the decoy image.

K. Secure V2I Binding Algorithm for securing Video Files and its Extraction

The fifth module includes a Secure V2I Binding algorithm is used for binding medical videos in their own photo in order to provide better security and it can be stored in the cloud server after applying the blowfish encryption algorithm meanwhile decoy image can be stored in fog node. The steps of V2I Binding algorithm are shown below:

1. Upload patient's photo as img
2. Upload his\her surgery video as vd
3. Enter a secret key for a secure binding.
4. Encrypt video file using the Blowfish encryption algorithm
5. Bind this image and video together using delimiters (img + vd)
6. Save the above result, is an image using a random naming mechanism in the server
7. Send a verification code to the user for later use

Table 2: Algorithms and their HMAC code

Hashing Functions	HMAC code
MD4	f685701487aee0218ae9d9a8ce9f89b2
MD5	150c719271d57f794671a677f592744c
SHA1	4c71f3e897514c6275f416b4aeb04989889e3f3d
SHA256	647156e9dbb328400ded5a6c9acde4ec9931ca82494eae6c7ffe3c6e0a2b28cb
SHA512	9cab5781a6ac2f1e1870c23b8dfe1414619e980858e66f1ba6b912c1601b798b949640d8ed6d25aa91716578d7a5d9c8049bd11cbc32c4702b7f711a55765c0c

The uploaded video can be downloaded for future medical reference. A user can view their surgery video with the following steps:

1. Enter the image name in which the video file is bound.
2. Enter their secret key used for V2I Binding
3. Explode the file to unbind video from the photo
4. Decrypt the video using Blowfish decryption algorithm
5. Download their video file

V. Performance Analysis and Results

The performance analysis is also carried out in terms of running time of various algorithms in the fog server and the service time taken for transferring the files to the for server and to the cloud server also analyzed.

A. Message Integrity Analysis

Message integrity analysis can be done using HMAC algorithms. A family of cryptographic hash functions is a Secure Hash Algorithm (SHA). Another family of hash functions is Message Digest (MD). Based on the analysis, SHA-512 is a more preferable algorithm for HMAC code generation than MD. HMAC code produced by different hash functions is shown in table 2.

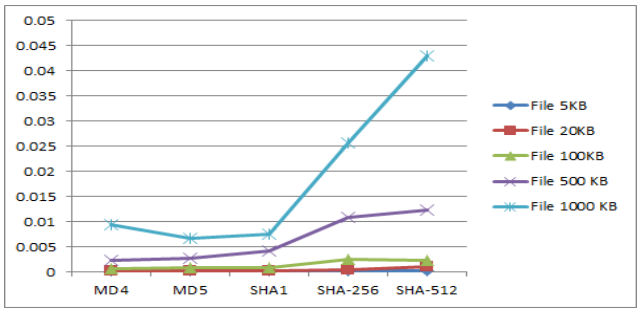

Figure 5: Service Time of HMAC algorithms

The following results show that SHA-512 is better than MD5. The longer hash code is more difficult to break that is why SHA-512 is used in this project. But comparatively, the time taken for execution is more by SHA-512 than MD5 and analysis chart is shown in table 2.

In Fig: 5, the execution time taken by different algorithms is shown, in which the vertical axis represents the time in seconds. This comparison is done with a 32-bit key size.

1. SHA generates more strong hashes than MD5. MD5 can be broken easily. SHA hash code is longer than MD5.
2. Occurrences of collision are less in SHA than MD5. It means that we get the same HMAC code for two different inputs i.e., hashes are not always unique.
3. SHA-512 has a good avalanche effect- when there is a small change in the input, the output changes significantly.
4. SHA-512 has the property that every bit of the hash code is a function of every bit of the input.

B. Analysis of Blowfish Algorithm

According to Fig: 6, the time taken to encrypt an image is more than the time taken for decrypt an image and more time is needed to encrypt/decrypt high-resolution images. But comparatively blowfish algorithm is more efficient than AES and 3DES algorithms.

Table 3: Performance analysis of Video Encryption

Video size	Blowfish	AES	3DES
732 KB	0.0212	0.0265	0.111
732 KB	0.0264	0.0277	0.212
2300 KB	0.072	0.078	0.348
5640 KB	0.169	0.194	0.86
5640KB	0.171	0.206	0.903

In this work, the above-mentioned algorithm is used not only for image files but also to encrypt and decrypt video files. The proposed V2I binding algorithm uses the blowfish algorithm instead of AES and 3DES.

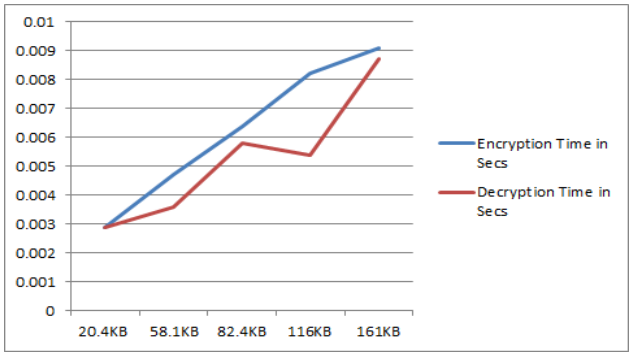

Figure 6: Running Time of Blowfish Algorithm

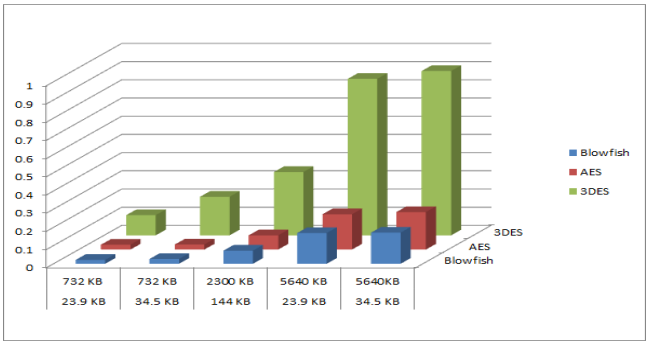

Figure 7: Comparative analysis of Blowfish, AES, and 3DES.

The comparative study is shown in Fig:7 and the time is taken by the algorithms for video encryption is shown in table 3. Based on the analysis we can conclude that 3DES algorithm takes more for video encryption and it is significantly high when compared with AES and Blowfish Algorithm. There is no significant change in performance of AES and Blowfish algorithm when the file size is too small but the efficiency of the Blowfish algorithm is better than AES for resolution video files.

C. Performance analysis of Decoy File Creation

The time taken to run this algorithm with a 2.6 GHz i3 processor is shown in figure8. Based on the analysis, we can conclude that the service time is reduced for high-resolution image files. So the decoy generation algorithm works well for all telemedicine services.

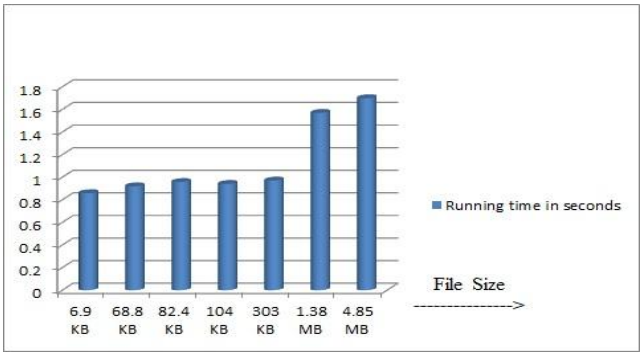

Figure 8: Running Time of Decoy Algorithm

D. Performance analysis of Cloud and Fog Computing

The performance analysis is done in terms of transmission time required for uploading files to the cloud server and to the fog nodes. It is observed that the time needed for uploading files to the cloud server is more than to the fog node. The processing time is also more in the cloud server. Sarkar et.al in their paper [1] explained the service latency, processing latency and transmission latency required for cloud and fog server in detail. Therefore we can conclude that fog computing technology is more efficient when real time applications increase.

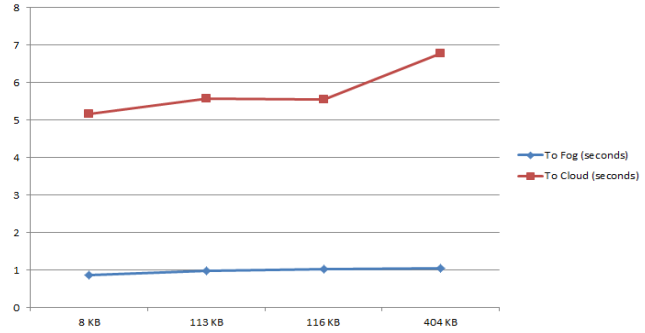

Figure 8: Comparison between cloud and fog computing

The graph in figure 9 also shows that the same. The performance of fog node is more efficient when real-time applications increase.

VI. CONCLUSION

These works mainly focus on securing patient's digital data within the cloud computing environment with the help of fog computing technology. First of all, a legitimate user can access original medical databases after verifying their authenticity twice. Here we need to set a decoy medical database to confuse attacker while the original medical database is kept hidden in a cloud data-center. The data from the decoy database can be returned if the authentication failed. The proposed new V2I binding algorithm is a more efficient method for securing surgery videos or private videos in their own profile photo. The analysis shows that the proposed method works well than the other cryptographic algorithms.

REFERENCES

1. Hamid HAA, Rahman SMM, Hossain MS, Almogren A, Alamri A. (2017), "A security model for preserving the privacy of medical big data in a healthcare cloud using a fog computing facility with pairing-based cryptography", in *IEEE Access*, vol.5, pp. 22313-22328.
2. Varghese B, Wang N, Nikolopoulos DS, Buyya R. (2017), "Feasibility of fog computing", in arXiv preprint arXiv:1701.05451.
3. Bernard Marr (2016, April 10), "What Is Fog Computing? And Why It Matters in Our Big Data and IoT World", Available: https://www.forbes.com/sites/bernardmarr/2016/10/14/what-is-fog-computing-and-why-it-mattersin-our-big-data-and-iot-world/.
4. Enabling Business Strategies With Cloud Computing, Lunarpages, 07-Mar-2018. Available: https://lunarpages.com/cloud-computing-business-strategies/.
5. Fog Computing Market worth 203.48 Million USD by 2022. Available: https://www.marketsandmarkets.com/PressReleases/fog-computing.asp
6. AWS Marketplace: Splunk Cloud. Available:

https://aws.amazon.com/marketplace/pp/B06XK299KVqd=1495224795283&sr=02&ref_=srh_res_product_title.
7. Wang T, Zhou J, Chen X, Wang G, Liu A, Liu Y. (2018), "A three-layer privacy preserving cloud storage scheme based on computational intelligence in fog computing", in *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 2, no. 1 pp. 3-12.
8. William R Claycomb and Alex Nicoll (2012), Insider Threats to Cloud Computing: Directions for New Research Challenges", in *IEEE 36th Annual Computer Software and Applications Conference, Izmir*, pp. 387-394.
9. Sarkar S, Chatterjee S, Misra S. (2018), " Assessment of the Suitability of Fog Computing in the Context of Internet of Things", in *IEEE Transactions on Cloud Computing*, vol. 6, no. 1, pp. 46-59.
10. Stolfo SJ, Salem MB, Keromytis AD. (2012), "Fog Computing: Mitigating Insider Data Theft Attacks in the Cloud," in *IEEE Symposium on Security and Privacy Workshops*, pp. 125–128.
11. Park Y, Stolfo SJ. (2012), "Software Decoys for Insider Threat", in *Proceedings of the 7th ACM Symposium on Information, Computer and Communications Security*, New York, NY, USA, pp. 93–94.
12. Kupwade Patil H, Seshadri R. (2014), "Big Data Security and Privacy Issues in Healthcare", in *2014 IEEE International Congress on Big Data, Anchorage, AK*, pp. 762–765.
13. Shreyank N Gowda (2016), "An advanced Diffie-Hellman approach to image steganography," in *2016 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, Bangalore, pp. 1-4.
14. Krists Magons (2016), "Applications and Benefits of Elliptic Curve Cryptography", *in SOFSEM (Student Research Forum Papers/Posters)*, p. 11.
15. Muhammad Arslan Usman (2018), " Using image steganography for providing enhanced medical data security," in *15th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, pp. 1–4.
16. Moody D, Peralta R, Perlner R, Regenscheid A, Roginsky A, Chen L. (2015), "Report on pairing-based cryptography ", in *Journal of research of the National Institute of Standards and Technology*, 120, p. 11.
17. Healthcare Cybersecurity Is a Top 2019 Executive Challenge, HealthITSecurity. Available: https://healthitsecurity.com/news/healthcare-cybersecurity-is-a-top-2019-executive-challenge.
18. Hospital Data Breaches Most Common, Affect the Most Patients, HealthITSecurity. Available: https://healthitsecurity.com/news/hospital-data-breaches-most-common-affect-the-most-patients.
19. Cisco Global Cloud Index: Forecast and Methodology, Cisco. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/global-cloud-index-gci/white-paper-c11-738085.html.
20. Utilizing Healthcare Data Security, Cloud for a Stronger Environment, HealthITSecurity. Available: https://healthitsecurity.com/news/utilizing-healthcare-data-security-cloud-for-stronger-environment.
21. A. Kahate, Cryptography and Network Security, Tata McGraw-Hill Education, 2003.
22. S. Mudepalli, V. S. Rao, and R. K. Kumar, "An efficient data retrieval approach using blowfish encryption on cloud ciphertext retrieval in cloud computing," in 2017 International Conference on Intelligent Computing and Control Systems (ICICCS), 2017, pp. 267–271.
23. Stallings W. Cryptography and Network Security: Principles and Practice (5th edition), Prentice Hall: 2011.
24. Attenuating Masquerader Data Theft Attack in the Cloud, ResearchGate. [Online]. Available: https://www.researchgate.net/publication/316514654_Attenuating_Masquerader_Data_Theft_Attack_In_The_Cloud. [Accessed: 31-Mar-2019].

AUTHORS PROFILE

Ms. **Elizabeth, M J** received the B.Tech. degree in Computer Science and Engineering from Vimal Jyothi Engineering College, Chemperi, Kannur University, Kerala in 2012 and she is currently pursuing her M.Tech. at Viswajyothi College of Engineering and Technology, Abdul Kalam Technological University, Trivandrum, Kerala, India.

Mr. Jobin Jose received his B.Tech. degree in computer science and engineering from Bharathiar University, Coimbatore in 2004 and the M.Tech. degree in Computer Science and Engineering from Karunya University, Coimbatore in 2008 and he is pursuing his Ph.D. degree from NIT Trichy. His current research interests include advanced computer architecture and computer networks.

Ms. Dona Jose is currently working as Assistant Professor in Dept. of Computer Science and Engineering at Viswajyothi College of Engineering and Technology, Kerala. She did her B.Tech(2006) and M.Tech(2010) from Mahatma Gandhi University College, Thodupuzha and Amrita University, Ettimadai respectively. She has authored several national and international publications to her credit with research interest in Algorithms and Machine Learning. She has been into teaching for the last eight years.