International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

A Novel Blockchain-Based Lightweight Encryption Technique in Fog Based IoT for Personal Healthcare Data Application

Janardhana Naidu J. N. S. S. [1], Ganesh E. N.[2]

Submitted: 04/11/2022 **Accepted:** 03/02/2023

Abstract: Different sensors, digital devices, and objects may all sense their surroundings thanks to the Internet of Things (IoT) concept, which also links them to the worldwide Internet for data exchange. Integrating Edge, Fog, and Cloud infrastructure is becoming more necessary to enable Internet of Things (IoT) applications that are both latency-sensitive and computationally heavy. Security in the IoT, particularly for healthcare applications, is a big issue. In this research, a unique, lightweight encryption method for Internet of Things (IoT) based on fog is developed. For the execution and interaction of IoT applications and compute instances, Fog Bus provides a platform-independent interface. It lets service providers manage their resources as well as customers run numerous apps simultaneously and developers create applications. In order to safeguard activities on sensitive data, Fog Bus also uses Blockchain, authentication, and encryption mechanisms. For enhancing security, the blockchain with Tiny Lightweight Symmetric Encryption (TLSE) is introduced. The TLSE chooses the best key by taking the Aquila Optimization Algorithm into account (AOA). The UCI machine library is used to gather health care data in order to verify the proposed approach. The suggested method is put into practice using Python, and it is contrasted with more established methods like Rivest-Shamir-Adleman (RSA) and elliptical curve cryptography (ECC), respectively.

Keywords: *blockchain technology, fog computing, encryption technique, tiny lightweight symmetric encryption, fog-Bus and aquila optimization algorithm.*

1. Introduction

By 2025, 39% of all IoT devices will be used in the healthcare sector. At the same time, a study by a clever business [1], Tractia, predicts that by 2025, the yearly revenue for current uses of blockchain would reach $19.9 billion. IoTs are presently employed widely in the healthcare industry to allow patients and professionals to constantly communicate with management. Time-sensitive applications are necessary for certain medical services, such as ECG and EEG testing, which necessitates ongoing analysis of patient health data and medical reports (PHD). By using IoT devices for healthcare in medical institutions and organizations, all of these may be imagined. However, the growing usage and increased scale of IoT devices have increased the volume and reality of information traffic [3]. The management of heavy IoT information traffic has grown to be a serious issue and cause for worry when employing integrated cloud server components [4].

As a consequence, dangers pertaining to patient security and privacy have increased. Openness, information seeking, accountability for well-being information, and partial safety are all risks in the name of patient safety [5]. Now that information can be duplicated and IoT devices for medical services may have different personalities, hackers and programmers can more easily traverse the IoT network. The IoT-cloud scenario is now facing a number of challenges [6], including poor connections caused by concentrated activity, retaliation attacks, security leaks, and surveillance of reported IoT devices. sharing of data between medical services IoT and cloud need trust, device identification, client network security, and confirmation for safe PHD exchange. medical assistance [7] IoT has a built-in scalability and flexibility issue. Gigabytes (GB) of healthcare data are still being produced by healthcare IoT, which presents an unexpected obstacle to its integration with blockchain [8]. Additionally, the variety of medical IoT devices highlights the question of security in IoT businesses [9]. Healthcare security and insurance Information from IoT devices has a clear source that can be identified. For instance, if not verified, hackers or attackers may hijack the ECG IoT device, pose as the genuine sensor, and provide fraudulent readings [10].

1Department of Computer Science and Engineering,
Vels Institute of Science Technology and Advanced Studies,
Chennai, Tamil Nadu 60011, India.
Email: jnss.janardhana@gmail.com

2Department of Computer Science and Engineering,
Vels Institute of Science Technology and
Advanced Studies, Chennai, Tamil Nadu 60011, India.
Email: enganesh50@gmail.com

For important medical care IoT devices, blockchain-based systems need enormous amounts of power, delay, and processing [11]. With distributed registration, it is possible to foresee the reproducibility of patient data across several edge-located fog centers. Hash hubs and blockchain both operate in a decentralized and proprietary manner. The change from centralized control to decentralized control has led to the popular adoption of both FC and blockchain [12]. Classified patient information may be recorded in several fog centers in order to provide IoT devices immediacy, security, and character. In a composite model, FC hubs may be utilized for mining purposes. In order to solve the issue of PHD verification and traceable provenance of IoT devices in medical services, FC uses the blockchain paradigm and significantly contributes by providing IoT firms with a decentralized and flexible solution at the edge. As miners may gather transaction data in batches, equipped FC hubs can observe legality. By decreasing parcel error during PHD transmission, FC gets over the healthcare IoT bottleneck caused by high data traffic [14]. The IoT-FC-Cloud architecture's safety and security issues may be resolved and overcome via blockchain [15].

The remainder of the article is already set out as follows, with section 2 mentioning similar studies on security in fog computing. Section 3 provides a detailed discussion of the anticipated method. The results are described in Section 4. In section 4, the paper's summary is offered.

2. Related Works

To establish security in the architecture of fog computing, many solutions are available. This section only reviews a few works.

In order to support Internet of Things (IoT) applications in smart urban communities, Parminder Singh *et al.* [16] have proposed the Blockchain and Fog-based Architecture Network (BFAN). The suggested solution uses blockchain, encryption, and authentication to safeguard physical data. It allows architects and system designers to deploy applications with a focus on smart cities. The suggested design aims to decrease inertia and energy consumption while ensuring further enhanced security measures. The redesign's outcomes show that the suggested engineering outperforms the current structures for the relevant metropolitan neighborhoods.

A three-stage architecture, an intelligence model, a numerical system, and advanced signature-based encryption (ASE) computation for IoT device identification proof, verification, and Patient Health Data (PHD) Verification have been described by Saurabh Shukla *et al.*, [17]. From a number of administrations, the healthcare IoT and the growth of secure connectivity to end users were beneficial. In fact, the suggested model and computation ought to provide a reliable technique to support various near-edge transmission and transmission kinds. By dividing the output of the unique ASE calculation under consideration into performance, bundle fault, reliability, and dangerous hub detection precision.

Elliptic Curve Diffie-Hellman Ephemeral (ECDHE) Key Trading Computation with Pre-Shared Key (PSK) has been developed by Sanaz Amanlou *et al.* [18] as a compact and secure confirmation system taking into account Haze gateway and IoT device. The buy-in convention is distributed via Message Queuing Telemetry Transport (MQTT) in a dispersed hash processing scheme. The perfect forward secrecy (PFS) component of the proposed ECDHE-PSK verification technique enhances the test's security via remarkably quick and reliable PSK calculation.

The Message Queue Telemetry Transport (MQTT) protocol over SSL/TLS has been established by Sejal Gupta *et al.*, [19]. Elliptic bend ElGamal cryptography calculation is offered since MQTT was defenseless against spying. This computation adds a homomorphic component and subsequently guards against man-in-the-center assaults. By shifting information to clouds and fog according to the information object, the effective core transformation and associated information loading presented in F and Flow research works aid to preserve core energy.

A microservice-based blockchain component for hashes has been presented by Whaiduzzaman *et al.*, [20] and serves as a decentralized client-server network medium (i.e., got gadget-based correspondence). Here, a security system created for a blockchain is used. Look at the encryption and decoding latencies from the Internet of Things, via integrated components, and a security system designed to check information logs in this investigation. Blocks on a blockchain combine existing calculations, particularly AES, RSA, and DES, to provide superior processing outcomes in terms of algorithmic productivity.

3. Proposed System Model

With the use of software devices that provide platform agnostic execution interfaces as well as organized communication, the fogbus system links many hardware components. Figure 1 depicts a high-level perspective of a linked IoT-Fog cloud system utilizing fogbus.

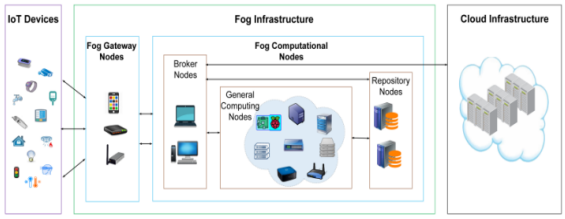

Fig. 1. Fog architecture

Different applications in the fog computing layer prioritize security as a top priority. Blockchain technology and

effective encryption techniques have been developed to address security concerns. The databases are first gathered from the open-source system. Following that, blockchain technology is created to separate the data into separate blocks and store them with a hash function and security key. The data is then encrypted and stored using the encryption process. The AOA algorithm is taken into account while choosing the key in the encryption method. The section below provides a detailed overview of the proposed approach.

3.1. Blockchain technology

Blockchain technology has been used in many various applications recently, including ad hoc networks, fog networks, cloud and edge computing, the internet of things, cryptocurrencies, and more. due to its peer-to-peer networking feature, which is described as a group of nodes that behave similarly in order to reduce the chance of single-point failure. Blockchain approaches are being developed in order to allow cost reduction and provide total anonymity using distributed ledgers. Bitcoin is the first blockchain built on the blockchain technology, which is a peer-to-peer network distributed through a timestamp server. The primary function of this bitcoin is to computationally prove the transactions' order. Typically, a chain of digital signatures is how bitcoin is defined. In this section, the blockchain architecture of the bitcoin blockchain is discussed. The blockchain is made up of a series of interconnected blocks that keep track of a long history of transactions in the style of the conventional public ledger. The parent block and a prior block hash are the sole blocks that make up the block header in figure 2. The block connected to the hashes of the block's predecessors may likewise be stored in the Ethereum blockchain. Typically, the blockchain is used as a genesis block without a parent block [21].

Block: Its basic definition is the block header, and its term for the body is block body. Block version, timestamps, nBits, Nonce, and Parent block hashes are all included in the block header. The block version specifies the guidelines that block verification must adhere to. Uneven cryptographic techniques are employed in blockchain technology to confirm the validity of messages. The digital signature is controlled by uneven cryptography, which is used in dishonest and unreliable environments [22].

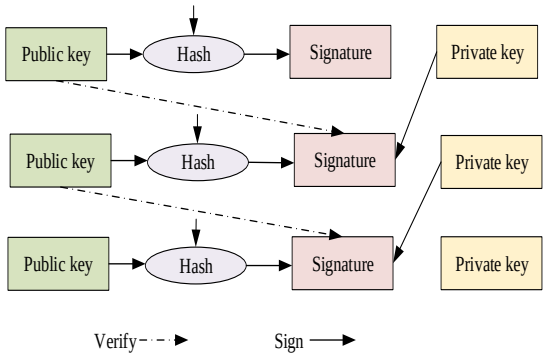

Fig. 2. Basic Architecture of the blockchain technology

Digital signature: Every user of blockchain technology is an owner since they have both a public key and a personal key. The secrecy of the personal key may not be guaranteed. For this reason, the blockchain's secrecy is strengthened by the use of the digital signature. The digital signature system has two stages: the signing stage and the verification stage. It is entirely safe. Data is maintained in blocks and safeguarded with the use of blockchain technology. After that, a new encryption method is created to improve data security.

3.2. Tiny Lightweight Symmetric Encryption

The proposed small symmetric encryption technique is designed to dynamically increase key confusion in each round. This method uses 128-bit key generation and 64-bit plaintext consumption. This method takes into account 32 cycles, each of which may include two rounds and result in 64 rounds [23]. The plaintext may first be split into two phases, $P0\ and\ P1$, each of which has 32 bits. Every part of plaintext is subject to the round operation op. Four 32-bit partial keys, designated as $k1, k2, k3\ and\ k4$ may be created from a single 128-bit key. As a result, two partial keys are taken into account: $k1\ and\ k3$, which are used for rounds with odd numbers, and $k2$ and $k4$, which are used for rounds with even numbers. The golden ratio is represented by the compute key schedule constant $K_{SC} = floor(2^{31}/\emptyset)$. The golden ratio is computed as $(1 + \sqrt{5/2})$. and taken as 1.61803. As seen below, the round function is calculated.

$$P0 += \left((P1\ lshift\ 4)\ AND\ K0\right)\ XOR\ (P1\ AND\ K_C)\ XOR\ (P1\ rshift\ 5)\ AND\ k1) \quad (1)$$

And I, I is even,

$$P1 += \left((P0\ lshift\ 4)\ AND\ K2\right)\ XOR\ (P0\ AND\ K_C)\ XOR\ (P0\ rshift\ 5)\ AND\ k3) \quad (2)$$

The first cycle generates partial keys like k0, k1, k2, and k3.

When the second cycle begins,

$K0$ is assumed to be a constant parameter for the odd round, while $K1$ is modified throughout the whole odd round, which is provided as follows.

$$K1 = K1 + \left(K0\ XOR\ \left(XTRACT\ (P0)\right)\right) \quad (3)$$

K2 is assumed to be constant in the even round, but K3 varies for all even rounds.

$$K3 = K3 + \left(K2\ XOR\ \left(XTRACT\ (P1)\right)\right) \quad (4)$$

To compute the integer parameter in the range of 0 to 32 from P0 and P1 connected to the variable being processed, the function's XTRACT () is used. This parameter's

definition states that it serves as an index to an array that may be generated on the fly based on the choice of the key parameter. This method will return the array parameter to which the computed index parameter points. As a result, the key ambiguity might be created dynamically and cannot be discovered prior to execution. In addition, the parameter varies with each algorithm run.

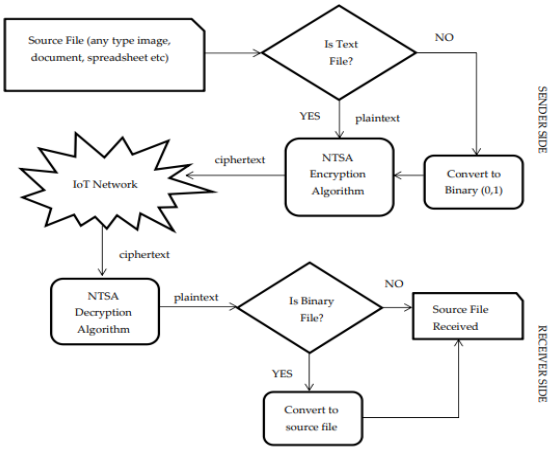

Fig. 3. Proposed encryption technique process.

The input data is first divided into chunks using blockchain technology. To produce the cipher text, the block data is transmitted to the encryption algorithm. Data from the input source is converted into a binary file, which is displayed as zeros and ones. With the use of a secret key that has been verified with sender and receiver agreement, the binary format may be passed to the proposed encryption method to produce encrypted text. The decryption method converts the encryption text to plain text once it reaches the recipient end. Binary data that has been modified and converted to input format. The AOA algorithm is used to produce the key in the encryption process.

3.3. Aquila optimization Algorithm

The most effective raptors are the aquila, which are found in the northern hemisphere. Aquila is a widely distributed species of aquila. Accipitridae, sometimes known as the related group of complete birds, includes aquila. Aquila are often dark brown in color with golden brown feathers on their necks. White spots may sometimes represent a little wing, although the aquila of this cluster is dependent on the color of the tail in addition to generally [24]. Aquila may catch a variety of prey, including ground animals, squirrels, marmots, deeps, hares, and rabbits, by using agility, speed, strong feet, and talons that are sharpened. The aquila's hunting habits may be stated as follows:

- The first method, a vertical stoop with a high soar, may be used for birds' in-flight hunting behavior since the aquila becomes better at a greater altitude. Aquila begins a lengthy, low-angle glide while its prey is being investigated, with the idea of increasing speed as they draw closer to one another. For this approach to be effective, the aquila needs a significant advantage over its victim. The tail, wings, and feet may all be extended before to the strike in order to make the attack on the prey seem like a thunderclap.
- The following maneuver, a brief glide assault with contour flying, is the most effective one that is often used by aquila because it allows it to improve at a lower level relative to the ground. If the target is capable of flying or running, it can almost always be pursued. Using this method to capture seabirds, nesting grouse, and ground squirrels may be advantageous.
- The following maneuver involves low flying and a gradual descending attack. The quila advances into the prey assault position while also lowering to the ground throughout this procedure. The aquila choose its target and then settles on the prey's back while also attempting to pierce. This kind of hunting may be used on sluggish prey, such as tortoises, foxes, hedgehogs, rattlesnakes, or any species without an escape reaction.
- The third strategy is for the quila to walk on the ground and try to drag in its prey while simultaneously walking and grabbing it. It may be used to remove the young of larger prey from the covered area.

In conclusion, aquila are perhaps the most effective, skilled, and successful hunters after humans. The following is how the mathematical modeling of the AO approach is presented:

Initial population

In AOA, the optimization procedure might start with the population of potential solutions (X), which is shown below the equations. The lower limit and upper bound of the particular problems may be used to generate this population stochastically. The best responses from each iteration are combined to create the best solution.

$$x = \begin{bmatrix} x_{1,1} & \cdots & x_{1,j} & x_{1,dim-1} & x_{1,dim} \\ x_{2,1} & \cdots & x_{2,j} & \cdots & x_{2,dim} \\ \cdots & \cdots & x_{i,j} & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n-1,1} & \cdots & x_{n-1,j} & \cdots & x_{n-1,dim} \\ x_{n,1} & \cdots & x_{n,j} & x_{n,dim-1} & x_{n,dim} \end{bmatrix}$$

=

Here, n may be regarded as the total number of candidate solutions, $x_i$as the decision values, x as the collection of available candidate solutions, and dim as the issue's dimension size.

$$x_{ij} = Rand \times \left(ub_J - lb_J\right) + lb_J, I = 1,2,\ldots,n\ J = 1,2,..,dim$$

In this case, lb_J is the j[th] lower bound and ub_J is the j[th] upper limit of the provided problem.

Fitness Evaluation

The minimum encryption time-based key is chosen from the random key parameters in the fitness assessment. The following is an explanation of how this system's fitness is assessed:

$$FF = Min(encryption\ time)$$

Mathematical design of AO

The activities of each stage of the hunt are shown using the projected AO method, simulating an aquila's character throughout a hunt. The four techniques used here to describe the optimization process of the projected AO technique are: selecting the search space by high soar with vertical stoop; exploring within a diverge search space through short glide attack with contour; and exploiting within a coverage search space through slow descent attack and low flight in addition to swooping through grab and walk. Distinct parameters relating to various situations are used to transition the AOA method from the exploitation to the exploration stages. The exploitation stages will then be properly handled, and the exploratory phases will be thrilled [25].

Phase 1: Expanded Exploration

The aquila uses a vertical stoop and a high soar to locate the prey and choose the best hunting site in its first strategy. Additionally, the AOA can often be computed from a high altitude to determine the search area.

$$x_1(T+1) = x_{best}(T) \times \left(1 - \frac{T}{t}\right) + (x_M(T) - x_{best}(T) * RAND)$$

Here, the variables $x_M(T)$ and$\left(1 - \frac{T}{t}\right)$ can be defined as the mean parameter locations parameter of the current solutions correlated with the t[th] iteration, $x_{best}(T)$ can be defined as the optimal achieved solution up until the t[th] iteration, and $x_1(T+1)$can be defined as the solution of the upcoming iteration that can be created using the search technique. Following are examples of the maximum and current iterations:

$$x_M(T) = \frac{1}{n}\sum_{I=1}^{n} x_I(T), \forall_J = 1,2,\dots,dim$$

Here, n may be thought of as the number of potential solutions, and dim as the size of the problem's dimensions.

Phase 2: Narrowed exploration

In the second method, the aquila soars over the reference prey, attacks the prey, tests the land, and confronts the target prey, from which the position of the prey may be determined. This method is known as the contour flying short glide assault. Additionally, while creating the assault, the AO specifically investigates the position of the reference prey.

$$x_2(T+1) = x_{best}(T) \times Levy(d) + (x_R(T) - (Y - X) * |RAND)$$

Here, $Levy(d)$ can be defined as the levy flight distribution function, d is defined as the dimension space, and $x_2(T+1)$ is defined as the solution of the following iteration that can be obtained by the second search technique. $x_R(T)$ is defined as the random solution in the period of [1,n] at the i[th] iteration. The formulation of the Levy flight function is as follows:

$$levy\ (d) = S \times \frac{U \times \sigma}{|V|^{\frac{1}{\beta}}}$$

Here, S is defined as the constant parameter values set to 0.01 and σ may be considered as the constant parameter. U and V can be defined as random parameters between 0 and 1.

$$\sigma = \left(\frac{\Gamma(1+\beta) \times sine|\left(\frac{\pi\beta}{2}\right)}{\Gamma\left(\frac{1+\beta}{2}\right) \times \beta \times 2^{\left(\frac{\beta-1}{2}\right)}}\right)$$

In this case, β the constant parameter may be specified as 1.5. In the search space, Y and X may be used to create the spiral form that is calculated as follows:

$$Y = R \times Cos(\theta)$$

$$X = R \times Sin(\theta)$$

$$R = R_1 + U \times d_1$$

$$\theta = -\omega \times d_1 + \theta_1$$

$$\theta_1 = \frac{3 \times \pi}{2}$$

Here, U is defined as the small parameter that is fixed to 0.0565, and R_1 is defined as the parameter between 1 and 20 for the fixed number of search cycles. ω can be defined as the small parameter that is taken as 0.005, d_1 can be defined as the integer numbers from 1 to the length of the search space.

Phase 3: Expanded exploitation

In the third method, the aquila descends ideally with a preliminary attack to determine the prey position, and in addition, the aquila may be ready for attack in addition to assault. This approach may be described as a low flight and a leisurely descending attack. To carry out the assault and capture the prey, the AO takes use of the target's selected position. The following list of this stage's features is provided:

$$x_3(T+1) = x_{best}(T) - x_M(T) \times \alpha - RAND + \left((Lb - Ub) \times RAND + Lb\right) \times \delta$$

Here, Ub and Lb are defined as the issue's upper and lower bounds, respectively., α, δ is defined as the exploitation adjustment variables and is taken as a small parameter (0,1), $x_M(T)$ is defined as the solution's mean parameter at the t[th] iteration, $x_{best}(T)$is defined as the precise location of the prey, and $x_3(T+1)$ is defined as the solution for the following iteration.

Phase 4: Narrowed exploitation

The fourth strategy involves the quila getting near to its target, attacking it, and flying across the area affected by

stochastic variations. This method is known as "grab and stroll the prey." Additionally, AO strikes the prey's last resting place.

$$x_4(T+1) = qf \times x_{best}(T) - (g_1 \times x(t) \times RAND) - g_2 \times Levy\,(d) + Rand \times g_1$$

Here g_2 is defined as the flight slope's lowering parameter, g_1 is defined as the AOA's various movements used to follow prey throughout the chase, and qf is defined as the quality function used to achieve equilibrium. The fourth search strategy is used to generate the search method, $x_4(T+1)$, which is defined as the answer for the next iteration. The mathematical formulation of the current iteration of this technique is as follows:

$$qf(t) = t^{\frac{2\times RAND(\ )-1}{(1-T)^2}}$$

$$g_1 = 2 \times RAND(\ ) - 1$$

$$g_1 = 2 \times \left(1 - \frac{T}{t}\right)$$

$Levy\,(d)$ is defined here as the levy flight distribution function, T is defined as the number of iterations, $RAND$ is defined as the range between 0 and 1, and qf is defined as the quality function parameter at the i[th] iteration.

4. Outcome Evaluation

In this part, the effectiveness of the suggested technique is evaluated. A laptop with an Intel Core i5-2450M CPU running at 2.50GHz and 6GB of RAM is used to apply the suggested approach in order to verify the existence of the anticipated blockchain-based security in fog computing. Python is used to carry out this procedure. Table 1 provides the recommended method implementation parameters. Utilizing performance indicators like encryption, decryption, waiting, and processing times, the suggested solution is put into practice and confirmed. The proposed method is contrasted with more established methods like RSA and ECC, respectively. The UCI machine library is where the input data is gathered [26].

Table 1 Proposed method parameters

S. No	Method	Description	Value
1	Proposed Method	Number of IoT devices	100
2		File size	5MB
3		Number of iterations	100
4		Number of populations	50
5		Upper bound	10
6		Lower bound	-10
7		Probability factor	0.5

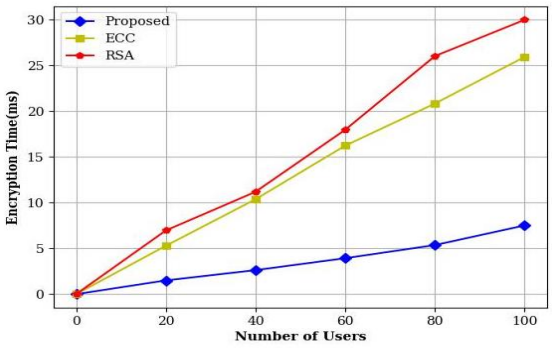

Fig. 4: Encryption time versus users

Figure 4 analyzes and illustrates the encryption time based on users. The proposed method is contrasted with more established methods like RSA and ECC. 20 users in the fog network can reach the 3ms in the predicted approach. Additionally, with a comparable user base of 20, RSA and ECC are accomplished in 7 and 5 milliseconds, respectively. The 20 users' predicted approach is achieved with a short encryption time based on comparative analysis. 40 users in the fog network may accomplish 4 ms using the projected strategy. Additionally, the RSA and ECC are obtained with a comparable user of 40 in 10 and 11 milliseconds, respectively. The 40 users in the comparative investigation show that the predicted approach achieves a quick encryption time. With the proposed method, 60 users in the fog network may accomplish 4 ms. Additionally, the RSA and ECC are obtained with a comparable user of 60 in 16 and 18 milliseconds, respectively. The anticipated approach is achieved with a short encryption time, according to the comparative study of the 60 users. 80 users in the fog network are able to reach the 5ms in the planned approach. Additionally, the RSA and ECC are accomplished with an 80 user at 22 and 26 milliseconds, respectively. The predicted approach is achieved with a short encryption time, according to the comparative study of the 80 users. With the planned approach, 100 users in the fog network can accomplish 6 ms. Additionally, the RSA and ECC are obtained with a comparable user of 100 in 26 and 30 milliseconds, respectively. From the comparative study, the planned approach is achieved with a short encryption time for 100 users.

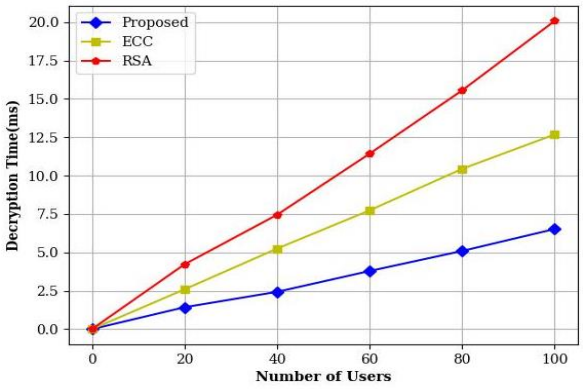

Fig. 5. Decryption time versus users

Figure 5 analyzes and exemplifies the decryption time based on users. The proposed method is contrasted with more established methods like RSA and ECC. In the envisioned approach, 20 users in the fog network reach the 1.8ms. Additionally, the RSA and ECC are obtained with a comparable user base of 20 in 2.5 and 3 milliseconds, respectively. The anticipated approach is achieved low decryption time from the comparative study, the 20 users. 40 users in the fog network may achieve a 2.5ms response time using the projected approach. Additionally, the RSA and ECC are obtained with a comparable user of 40 in 5.2 and 7.5 milliseconds, respectively. The 40 users in the comparative investigation show that the predicted approach has a short decryption time. With the proposed method, 60 users in the fog network may accomplish 4 ms. Additionally, the RSA and ECC are obtained with a comparable user of 60 in 7.5 and 11.5 milliseconds, respectively. The proposed approach is achieved low decryption time from the comparative study, the 60 users. 80 users in the fog network are able to reach the 5ms in the planned approach. Additionally, with an 80 user, RSA and ECC are obtained in 11 and 15 milliseconds, respectively. The proposed approach is achieved low decryption time from the comparative study, the 80 users. With the planned approach, 100 users in the fog network can accomplish 6 ms. Additionally, the RSA and ECC are obtained with a comparable user of 100 in 12.5 and 20 milliseconds, respectively. The anticipated approach is achieved low decryption time from the comparative study, the 100 users.

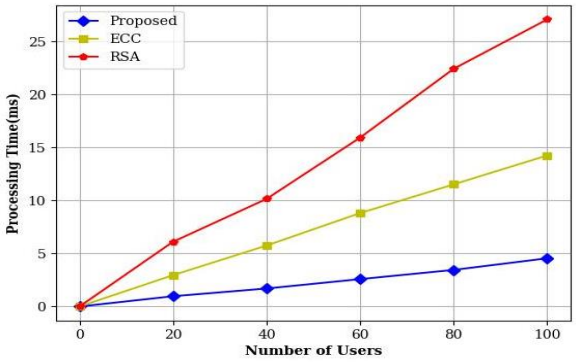

Fig. 6. Processing time versus users

Figure 6 analyzes and illustrates the processing time based on users. The proposed method is contrasted with more established methods like RSA and ECC. 20 users in the fog network are able to attain the 2ms in the planned approach. Additionally, the RSA and ECC are obtained with a comparable user base of 20 in 2.5 and 3 milliseconds, respectively. The 20 users' predicted approach is achieved with a minimal processing time based on comparison analysis. With the planned method, 40 users in the fog network may accomplish 3 ms. Additionally, the RSA and ECC are obtained with a comparable user of 40 in 5.2 and 10.5 milliseconds, respectively. The planned approach is achieved with a low processing time based on the comparative study with the 40 users. With the proposed method, 60 users in the fog network may accomplish 4 ms. Additionally, the RSA and ECC are obtained with a comparable user of 60 in 9 and 16 milliseconds, respectively. According to the comparative study, the planned approach achieves a low processing time for the 60 users. In the envisioned approach, 80 users in the fog network reach the 4.5ms. Additionally, with an 80 user, RSA and ECC are obtained in 14 and 22 microseconds, respectively. According to the comparative study, the planned approach achieves a low processing time with 80 users. With the predicted method, 100 users in the fog network can reach the 4.8ms. Additionally, the RSA and ECC are obtained with a comparable user of 100 in 14.8 and 26 milliseconds, respectively. From the comparative study, the predicted approach is achieved with a short processing time for 100 users.

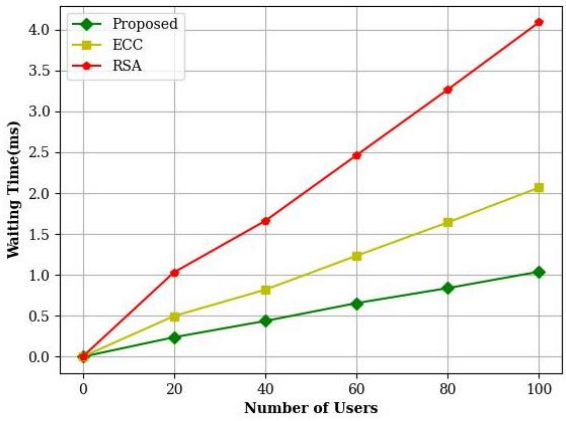

Fig. 7. Waiting time versus users

Figure 7 analyzes and illustrates the waiting time based on the users. The proposed method is contrasted with more established methods like RSA and ECC. The 0.4ms is accomplished at 20 users in the fog network using the projected approach. Additionally, the RSA and ECC are obtained with a comparable user of 20 in 0.5 and 1 milliseconds, respectively. The 20 users in the comparative study show that the anticipated approach achieves a low waiting time. 40 users in the fog network are able to attain the 0.5ms using the projected approach. Additionally, the RSA and ECC are obtained with a comparable user of 40 in 0.8 and 1.6 milliseconds, respectively. The 40 users in the comparative investigation show that the anticipated approach achieves a low waiting time. The 0.6ms is accomplished at 60 users in the fog network using the projected approach. Additionally, the RSA and ECC are obtained with a comparable user of 60 in 1.5 and 2.5 milliseconds, respectively. According to the comparative study, the predicted approach achieves a low waiting time for the 60 users. The 0.8ms is accomplished at 80 users in the fog network using the predicted approach. Additionally, the RSA and ECC obtain equivalent results with an 80 user in 1.6 and 3.2 milliseconds, respectively. The 80 users in the comparative study show that the proposed approach achieves minimal waiting times. With the planned approach, 100 users in the fog network may accomplish 1.0 ms. Additionally, the RSA and ECC obtained equivalent

user of 100 in 2.1ms and 4.1ms, respectively. From the comparative study, it can be shown that the planned approach achieves a low waiting time for 100 users.

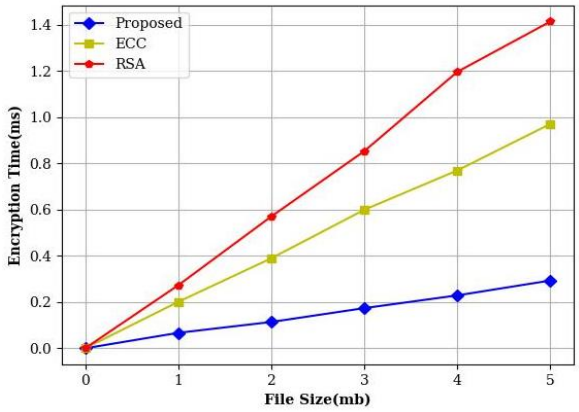

Fig. 8. Encryption time versus file size

Figure 8 analyzes and exemplifies the encryption time based on file size. The proposed method is contrasted with more established methods like RSA and ECC. The 0.1ms is attained at 1MB file size in the fog network using the projected approach. Additionally, the RSA and ECC attain equal 1MBfile sizes in 0.2ms and 0.28ms, respectively. According to the comparative study, the proposed approach is able to achieve minimal encryption time for files up to 1MB in size. In the fog network, the proposed approach achieves a 0.28ms response time at a 2MB file size. Additionally, the RSA and ECC reach equivalent 2MB file sizes in 0.32ms and 0.58ms, respectively. According to the comparative research, the proposed approach is able to achieve minimal encryption time for files up to 2MB in size. In the envisioned approach, the fog network achieves 0.18ms at a 3MB file size. Additionally, the RSA and ECC reach equivalent 3MB file sizes in 0.6ms and 0.8ms, respectively. According to the comparative research, the proposed approach is able to achieve minimal encryption time for files up to 3MB in size. In the planned approach, the fog network's 0.21ms at 4MB file size is accomplished. Additionally, the RSA and ECC reach equivalent 4MB file sizes in 0.7ms and 1.2ms, respectively. According to the comparative research, the proposed approach is able to achieve minimal encryption time for files up to 4MB in size. In the planned approach, the fog network's 0.3ms is reached at a 5MB file size. Additionally, the RSA and ECC reach the same 5MB file size in 1.0ms and 1.4ms, respectively. According to the comparative research, the proposed approach is able to achieve minimal encryption time for files up to 5MB in size.

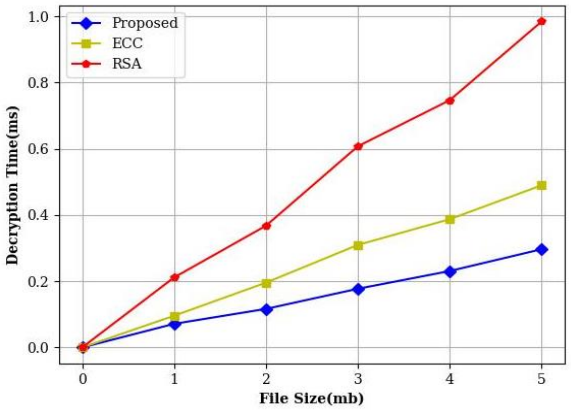

Fig. 9. Decryption time versus file size

Figure 9 analyzes and illustrates the decryption time based on file size. The proposed method is contrasted with more established methods like RSA and ECC. The 0.1ms is attained at 1MB file size in the fog network using the projected approach. Additionally, RSA and ECC reach equivalent 1MB file sizes in 0.18 and 0.21 milliseconds, respectively. According to the comparative investigation, the predicted approach achieves a quick decryption time for files up to 1MB in size. In the fog network, the anticipated approach achieves a 0.19ms response time at a 2MB file size. Additionally, the RSA and ECC reach equivalent 2MB file sizes in 0.2ms and 0.38ms, respectively. According to the comparative investigation, the proposed approach achieves a quick decryption time for files up to 2MB. In the fog network, the planned approach achieves the 0.19ms at 3MB file size. Additionally, the RSA and ECC reach equivalent 3MB file sizes in 0.3ms and 0.6ms, respectively. According to the comparative investigation, the proposed approach achieves a quick decryption time for files up to 3MB. In the planned approach, the fog network's 0.22ms at 4MB file size is accomplished. In addition, RSA and ECC reach equivalent 4MB file sizes in 0.4 and 0.7 milliseconds, respectively. According to the comparative investigation, the proposed approach achieves a quick decryption time for files up to 4MB. In the fog network, the proposed approach achieves a 0.28ms response time at a 5MB file size. Additionally, the RSA and ECC reach the same 5MB file size in 0.5ms and 0.95ms, respectively. According to the comparative investigation, the proposed approach achieves a quick decryption time for files up to 5MB.

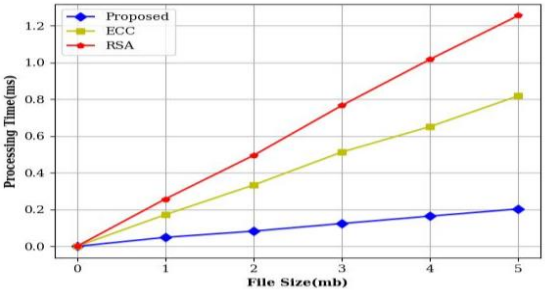

Fig. 10. Processing time versus file size

Figure 10 analyses and illustrates the processing time based on file size. The proposed method is contrasted with more established methods like RSA and ECC. In the fog network, the proposed approach achieves a 0.12ms response time at 1MB file size. Additionally, RSA and ECC reach equivalent 1MB file sizes in 0.18 and 0.22 milliseconds, respectively. According to the comparative study, the proposed approach is able to process files up to 1MB in a short amount of time. In the fog network, the proposed approach achieves a 0.15ms response time at a 2MB file size. Additionally, the RSA and ECC reach equivalent 2MB file sizes in 0.3ms and 0.5ms, respectively. According to the comparative investigation, the proposed approach achieves a low processing time with a 2MB file size. In the fog network, the planned approach achieves the 0.19ms at 3MB file size. Additionally, the RSA and ECC reach equivalent 3MB file sizes in 0.5ms and 0.7ms, respectively. According to the comparative research, the anticipated approach can handle files up to 3MB in size. At 4MBfile size, the anticipated approach achieves 0.19ms in the fog network. Additionally, RSA and ECC reach equivalent 4MB file sizes in 0.62ms and 1.02ms, respectively. According to the comparative research, the anticipated approach can handle files up to 4MB in size. In the proposed approach, the 5MB file size in the fog network allows for a 0.2ms response time. Additionally, the RSA and ECC reach the same 5MB file size in 0.8ms and 1.3ms, respectively. According to the comparative investigation, the anticipated approach can handle files up to 5MB in size.

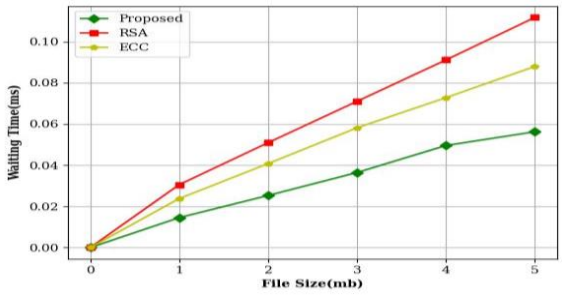

Fig. 11. Waiting time versus file size

Figure 11 analyzes and illustrates the waiting time based on file size. The proposed method is contrasted with more established methods like RSA and ECC. In the fog network, the predicted approach achieves 0.01ms at 1MB file size. Additionally, the RSA and ECC attain equivalent 1MBfile sizes in 0.021 and 0.022 milliseconds, respectively. According to the comparative investigation, the planned approach achieves a low waiting time for files up to 1MB in size. In the planned approach, the 2MB file size in the fog network is reached in 0.025ms. Additionally, the RSA and ECC are successful in achieving a comparable 2MB file size in 0.041ms and 0.048ms, respectively. According to the comparative investigation, the planned approach achieves a short waiting time with a 2MB file size. In the predicted approach, the 3MB file size in the fog network results in 0.038ms. Additionally, the RSA and ECC reach equivalent 3MB file sizes in 0.05ms and 0.068ms, respectively. The 3MB file size and anticipated methodology are reached by comparative analysis. In the proposed approach, the 4MBfile size in the fog network allows the 0.05ms to be attained. Additionally, the RSA and ECC reach equivalent 4MB file sizes in 0.07ms and 0.085ms, respectively. The 4MB file size and anticipated methodology are reached via comparative analysis. In the planned approach, the 5MB file size in the fog network allows for 0.05ms to be realized. Additionally, the RSA and ECC are successfully completed with a comparable file size of 5MB in 0.07ms and 0.11ms, respectively. The 5MB file size and anticipated methodology are reached via comparative analysis.

5. Conclusion

In this research, a unique, lightweight encryption method for Internet of Things (IoT) based on fog is developed. For the execution and interaction of IoT applications and compute instances, FogBus provides a platform-independent interface. It enables consumers run many apps simultaneously, service providers manage their resources, and developers create applications. FogBus also uses encryption, authentication, and Blockchain to safeguard activities on sensitive data. For enhancing security, the blockchain with TLSE has been deployed. The TLSE chooses the best key while taking AOA into account. The UCI machine library is used to gather health care data in order to verify the proposed approach. The suggested method is put into practice using Python, and it is contrasted with more established methods like ECC and RSA, respectively. Based on the study, it can be concluded that the proposed approach produces effective results in terms of processing time, waiting time, encryption time, and decryption time.

References

[1]. Kalaria, Rudri, A. S. M. Kayes, Wenny Rahayu, and Eric Pardede. "A Secure Mutual authentication approach to fog computing environment." Computers & Security 111 (2021): 102483.

[2]. Alwakeel, Ahmed M. "An overview of fog computing and edge computing security and privacy issues." Sensors 21, no. 24 (2021): 8226.

[3]. Singh, Sunakshi, and Vijay Kumar Chaurasiya. "Mutual authentication scheme of IoT devices in fog computing environment." Cluster Computing 24, no. 3 (2021): 1643-1657.

[4]. Yang, Xiaodong, Wanting Xi, Aijia Chen, and Caifen Wang. "An environmental monitoring data sharing scheme based on attribute encryption in cloud-fog computing." PloS one 16, no. 9 (2021): e0258062.

[5]. Amanlou, Sanaz, Mohammad Kamrul Hasan, and Khairul Azmi Abu Bakar. "Lightweight and secure authentication scheme for IoT network based on publish–subscribe fog computing model." Computer Networks 199 (2021): 108465.

[6]. Liu, Yanhui, Jianbiao Zhang, and Jing Zhan. "Privacy protection for fog computing and the internet of things data based on blockchain." Cluster Computing 24, no. 2 (2021): 1331-1345.
[7]. Meng, Fei, Leixiao Cheng, and Mingqiang Wang. "ABDKS: attribute-based encryption with dynamic keyword search in fog computing." Frontiers of Computer Science 15, no. 5 (2021): 1-9.
[8]. Shynu, P. G., Varun G. Menon, R. Lakshmana Kumar, Seifedine Kadry, and Yunyoung Nam. "Blockchain-based secure healthcare application for diabetic-cardio disease prediction in fog computing." IEEE Access 9 (2021): 45706-45720.
[9]. Ngabo, Desire, Dong Wang, Celestine Iwendi, Joseph Henry Anajemba, Lukman Adewale Ajao, and Cresantus Biamba. "Blockchain-based security mechanism for the medical data at fog computing architecture of internet of things." Electronics 10, no. 17 (2021): 2110.
[10]. Kumar, Prabhat, Randhir Kumar, Govind P. Gupta, and Rakesh Tripathi. "A Distributed framework for detecting DDoS attacks in smart contract-based Blockchain-IoT Systems by leveraging Fog computing." Transactions on Emerging Telecommunications Technologies 32, no. 6 (2021): e4112.
[11]. Liu, Yanhui, Jianbiao Zhang, and Jing Zhan. "Privacy protection for fog computing and the internet of things data based on blockchain." Cluster Computing 24, no. 2 (2021): 1331-1345.
[12]. Dewanta, Favian, and Masahiro Mambo. "Bpt scheme: establishing trusted vehicular fog computing service for rural area based on blockchain approach." IEEE Transactions on Vehicular Technology 70, no. 2 (2021): 1752-1769.
[13]. Alzoubi, Yehia Ibrahim, Ahmad Al-Ahmad, and Hasan Kahtan. "Blockchain technology as a Fog computing security and privacy solution: An overview." Computer Communications 182 (2022): 129-152.
[14]. Eddine, Merzougui Salah, Mohamed Amine Ferrag, Othmane Friha, and Leandros Maglaras. "EASBF: An efficient authentication scheme over blockchain for fog computing-enabled internet of vehicles." Journal of Information Security and Applications 59 (2021): 102802.
[15]. Kong, Ming, Junhui Zhao, Xiaoke Sun, and Yiwen Nie. "Secure and efficient computing resource management in blockchain-based vehicular fog computing." China Communications 18, no. 4 (2021): 115-125.
[16]. Singh, Parminder, Anand Nayyar, Avinash Kaur, and Uttam Ghosh. "Blockchain and fog based architecture for internet of everything in smart cities." Future Internet 12, no. 4 (2020): 61.
[17]. Shukla, Saurabh, Subhasis Thakur, Shahid Hussain, John G. Breslin, and Syed Muslim Jameel. "Identification and authentication in healthcare internet-of-things using integrated fog computing based blockchain model." Internet of Things 15 (2021): 100422.
[18]. Amanlou, Sanaz, Mohammad Kamrul Hasan, and Khairul Azmi Abu Bakar. "Lightweight and secure authentication scheme for IoT network based on publish–subscribe fog computing model." Computer Networks 199 (2021): 108465.
[19]. Gupta, Sejal, Ritu Garg, Nitin Gupta, Waleed S. Alnumay, Uttam Ghosh, and Pradip Kumar Sharma. "Energy-efficient dynamic homomorphic security scheme for fog computing in IoT networks." Journal of Information Security and Applications 58 (2021): 102768.
[20]. Whaiduzzaman, Md, Md Julkar Nayeen Mahi, Alistair Barros, Md Ibrahim Khalil, Colin Fidge, and Rajkumar Buyya. "BFIM: Performance measurement of a blockchain based hierarchical tree layered fog-IoT microservice architecture." IEEE Access 9 (2021): 106655-106674.
[21]. Alzoubi, Yehia Ibrahim, Ahmad Al-Ahmad, and Hasan Kahtan. "Blockchain technology as a Fog computing security and privacy solution: An overview." Computer Communications 182 (2022): 129-152.
[22]. Alam, Tanweer. "IoT-Fog: A communication framework using blockchain in the internet of things." arXiv preprint arXiv:1904.00226 (2019).
[23]. Rajesh, Sreeja, Varghese Paul, Varun G. Menon, and Mohammad R. Khosravi. "A secure and efficient lightweight symmetric encryption scheme for transfer of text files between embedded IoT devices." Symmetry 11, no. 2 (2019): 293.
[24]. Abualigah, Laith, Dalia Yousri, Mohamed Abd Elaziz, Ahmed A. Ewees, Mohammed AA Al-Qaness, and Amir H. Gandomi. "Aquila optimizer: a novel meta-heuristic optimization algorithm." Computers & Industrial Engineering 157 (2021): 107250.
[25]. Zhang, Yu-Jun, Yu-Xin Yan, Juan Zhao, and Zheng-Ming Gao. "AOAAO: The hybrid algorithm of arithmetic optimization algorithm with aquila optimizer." IEEE Access 10 (2022): 10907-10933.Mahajan, Shubham, Laith Abualigah, Amit Kant Pandit, and Maryam Altalhi. "Hybrid Aquila optimizer with arithmetic optimization algorithm for global optimization tasks." Soft Computing 26, no. 10 (2022): 4863-4881. https://archive.ics.uci.edu/ml/index.php