symmetry

MDPI

Article

Effective Identification and Authentication of Healthcare IoT Using Fog Computing with Hybrid Cryptographic Algorithm

P. Britto Corthis [1], G. P. Ramesh [1,*], Miguel García-Torres [2,*] and Roberto Ruíz [3]

[1] Department of Electronics and Communication Engineering, St. Peter's Institute of Higher Education and Research, Chennai 600077, India; brittocorthis.ece@spiher.ac.in
[2] Data Science and Big Data Lab, Pablo de Olavide University, 41013 Seville, Spain
[3] Data Analytics Science & Engineering, Pablo de Olavide University, 41013 Seville, Spain; robertoruiz@upo.es
* Correspondence: rameshgp.ece@spiher.ac.in (G.P.R.); mgarciat@upo.es (M.G.-T.)

Abstract: Currently, Internet of Things (IoT)-based cloud systems face several problems such as privacy leakage, failure in centralized operation, managing IoT devices, and malicious attacks. The data transmission between the cloud and healthcare IoT needs trust and secure transmission of Electronic Health Records (EHRs). IoT-enabled healthcare equipment is seen in hospitals that have been implementing the technology for many years. Nonetheless, medical agencies fail to consider the security risk associated with healthcare IoT devices, which are readily compromised and cause potential threats to authentication and encryption procedures. Existing cloud computing methods like homomorphic encryption and the elliptic curve cryptography are unable to meet the security, identity, authentication, and security needs of healthcare IoT devices. The majority of conventional healthcare IoT algorithms lack secure data transmission. Therefore, fog computing is introduced to overcome the problems of IoT device verification, authentication, and identification for scalable and secure transmission of data. In this research manuscript, fog computing includes a hybrid mathematical model: Elliptic Curve Cryptography (ECC) and Proxy Re-encryption (PR) with Enhanced Salp Swarm Algorithm (ESSA) for IoT device verification, identification, and authentication of EHRs. ESSA is incorporated into the PR algorithm to determine the optimal key size and parameters of the PR algorithm. Specifically, in the ESSA, a Whale Optimization Algorithm (WOA) is integrated with the conventional Salp Swarm Algorithm (SSA) to enhance its global and local search processes. The primary objective of the proposed mathematical model is to further secure data sharing in the real time services. The extensive experimental analysis shows that the proposed model approximately reduced 60 Milliseconds (ms) to 18 milliseconds of processing time and improved 25% to 3% of reliability, compared to the traditional cryptographic algorithms. Additionally, the proposed model obtains a communication cost of 4260 bits with a memory usage of 680 bytes in the context of security analysis.

Keywords: elliptic curve cryptography; healthcare management system; internet of things; proxy re-encryption; salp swarm algorithm

Citation: Corthis, P.B.; Ramesh, G.P.; García-Torres, M.; Ruíz, R. Effective Identification and Authentication of Healthcare IoT Using Fog Computing with Hybrid Cryptographic Algorithm. *Symmetry* **2024**, *16*, 726. https://doi.org/10.3390/sym16060726

Academic Editor: Michel Planat

Received: 13 March 2024
Revised: 7 May 2024
Accepted: 15 May 2024
Published: 11 June 2024

1. Introduction

In recent decades, IoT-based cloud systems have gained more attention among researchers, because of their higher flexibility and scalability [1]. However, they faces several problems such as the handling of enormous IoT devices, privacy leakage, malicious attacks, and failure in single point centralized operation. Specifically, the IoT has inherent problems for scalability and capacity, especially in healthcare [2,3]. In a real-time scenario, IoT devices create a huge amount of healthcare data. In IoT networks, the heterogeneity and homogeneity of the IoT devices has privacy and security problems, where the identification of IoT devices is indirectly linked to the security and protection of healthcare data [4,5]. For instance, intruders or attackers easily impersonate sensors and hack the IoT devices, which results in faulty values. The motivation of the present study is to design an authenticable,

reliable, and secure IoT-based cloud system for sensitive healthcare data [6]. The combination of IoT and fog computing currently works as a centralized server and resolves many problems like packet errors, unsafe codes in smart contracts, malicious behavior of nodes, etc. [7,8].

Cloud servers are utilized to analyze, process, and store enormous amounts of healthcare data in IoT systems, which are generated from IoT devices [9]. Three major concerns that need to be addressed in cloud servers and healthcare IoT for the secure transmission of data remain, such as: (i) effective key verification in the distributed environments, (ii) the authentication of EHRs, and (iii) the secure identification of IoT devices. Error and data loss are intolerant in the IoT systems, and healthcare data is more sensitive, needing to be updated every second [10,11]. Several analytical models, protocols, and algorithms are designed for healthcare IoT in order to overcome the aforementioned problems. For the secure transmission of EHRs, the majority of existing studies focused on complex algorithms and heavy communication protocols related to memory and computation requirements [12,13]. Due to the centralization of cloud servers, most of the existing studies face a problem of single point failure [14,15]. In this research manuscript, an intelligent fog computing-based model is implemented for secure transmission of EHRs. The proposed model has a hybrid mathematical model (ECC and PR with ESSA) in order to resolve the above-discussed problem. The contributions are presented below:

- The design of an effective fog computing based on hybrid mathematical model for the reliable and secure sharing of healthcare data between doctors, IoT devices, patients, and fog nodes.
- A proposal for decentralized fog computing with a hybrid mathematical model (ECC and PR with ESSA) for reliable data transaction and transmission. The proposed model effectively authenticates, identifies, and verifies the EHRs.
- The hybrid mathematical model (ECC and PR with ESSA) performs effective target and data source verification, and transmitted healthcare data authentication utilizing dissimilar IoT devices and fog nodes. The generated medical data is encrypted and decrypted using a hybrid cryptographic algorithm.
- The proposed ESSA determines the optimal key size and parameters of the PR algorithm. This process ensures that the generated keys are the appropriate length for security that reduces the computational overhead and overall resource consumption. The proposed algorithm's efficacy is analyzed using dissimilar performance measures such as energy consumption, throughput, response time, execution time, processing time, packet error, reliability, and verification time.

This manuscript is designed in this manner: research articles related to the topic "healthcare management system" are surveyed in Section 2. The details about methodology, security analysis, and the conclusion of the proposed model are described in Sections 3–5.

2. Literature Survey

Tuli et al. [16] introduced a novel healthcare system (Health–Fog) that integrates ensemble deep learning models with the edge computing devices. In this study of the literature, the developed system was implemented in the real time applications, particularly for heart disease analysis. The fog-bus was utilized in the fog-enabled cloud system for deploying and testing the effectiveness of the developed system by means of power consumption, jitter, execution time, accuracy, network bandwidth, and latency. Here, the experimental analysis was done for heart disease data for predicting whether the patients have a heart problem or not. Das and Namasudra [17] have integrated Serpent, Advanced Encryption Standard (AES) and ECC algorithms for securing sensitive healthcare data in the IoT systems. The developed system integrates asymmetric and symmetric based encryption algorithms that superiorly enhances the security of healthcare data and integrity of data. The performance comparisons and security evaluations were presented in this study to validate the efficacy of the developed system. The discussion and the obtained results demonstrate the efficacy of the developed system.

Geetha et al. [18] introduced a novel Secure Medical Image Management (SMIM) system based on the Pigeon Inspired Optimization (PIO) algorithm. Initially, the medical images were transformed into twelve shares by implementing Secret Share Creation (SSC) method, and further, the encryption process was accomplished utilizing the ECC algorithm. In this study, the PIO algorithm was employed for optimizing the key generation process in the ECC algorithm that aims to increase medical data security. In this research study, a widespread experimental analysis was done in order to demonstrate the effectiveness of the presented system, and the results state that the presented system obtained a better Peak Signal to Noise Ratio (PSNR) than the existing systems. Park et al. [19] have integrated a PR algorithm with blockchain for the effective management of EHRs. In this study, the proxy server initially re-encrypts the ciphertext between the servers in order to resolve data sharing problems. Generally, the outsourcing companies cannot access the EHRs or servers, because the servers were separated from the blockchain systems. Here, blockchain technology helps data owners to access and control EHRs by utilizing smart contracts, which enables the effective and secure sharing of EHRs.

Sutradhar et al. [20] integrated a fruit-fly optimization algorithm with a dynamic encryption algorithm for the effective encryption of healthcare data. Here, the evaluation measures—such as storage footprint, power consumption, communication overhead, and lifetime—demonstrate the efficacy of the developed system, and the achieved results were better than the comparative models. In addition, Ali et al. [21] integrated a homomorphic encryption algorithm with blockchain for securing healthcare data. The presented model frequently updates the policies and performs a secure key revocation process, which enables secure access to the patient's healthcare data. The integration of blockchain with the presented model resolves the security issues and improves the efficiency of digital healthcare data sharing. The extensive numerical analysis states that the presented model provides secure and transparent data sharing with cost-effectiveness.

Verma et al. [22] implemented a novel healthcare monitoring system that integrates automated monitoring with deep learning models. The developed healthcare monitoring system utilizes fog-bus in order to demonstrate its effectiveness by means of power consumption, network bandwidth, process execution time, jitter, accuracy, and latency. Ahmad et al. [23] developed a new IoT based health monitoring system using machine learning techniques. Initially, a rectangular window was applied for categorizing the time signals, and further, feature vectors were extracted from the categorized signals by implementing Mel Frequency Cepstral Coefficients (MFCC). By using the extracted features, the Support Vector Machine (SVM) classifies the signals into faulty or normal.

Sundas et al. [24] presented a smart healthcare system—HealthGuard—which integrates four models: random forest, decision tree, k-nearest neighbor, and artificial neural network. Here, around eight medical devices were utilized for training the healthcare system. The numerical analysis demonstrates the efficacy of the presented system and its ability against harmful attacks. Qureshi et al. [25] presented a mobile health system based on machine learning and statistical techniques. In this study, the presented system classifies cardiovascular diseases based on their seriousness. The efficacy of the presented mobile health system was validated by means of specificity, sensitivity, and accuracy, and the achieved results were superior to those of the existing systems.

Shahnawaz Ahmad, Shabana Mehfuz, and Javed Beg [26] proposed the Hybrid Cryptographic Approach (HCA), which was introduced to improve the Key Management System (KMS) in cloud environments. A combination of AES and ECC cryptography was utilized to encrypt and decrypt the data, while an asymmetric ECC approach was used to generate the key. It was discovered that the hybrid ECC-AES model required less time than the AES model and other versions that were already in use. The suggested technique is more dependable than AES and has been able to resolve the key exchange issue. HCA-KMS was therefore created to offer strong security standards for medical data. Data security cannot be achieved by using encryption and key management techniques only as a preventative measure against specific threats.

Kondaka et al. [27] implemented a healthcare system based on machine learning models with an intelligent cloud system. The efficacy of the developed healthcare system was validated by means of communication efficiency, accuracy, and storage capacity. The extensive empirical analysis states that the developed healthcare system delivers a drastic improvement in the healthcare parameters. Shukla et al. [28] presented an Advanced Signature-Based Encryption (ASE) algorithm for health data authentication and IoT device verification and identification. The primary aim of this study was to secure patient's health data in the real time service. The presented ASE algorithm provides secure services for transmission and transaction near the edge. The extensive simulation analysis showed that this AES algorithm achieved better throughput and reliability. The literature survey of the existing studies is given in Table 1.

Table 1. Literature survey of the existing studies.

Author	Methodology	Drawbacks
Tuli et al. [16]	Ensemble deep learning models	The ensemble of different deep learning models increases the time complexity of the developed healthcare system.
Das and Namasudra [17]	Serpent, AES, and ECC	The integration of three encryption algorithms increases the system complexity and execution time.
Geetha et al. [18]	SSC, ECC, and PIO	The presented system's performance was further improved by integrating the SMIM system with the blockchain technology, especially in the healthcare sector.
Park et al. [19]	PR algorithm with blockchain	The security analysis demonstrates the efficacy of the developed system, but the PR algorithm introduces additional computational overhead.
Sutradhar et al. [20]	Fruit-fly optimization algorithm with dynamic encryption algorithm	Key management was a major challenge in this study.
Ali et al. [21]	Homomorphic encryption algorithm with blockchain	However, the compatibility and reliability of the presented healthcare system needs to be further improved in the real-world settings.
Verma et al. [22]	Deep learning models	Generally, deep learning models need an enormous amount of labeled data, but it was challenging to obtain labeled data in the healthcare system.
Ahmad et al. [23]	MFCC and SVM	The inclusion of traditional machine learning models in the healthcare system faces concerns like outliers and overfitting.
Sundas et al. [24]	HealthGuard, integrates four models: random forest, decision tree, k-nearest neighbor, and artificial neural network.	
Qureshi et al. [25]	SVM and decision tree	
Shahnawaz Ahmad, Shabana Mehfuz, and Javed Beg [26]	HCA-KMS	However, data security cannot be achieved by using encryption and key management techniques only as a preventative measure against specific threats.
Kondaka et al. [27]	iCloud assisted intensive deep learning model	The presented model has high processing time, particularly in the context of healthcare monitoring paradigm
Shukla et al. [28]	ASE algorithm	ASE algorithm has complex mathematical operations that result in high computational overhead.

According to the comprehensive analysis, secure communication through distributed fog computing and decentralized technology is necessary for the transmission of healthcare IoT. IoT devices in the healthcare sector are resource-constrained and perform inadequately as their number increases. As a result, the fog nodes receive the healthcare data, and the IoT network is split up into multiple nodes. These fog nodes are dispersed among multiple clusters. The storage of IoT data related to healthcare was also unattainable, so this research turned to fog nodes, which offer a secure platform with extra cryptographic security features. Furthermore, previous efforts to ensure secure data transmission in the context of the IoT in healthcare exclusively addressed complicated communication protocols and algorithms associated with computation and memory needs; therefore, they encountered a single point of failure because of a centralized cloud server. As a result, healthcare identification and authentication continue to be a difficult issue that has not received enough attention. In order to overcome the above-stated problems, a novel healthcare management system based on a hybrid cryptographic algorithm (ECC and PR with ESSA) is introduced in this research manuscript. The ESSA determines the optimal key size and PR algorithm parameters, which ensures that the generated keys have suitable length for security to reduces the computational overhead and overall resource consumption in the healthcare data.

3. Methods

In this manuscript, the proposed fog computing system comprises healthcare IoT devices, smart contracts, fog computing storage, and IoT networks. Here, the healthcare data are stored in the master fog servers and fog nodes instead of storing in cloud servers and blockchain [29–31]. In this scenario, the fog-storage effectively groups the EHRs, which are created from the devices. It is necessary to prove that the healthcare data and IoT devices are authenticated with key exchange and device identification. The research steps involved in this work are depicted in Figure 1.

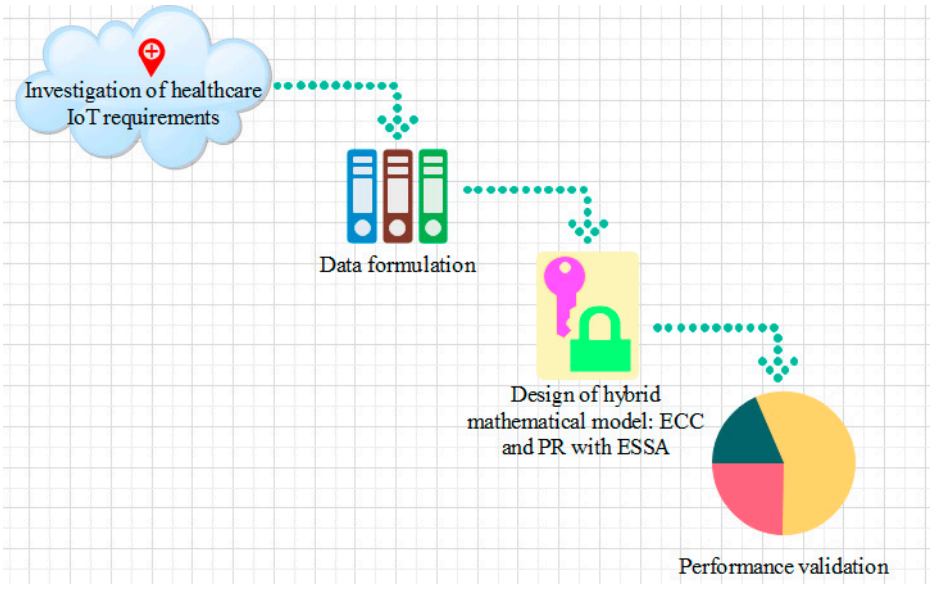

Figure 1. Research steps involved in this work.

3.1. System Model

The designed system model includes three outlines, such as key components, interactions and processes, and security measures. The important key components in this system are fog nodes, healthcare IoT devices, cloud infrastructure, fog-computing infrastructure, an authentication and identification module, and a hybrid cryptographic algorithm. Next, the interactions and processes phase comprises following steps: (i) registration (IoT devices are registered with this system); (ii) authentication (used hybrid cryptographic algorithm

(ECC and PR with ESSA) for securing device authentication); (iii) access control (access control policies are enforced for restricting unauthorized access); (iv) data encryption (to ensure the integrity and confidentiality of sensitive medical data); (v) fog computing (to improve system efficiency and to enable real time analytics and response); and (vi) secure communication (to enable secure communication among cloud, fog nodes and IoT devices using the hybrid cryptographic algorithm). Finally, the security measures phase includes three important processes such as key management, the monitoring of system activities, and regular authentication checks. A pictorial presentation of the system model is given in Figure 2.

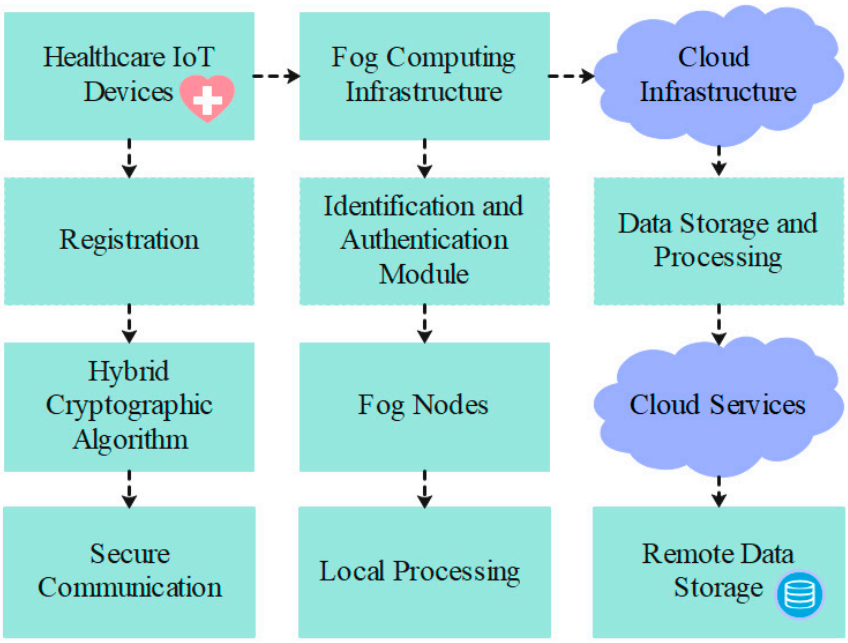

Figure 2. Pictorial presentation of the system model.

Testing the identification and authentication of IoT devices to monitor electronic health records (EHR) involves ECC security measures to ensure the integrity and confidentiality of patient data. This process contains registering each IoT device within the network, capturing unique identifiers and relevant metadata. Strong authentication mechanisms, including multi-factor authentication and role-based access control, are then employed to verify the identity of devices and restrict access to authorized users only. Secure communication protocols, specifically HTTPS, are implemented to encrypt data transmission and prevent unauthorized access. Penetration testing, regular audits, and security assessments are conducted to identify vulnerabilities. Continuous monitoring tools are utilized to track device activities and detect suspicious behavior, while updates and patches are regularly applied to mitigate emerging threats. The primary purpose of this approach is to safeguard electronic health records, prevent unauthorized access, and maintain compliance with user requirements ensuring the security and privacy of patient information throughout the healthcare ecosystem.

3.2. Elliptic Curve Cryptography Algorithm

The ECC is utilized for performing many security functions such as digital signatures, authentication, encryption, etc. [32,33]. A finite elliptic curve in a prime field is defined in Equation (1).

$$E(p,a,b) = \left\{ (x,y) \middle| x,y \in Z_p, y^2 = x^3 + ax + b(modp) \right\} \cup \{'0\} \quad (1)$$

where $Z_p = \{0, 1, \ldots., p-1\}$, p is represented as a prime number, and $a \wedge b \in Z_p$ should satisfy the criteria presented in Equation (2).

$$4a^3 + 27b^2 \neq 0(modp) \tag{2}$$

where the point at infinity is represented as ′0 and it acts as an identity element [34,35]. The inverse of a point $P = (x, y) \in E(p, a, b)$ is specified in Equation (3).

$$-P = \begin{Bmatrix} '0 & P = '0 \\ (x, p-y) & otherwise \end{Bmatrix} \tag{3}$$

In every pair of points $P, Q \in E(p, a, b)$, the sum point $R = P + Q$ is determined as the inverse of 3rd intersection line $P'0$ with curve E. If $P = -Q$, the additional results are determined ′0, as shown in Equation (4).

$$P + Q = \begin{Bmatrix} P, & Q = '0 \\ Q, & P = '0 \\ '0 & P = -Q \\ R(x_R, y_R), & otherwise \end{Bmatrix} \tag{4}$$

where $x_R = \lambda^2 - x_P - x_Q(modp), \wedge y_R = \lambda(x_P - x_R) - y_p(modp)$. The function λ is determined in Equation (5).

$$\lambda = \begin{Bmatrix} (y_Q - y_P)/(x_Q - x_P), & P \neq Q \\ (3x_P^2 + a)/2y_{P,} & P = Q \end{Bmatrix} \tag{5}$$

In a point $P \in E(p, a, b)$, a scalar point multiplication $k \in Z_p$ is denoted in Equation (6).

$$kP = \begin{Bmatrix} O, & k = 0 \\ P + (k-1)/P, & otherwise \end{Bmatrix} \tag{6}$$

Based on Hasse's theorem, the points on a finite elliptic curve $E(p, a, b)$ are bounded as shown in Equation (7).

$$p + 1 - 2\sqrt{p} \leq E(p, a, b) \leq p + 1 + 2\sqrt{p} \tag{7}$$

In the present scenario, there is no effective algorithm in finding $k \in Z_p$ and the hardness of this concern forms the basis of ECC. The secret k is hidden in the structure of $P = kG$. The visual diagram of elliptic curve is denoted in Figure 3 [36].

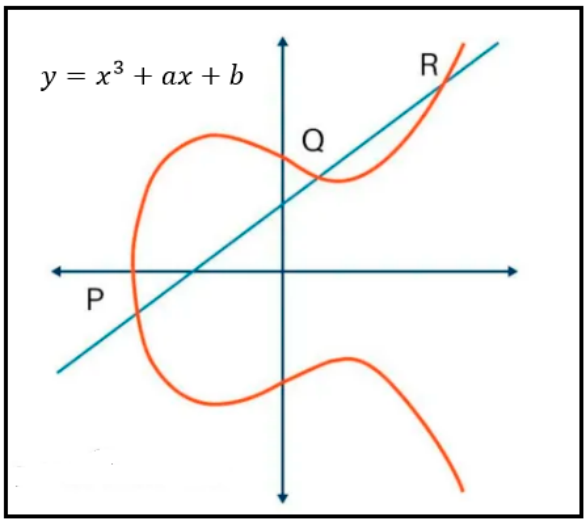

Figure 3. Visual diagram of elliptic curve.

3.3. Proxy Re-Encryption with Enhanced Salp Swarm Algorithm

In this scenario, the PR algorithm converts the encrypted ciphertext of ECC into another encryption using another key. In the initial phase of the PR algorithm, the data owner encrypts the healthcare data utilizing a public key PK_A. In this algorithm, the data owner generates a pre key $RK_{A\to B}$ to the master fog servers with fog nodes. Here, the fog servers transform the ciphertext, which is encrypted using the public key PK_A of the data owner. The respective ciphertext is decrypted by using the recipient's (doctors/patients) secret key SK_B without utilizing the elemental plaintexts SK_A and SK_B [37,38]. The systematic process of the PR algorithm is determined below; here, prime order is denoted as q, multiplicative group of q is stated as G, and the random generator of G is specified as g. The step involved in the proxy re-encryption algorithm is given below.

- **Key generation:** Initially, the data owner selects a random element $g^n \in G$ as a secret key SK_A, n represents private key, $a \in Z_q$ and a public key PK_A. In addition to this, the recipient's (doctors/patients) public key and secret key pair (PK_B, SK_B is $\left(b, g^b\right)$. Here, the pre-key $RK_{A\to B} = b/a(modq)$ is used to transmit the ciphertext, which are encrypted and decrypted by PK_A and SK_B.
- **Encryption:** The sender selects $r \in Z_q$ in order to encrypt the healthcare data, and generates ciphertext $C_A = (C_{A1}, C_{A2}) = (g^r m, g^{ar})$.
- **Decryption:** The data owner decrypts the healthcare data utilizing a secret key by calculating $C_{A1}/(C_{A2})^{1/a}$ based on ciphertext $C_A = (C_{A1}, C_{A2})$.
- **Re-encryption:** The master fog servers convert C_A to C_B based on $RK_{A\to B}$. It is decrypted by the recipients (doctors/patients): $C_{B1} = C_{A1}$ and $C_{B2} = (C_{A2})^{RK_{A\to B}}$. By calculating $C_{B1}/(C_{B2})^{1/b}$, the recipients (doctors/patients) decrypt healthcare data with a secret key b by utilizing the ciphertext (C_{B1}, C_{B2}. In this algorithm, the generated healthcare data is encrypted twice; initially, the healthcare data is encrypted by the data owner using public key, secondly, re-encrypted utilizing pre-key. In this scenario, the PR algorithm is worked on the basis of ElGamal encryption algorithm [39,40]. The schematic diagram of PR algorithm is given in Figure 4.

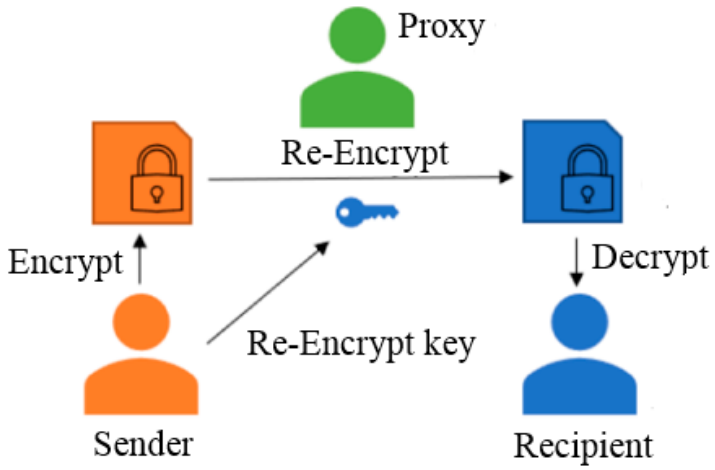

Figure 4. Schematic diagram of PR algorithm.

This study presented a model of secure data transmission by utilizing identification and authentication using ECC and PR. The dynamic topology of the network leads to failure of the links, which changes the key size and similarly offers a chance for the attackers to come in into the network. The identification of such unauthorized users requires secure key exchange process. Therefore, ESSA is used for optimal generation of keys to enhance security. The combination of ECC and PR with ESSA provides security in an efficient manner to authenticate the client entering the network.

In this study, the ESSA is implemented for the optimal generation of keys in the PR algorithm for the scalable transmission of healthcare data. The SSA is one of the effective meta-heuristic optimization algorithms, which mimics salps swarming behavior. Generally, all the salps are connected in the form of chain-like structure that assists salps in predation and movement. In this algorithm, the salps are categorized into two groups such as follower's salps (key length) and leader's salps (key pairs) [41,42]. The leader's salps explores the location, where it considers food source as the objective function and rapidly updates its positions based on Equation (8).

$$X_{1,j}(t+1) = \begin{cases} X_j(t) + c_1\left(\left(ub_j - lb_j\right) \times c_2 + lb_i\right), c_3 \leq 0.5 \\ X_j(t) - c_1\left(\left(ub_j - lb_j\right) \times c_2 + lb_i\right), c_3 > 0.5 \end{cases} \quad (8)$$

where lower bounds are represented as lb_j, upper bounds are denoted as ub_j, and number of iterations is stated as t. In the j^{th} dimension, X_j and $X_{1,j}$ are denoted as the present positions of the food source and salp leader. The independent random numbers are indicated as c_2 and c_3, which usually ranges between zeros to one. The parameter coefficient c_1 is crucial in the conventional SSA for balancing the exploitation and exploration ability, which non-linearly decreases from two to zero, as mentioned in Equation (9).

$$c_1 = 2e^{-\left(\frac{4t}{T}\right)^2} \quad (9)$$

where the maximum number of iterations is indicated as T and the present iteration is stated as t. The key length moves with the key pairs after updating the positions. The key lengths are updated as per the key pairs, which are expressed in Equation (10).

$$X_{i,j}(t+1) = \frac{1}{2}) \quad (10)$$

where $i \geq 2$, $X_{i,j}$ is represented as the i^{th} follower's positions in the j^{th} dimension.

In the ESSA, the WOA is integrated with the conventional SSA for enhancing the global and local search processes. First, the WOA generates a random population for a predefined number of search agents. WOA, a unique intelligent optimization method which draws inspiration from the way whales hunt in groups in the wild. The benefits of this method include its ease of implementation, reduced number of parameters, and basic simplicity.

In the ESSA, the search agents are moved with the best-fit agents. If the present best fit agent is entrapped in local optima, all other search agents fall with it [43,44]. Before performing every operation using WOA, the leader mechanism Equation (9) of conventional SSA is utilized for updating the population position in order to escape from the problem of dilemma. In the ESSA, the non-linear parameter c_1 is used in the bubble-net attacking phase and for optimal key generation, which superiorly balances the key length and key pairs. The parameters considered in the ESSA are determined as follows: lower bound is 0.3, upper bound is 0.9, number of iterations is 100, and population size is 100. A numerical examination of the proposed hybrid mathematical models (ECC and PR with ESSA) is presented in Section 4. The flowchart of the proposed model is shown in Figure 5. The pseudocode of ECC and PR with ESSA model is represented as Algorithm 1.

Algorithm 1: Pseudocode of ECC and PR with ESSA model

Input: Encrypted H, Signers (fog node), and Signed Public key (SK_{Pub})
1: **START**
2: **for** each H_{IoT} device aC_t is issued
3: (Fog computing model is created)
4: Data classification
5: **if** $(PHD = Sensitive_{Data})$ then
6: get geo-location and send the data for verification to f_n
7: **else if** $(HD = non - sensitive)$
8: then
9: H send to f_n to C_s
10: f_n allocates the HD to f_s.
11: **for** each H to $(H_{IoT} \leftarrow C_t)$
12: $C_t + T_S \leftarrow H$
13: **if** $f_s = Available$
14: allocate the HD
15: **else** no allocation
16: **end if**
17: **end**
18: **function** HD Authentication H
19: $HD_{Retrive}(Prvt_k)$
20: **if** $C = C$
21: **then**
22: Mutual Authentication $(H_{IoT_1}, H_{IoT_2}, f_n, f_s, p_{ax}, d_y)$
23: **function** VERIFICATION $\left(C, SK_{pub}\right)$
24: $Hash_C \leftarrow$ calculate hash of the received encrypted H to be verified
25: Using Public Key SK_{Pub} of H_{IoT}, extract $Hash_P$ of H_{IoTHD} file
26: **if** $hash = hash$ then
27: **return** C
28: **else**
29: **return** "Signature incorrect"
30: **end if**
31: **end function**

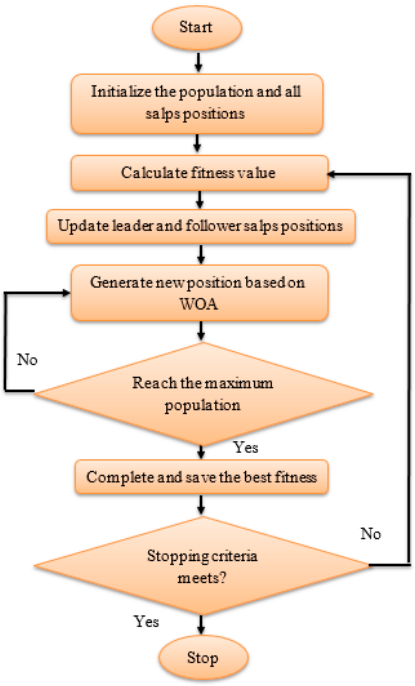

Figure 5. Flowchart of the proposed model.

4. Results and Discussion

In this scenario, the proposed hybrid mathematical model's (ECC and PR with ESSA) execution is analyzed and evaluated. The effectiveness of the fog computing-based model is analyzed by performing different experiments and simulations. The primary objective of this experiment is to identify and authenticate the IoT devices and healthcare data for the scalable transmission of patient EHRs in a fog–IoT environment. Here, the experiment is carried-out for healthcare IoT in a real-time scenario. Furthermore, the python-Spyder editor tool and iFogSim are utilized for simulating the fog computing-based model. the hybrid mathematical model (ECC and PR with ESSA) in particular is implemented by using the iFogSim simulator. The proposed model is implemented utilizing python and NetBeans with many classes, modules, and main packages. The software and hardware specifications used for implementation are detailed in Table 2. The efficacy of the fog computing-based model is analyzed in light of response time, energy consumption, throughput, execution time, processing time, packet error, reliability, and verification time.

Table 2. Software and hardware specifications used for implementation.

Software and Hardware	Specifications
Language	Python
Platform	iFogSim and Spyder
Type of system	64-bit windows operating system
Processor	Intel core i9 12th generation
Random Access Memory (RAM)	128 GB
Computer Processing Unit (CPU)	5.30 GHz

4.1. Quantitative Analysis

Here, the IoT system comprises clients, IoT devices, fog nodes, and sensors. A hybrid cryptographic algorithm is utilized in this fog computing-based model for transmitting the EHRs to fog storage, and the devices permit the doctors and patients in acquiring the patients' health details. The cloud servers verify the IoT devices which raises the overhead associated with infrastructure and maintenance costs. Consequently, the IoT data can be authenticated using the IoT device identity. Identification of IoT devices in healthcare is crucial for access control and authentication. An integrated fog computing-based model operates in a decentralized and distributed setting which can be used to accomplish this. The following subsection shows the analysis of the identification and authentication of Healthcare IoT outcomes.

4.1.1. Analysis of Identification in Healthcare IoT

Here, the Giga Ethernet is used for fog node communication, the maximum CPU power is set as 150 watts, and the idle CPU power is set as 100 watts. The description about fog devices is given in Table 3. In this research manuscript, the performance of the proposed fog computing system is validated for five different configurations. In configuration 1, three EEG IoT devices are utilized for transmitting the data to the fog devices. Configuration 2 contains four EEG IoT devices for transmitting the data to the fog devices 1 and 2. Correspondingly, configurations 3, 4, and 5 have five, six, and seven EEG IoT devices, respectively, for transmitting data to fog devices 1 and 2. Configuration 6 has 100 EEG IoT devices connected to fog device 1 and 2 for scaled data analysis. Additionally, the configuration of the EEG sensor is detailed as follows: length of network in bytes is 24,000 bytes, length of CPU is 1250 million instructions, and the arrival time of data packet is 25 milliseconds (ms). The description of edge module with regard to length of network and length of CPU in displayed in Table 4. Table 4 clearly shows that the Electroencephalogram (EEG) data stream accomplished a network length of 2100 with 1200 Million Instructions Per Second (MIPS), which is better than the other types of tuples. In addition to this, the description of NLs for configurations 1, 2, 3, 4, 5, and 6 are represented in Tables 5–10.

Tables 5–10 show how latency values are calculated for different configurations with various sources.

Table 3. Description of fog devices.

Type of Device	RAM (GB)	CPU (GHz)
Cloud server 1	4	4
Master fog controller	3	3
Fog device 1 (mobile device)	2	2.6
Fog device 2 (mobile device)	2	2.6

Table 4. Description of edge module.

Types of Tuples	Length of Network in Bytes	Length of CPU in Million Instructions Per Second (MIPS)
Sensitive data stream	1700	2800
Health data stream	1700	2200
Raw Electroencephalogram (EEG) data stream	2100	1200

Table 5. Description of Network Links (NLs) for configuration 1.

Source	Destination	Latency in ms
EEG IoT-1	Fog device 1	38
EEG IoT-2	Fog device 1	43
EEG IoT-3	Fog device 1	44
Fog device 1	Cloud server 1	67

Table 6. Description of NLs for configuration 2.

Source	Destination	Latency in ms
EEG IoT-1	Fog device 1	38
EEG IoT-2	Fog device 1	43
EEG IoT-3	Fog device 2	45
EEG IoT-4	Fog device 2	50
Fog device 1	MFC	60
Fog device 2	MFC	65
MFC	Cloud server 1	68

* MFC-Master Fog Controller.

Table 7. Description of NLs for configuration 3.

Source	Destination	Latency in ms
EEG IoT-1	Fog device 1	38
EEG IoT-2	Fog device 1	43
EEG IoT-3	Fog device 1	45
EEG IoT-4	Fog device 2	50
EEG IoT-5	Fog device 2	55
Fog device 1	MFC	60
Fog device 2	MFC	65
MFC	Cloud server 1	68

Table 8. Description of NLs for configuration 4.

Source	Destination	Latency in ms
EEG IoT-1	Fog device 1	38
EEG IoT-2	Fog device 1	43
EEG IoT-3	Fog device 1	45
EEG IoT-4	Fog device 2	50
EEG IoT-5	Fog device 2	55
EEG IoT-6	Fog device 2	55
Fog device 1	MFC	60
Fog device 2	MFC	65
MFC	Cloud server 1	68

Table 9. Description of NLs for configuration 5.

Source	Destination	Latency in ms
EEG IoT-1	Fog device 1	38
EEG IoT-2	Fog device 1	43
EEG IoT-3	Fog device 1	45
EEG IoT-4	Fog device 2	50
EEG IoT-5	Fog device 2	55
EEG IoT-6	Fog device 2	55
EEG IoT-7	Fog device 2	60
Fog device 1	MFC	60
Fog device 2	MFC	65
MFC	Cloud server 1	68

Table 10. Description of NLs for configuration 6.

Source	Destination	Latency in ms
EEG IoT-1 to EEG IoT-25	Fog device 1	38
EEG IoT-26 to EEG IoT-50	Fog device 1	43
EEG IoT-56 to EEG IoT-75	Fog device 2	45
EEG IoT-76 to EEG IoT-100	Fog device 2	50
Fog device 1	MFC	60
Fog device 2	MFC	65
MFC	Cloud server 1	68

Table 5 clearly shows that fog device with cloud server for the configurations 1 achieved latency of 67 ms. Table 6 demonstrates the description of NLs for the configurations 2; where the MFC to cloud server 1 obtains a latency of 68 ms. Following that, Table 7 presents the description of NLs for configuration 3, which also obtains a similar latency of 68 ms for MFC to cloud server 1. Tables 8–10 show the description of NLs for configurations 4, 5, and 6, which attained a latency of 68 ms on source (MFC) to destination (cloud server 1).

4.1.2. Analysis of Authentication in Healthcare IoT

Figures 6–8 show that the response time, energy consumption, and throughput of the fog computing nodes are better compared to the cloud servers. In particular, Figure 6 shows the response time of cloud servers and fog computing for dissimilar configurations in ms. Figure 6 shows that the response time increases in both cloud servers and fog computing, when the number of EEG IoT devices are increased. Here, the response time is the time consumed by the cloud servers and fog nodes for responding to the IoT devices or end-user request. The transmission time is ignored when computing the response time. The response time is the combination of wait time (time taken by the data packets in a queue before serving) and service time.

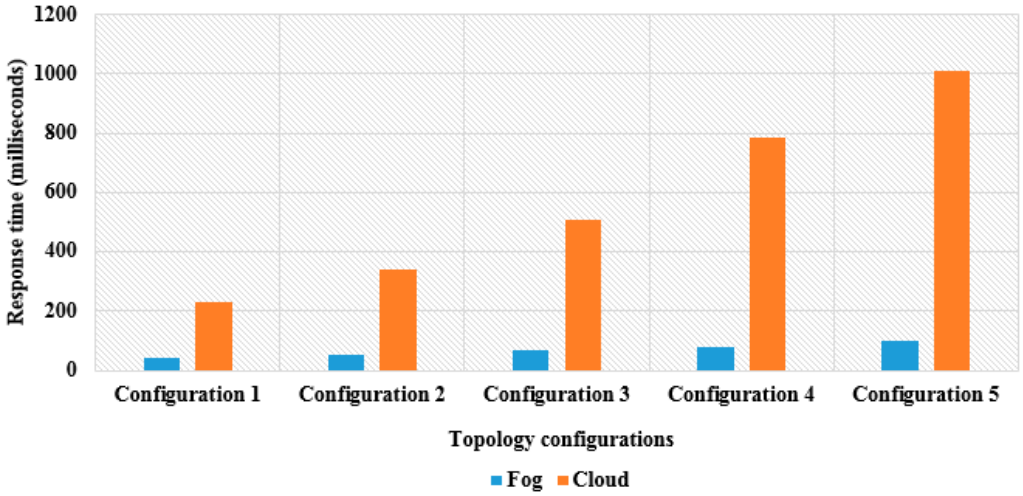

Figure 6. Response time of fog and cloud computing.

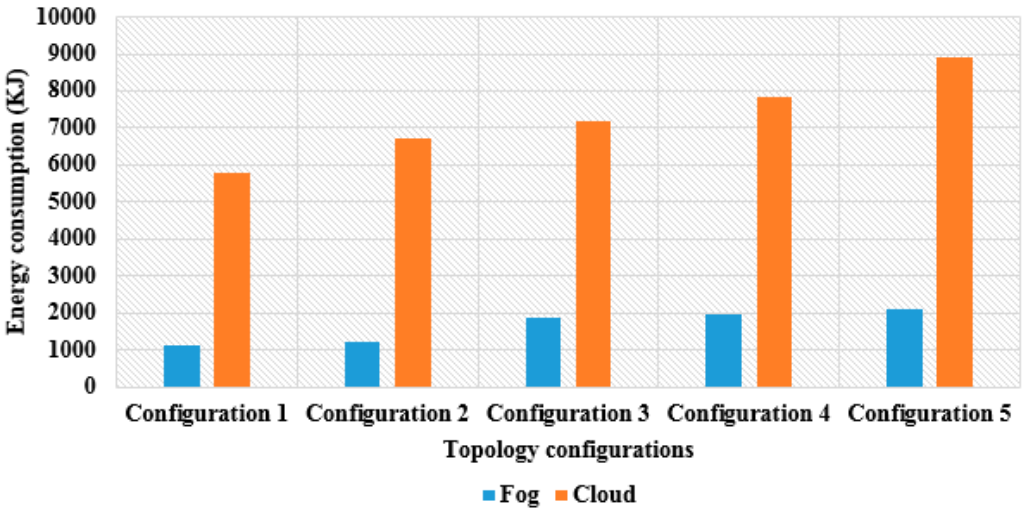

Figure 7. Energy consumption of fog and cloud computing.

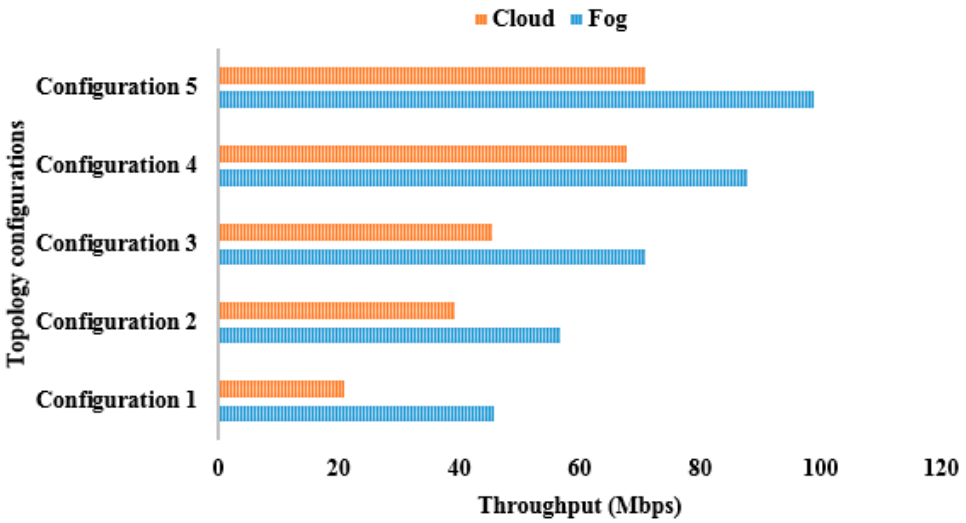

Figure 8. Throughput of fog and cloud computing.

Conversely, Figure 7 shows that the cloud servers consumed more energy than the fog nodes in dissimilar topology configurations. In a cloud environment, the generation of

enormous amounts of healthcare data from IoT devices and excessive processor switching leads to ineffective usage of data packets that results in higher energy consumption. Additionally, throughput is computed by estimating the success rate of data sharing from the fog nodes to the user. As seen in Figure 8, the throughput increases when the EEG IoT devices are increased. The response time, energy consumption, and throughput of cloud computing and fog computing are shown in Figures 6–8.

The execution time of cloud servers and fog nodes for dissimilar topology configurations is represented in Figure 9. Figure 9 shows that the fog node's execution time is less than that of the cloud servers. While increasing the number of devices, the execution time automatically increased in both fog and cloud computing. The devices are increased in the configurations 1, 2, 3, 4, and 5 from 3 to 7, and simulators like iFogSim are utilized for healthcare IoT verification. Five dissimilar configurations are designed in the iFogSim simulation to verify the success rate of data transmission between the doctors and patients after authentication. The execution time of fog and cloud computing is specified in Figure 9.

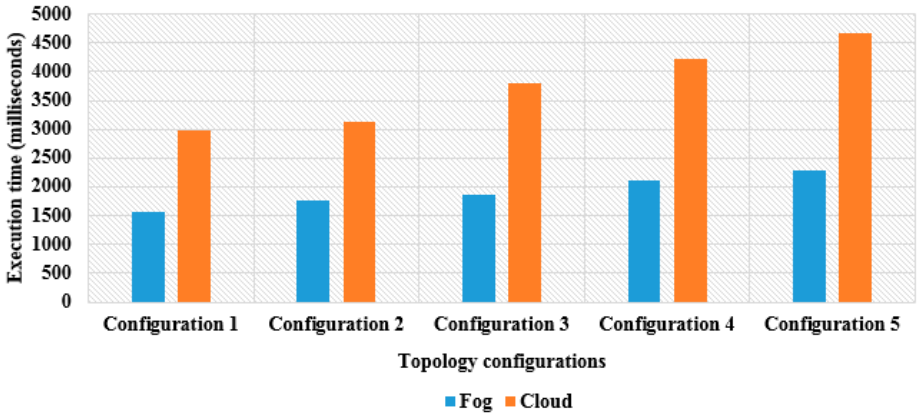

Figure 9. Execution time of fog and cloud computing.

The scalable analysis for configuration 6 with number of devices (20, 40, 60, 80, and 100) is specified in Table 11. The data in Table 11 shows the relationship between the number of devices, response time, energy consumption, and execution time within the IoT system. As the number of devices increases, there is a notable trend of rising response times, implying potential scalability challenges. Correspondingly, energy consumption also escalates with the growing number of devices, indicating the demand for energy resources to manage larger device populations. Additionally, the execution time extends as the number of devices increases, likely due to increased processing overhead associated with handling more devices concurrently. These observations underscore the need of the proposed system to efficiently accommodate expanding device counts. The proposed system addresses potential bottlenecks and enhance system performance, especially in scenarios where real-time responsiveness and energy efficiency are critical considerations.

Table 11. Scalable analysis for configuration 6 of 100 nodes.

Number of Devices	Response Time (in ms)	Energy Consumption (in J)	Execution Time (in ms)
20	63	2400	2600
40	78	3100	2900
60	96	3600	3500
80	105	4200	4200
100	128	4700	4700

Table 12 indicates that the processing time of the mathematical model is increased, while the number of blocks increases. Table 12 clearly shows that the fog node's processing

time is limited than the cloud servers. The hybrid mathematical model (ECC and PR with ESSA) has a lower processing time than the traditional cryptographic algorithms: ECC, PR, Rivest–Shamir–Adleman (RSA) 2048 bits, Digital Signature Algorithm (DSA), and Diffie–Hellman (DH) in fog and cloud computing. The minimal processing time of the hybrid model in fog and cloud computing is 220.18 ms and 598.60 ms with ten blocks. Conversely, the higher processing time of ECC and PR with ESSA in fog and cloud computing is 389.34 ms and 802.10 ms with 50 blocks. The results show that the hybrid mathematical model (ECC and PR with ESSA) has better performance than the traditional cryptographic algorithms in the fog and cloud computing environments.

Table 12. Processing time of fog and cloud computing in ms.

Processing Time (ms)			
Blocks	**Algorithms**	**Fog Computing**	**Cloud Computing**
10	RSA	287.34	675.20
	DSA	280.38	660.92
	DH	267.62	653.23
	ECC	245.33	644.58
	PR	238.02	623.10
	ECC and PR with ESSA	220.18	598.60
20	RSA	337.65	722.65
	DSA	310.22	712.30
	DH	308.72	708.74
	ECC	302.33	690.44
	PR	298.22	687.04
	ECC and PR with ESSA	280.09	630.42
30	RSA	376.80	755.54
	DSA	350.88	748.61
	DH	342.72	740.77
	ECC	333.11	712.29
	PR	329.23	707.46
	ECC and PR with ESSA	309.12	690.44
40	RSA	410.62	852.02
	DSA	405.42	830.87
	DH	394.50	820.82
	ECC	387.60	810
	PR	360	808.18
	ECC and PR with ESSA	344.22	788.43
50	RSA	430.71	876.29
	DSA	421.92	860.92
	DH	416.56	859.33
	ECC	411.28	840.59
	PR	400.38	833.23
	ECC and PR with ESSA	389.34	802.10

Table 13 represents the packet error of cryptographic algorithms between cloud servers and fog nodes at dissimilar time intervals (10, 20, 30, 40, and 50 min). Table 13 clearly shows

that the packet error in cloud servers is higher related to the fog-nodes. In this scenario, the iFogSim is utilized for simulating the cryptographic algorithms. The iFogSim effectively records the packet error during data transmission between the cloud servers, end-users, fog computing nodes, and healthcare IoT. Table 13 clearly shows that ECC and PR with ESSA for 50 intervals obtained a packet error of 24 for fog computing and 38 for cloud computing.

Table 13. Packet error of fog and cloud computing.

Packet Error			
Interval (Minutes)	**Algorithms**	**Fog Computing**	**Cloud Computing**
10	RSA	16	24
	DSA	15	22
	DH	13	21
	ECC	11	18
	PR	8	16
	ECC and PR with ESSA	6	12
20	RSA	19	32
	DSA	17	27
	DH	16	25
	ECC	14	22
	PR	10	21
	ECC and PR with ESSA	8	16
30	RSA	21	31
	DSA	20	29
	DH	18	28
	ECC	16	26
	PR	13	23
	ECC and PR with ESSA	11	22
40	RSA	24	42
	DSA	22	40
	DH	21	38
	ECC	20	35
	PR	17	32
	ECC and PR with ESSA	16	28
50	RSA	34	52
	DSA	32	50
	DH	30	47
	ECC	29	44
	PR	27	40
	ECC and PR with ESSA	24	38

Correspondingly, Table 14 represents the reliability percentage of cryptographic algorithms in cloud servers and fog nodes at dissimilar time intervals (10, 20, 30, 40, and 50 min). As shown in Table 14, the fog nodes have a higher reliability percentage than the cloud server. The minimal reliability percentage of the hybrid model in fog and cloud computing is 80% and 50% at a time interval of 10 min. Conversely, the maximum reliability percentage of the hybrid model in fog and cloud computing is 96% and 75% at a time

interval of 50 min. The obtained results demonstrate the efficacy of the hybrid model over the traditional cryptographic algorithms.

Table 14. Reliability percentage of fog and cloud computing.

Reliability (%)			
Interval (Minutes)	**Algorithms**	**Fog Computing**	**Cloud Computing**
10	RSA	55	36
	DSA	58	40
	DH	60	42
	ECC	62	45
	PR	74	52
	ECC and PR with ESSA	80	55
20	RSA	50	36
	DSA	58	42
	DH	60	44
	ECC	64	48
	PR	76	54
	ECC and PR with ESSA	85	60
30	RSA	58	40
	DSA	62	46
	DH	63	50
	ECC	66	52
	PR	78	57
	ECC and PR with ESSA	90	65
40	RSA	56	42
	DSA	62	50
	DH	68	54
	ECC	70	56
	PR	80	61
	ECC and PR with ESSA	95	70
50	RSA	72	52
	DSA	76	56
	DH	79	59
	ECC	82	62
	PR	88	70
	ECC and PR with ESSA	96	75

Table 15 represents the verification time analysis between the fog computing and cloud computing environments for different cryptographic algorithms such as RSA, DSA, DH, ECC, PR, and ECC and PR with ESSA. This security verification logic analysis helps in identifying logical vulnerabilities for enhancing the overall reliability and security of healthcare services, ensuring the overall security, protecting patient data, and maintaining regulatory compliance. As shown in Table 15, in comparison to other algorithms, the ECC and PR with ESSA has minimal verification time for different blocks (10, 20, 30, 40, and 50). Table 15 clearly shows that proposed ECC and PR with ESSA achieved a verification time of 486 ms and 910 ms for fog computing and cloud computing, respectively, at 50 blocks.

Table 15. Verification time of fog and cloud computing in ms.

Verification Time (ms)			
Blocks	**Algorithms**	**Fog Computing**	**Cloud Computing**
10	RSA	385	763
	DSA	350	754
	DH	327	748
	ECC	318	742
	PR	308	720
	ECC and PR with ESSA	297	692
20	RSA	480	828
	DSA	476	805
	DH	463	800
	ECC	414	788
	PR	396	784
	ECC and PR with ESSA	384	767
30	RSA	490	894
	DSA	488	881
	DH	472	877
	ECC	455	869
	PR	449	846
	ECC and PR with ESSA	418	784
40	RSA	564	998
	DSA	542	987
	DH	492	982
	ECC	488	970
	PR	460	918
	ECC and PR with ESSA	444	883
50	RSA	598	979
	DSA	592	962
	DH	557	953
	ECC	518	940
	PR	508	931
	ECC and PR with ESSA	486	910

Table 16 provides the comparative analysis of response time by analyzing the existing technique ASE [28]. The comparison is taken for configurations 1, 2, 3, 4 and 5. Table 16 clearly shows that proposed ECC and PR with ESSA has accomplished a better response time of 999.9843 ms for cloud computing and 90.6459 ms for fog computing at configuration 5. In comparison, the existing ASE [28] obtained 1691.5143 ms and 93.5615 ms for cloud and fog computing, respectively.

Table 16. Comparative analysis of response time.

Configurations	Response Time (in ms)		
	Algorithms	**Fog Computing**	**Cloud Computing**
1	Existing ASE [28]	41.2971	213.3125
	ECC and PR with ESSA	37.8256	205.2461
2	Existing ASE [28]	53.6751	275.7275
	ECC and PR with ESSA	51.7238	320.3644
3	Existing ASE [28]	71.4285	451.5332
	ECC and PR with ESSA	68.6384	448.7168
4	Existing ASE [28]	84.7122	684.7891
	ECC and PR with ESSA	79.3742	790.2873
5	Existing ASE [28]	93.5615	1691.5143
	ECC and PR with ESSA	90.6459	999.9843

From the overall analysis of identification and authentication of healthcare IoT, the proposed ECC and PR with ESSA model enhanced reliability by 25% to 3% and reduced processing time from 60 milliseconds (ms) to 18 milliseconds when compared to the conventional cryptographic algorithms. Furthermore, the maximum reliability percentages of the ECC and PR with the ESSA model in fog and cloud computing are 96% and 75% at a time interval of 50 min compared to typical cryptographic algorithms. Furthermore, the proposed ECC and PR with ESSA reduces 40% of response time when compared with existing ASE [28] algorithm.

4.2. Discussion

The current use of fog computing significantly overcomes the problem of IoT device verification, identification, and authentication for scalable transmission of EHRs. In this research manuscript, the proposed fog computing system includes a hybrid mathematical model: ECC and PR with ESSA for secure transmission of EHRs. The proposed fog computing system with a mathematical model provides secured services for transmission and transaction. The ECC and PR with ESSA securely exchange the patients' health information between different systems and healthcare providers. It is crucial to maintain continuity of healthcare, particularly when different specialists are involved in the treatment and when patients' move between healthcare facilities. By inspecting the experimental results, the hybrid mathematical model: ECC and PR with ESSA has better response time than the existing ASE [28] algorithm, which has complex mathematical operations that result in high computational overhead in both fog computing and cloud environments. Furthermore, the proposed ECC and PR with ESSA minimizes 40% of response time than the existing ASE [28] algorithm. The reliability percentage of the hybrid mathematical model: ECC and PR with ESSA is 96% and 75% in fog computing and cloud computing environments, where it is significantly better than the conventional algorithms. The effectiveness of the proposed fog computing system is represented in Tables 11–15 with asymmetric methods. Furthermore, in the context of medical data security and exchange, the hybrid mathematical model: ECC and PR with ESSA has a communication cost of 4260 bits and memory usage of 680 bytes.

5. Conclusions

In this research manuscript, a hybrid mathematical model: ECC and PR with ESSA is used in fog computing for effective IoT device verification, identification, and authentication of EHRs. Here, the proposed machine learning model superiorly detects and predicts the system vulnerabilities in the IoT environment. Generally, fog computing reduces the network response time and boosts network security and decentralization ability. In cloud

communication, the fog nodes effectively manage a considerable amount of computing. It is possible to achieve optimal solutions in real time services with the help of fog computing and IoT technologies. As seen in the results section, the security analysis demonstrates the effectiveness of the proposed model. In particular, the performance measures: energy consumption, throughput, response time, execution time, processing time, packet error, reliability, and verification time prove that the proposed model superiorly resolves the security problems. Compared to the conventional cryptographic algorithms, the proposed model approximately reduced 15% to 25% of energy consumption, execution time, packet error, and processing time. Furthermore, the maximum reliability percentage of the ECC and PR with ESSA model in fog and cloud computing is 96% and 75% at a time interval of 50 min, which are far better than the traditional cryptographic algorithms. In the hybrid mathematical model: ECC and PR with ESSA, an appropriate objective function is defined as capturing the trade-off between security, computational efficiency, and key size, even though the data transmission between the cloud and healthcare IoT needs additional privacy and secure transmission of EHRs. As a future extension of this research, a blockchain secure interface will be integrated with different attack in an IoT–fog computing system for the secure transmission of healthcare data.

Author Contributions: The paper investigation, resources, data curation, writing—original draft preparation, writing—review and editing, and visualization were done by P.B.C. The paper conceptualization and software were conducted by G.P.R. The validation and formal analysis, methodology, supervision and project administration of the paper were conducted by M.G.-T. and R.R. All authors have read and agreed to the published version of the manuscript.

Funding: This research was partially funded by MCIN/AEI/10.13039/501100011033 under the grant PID2020-117759GB-I00, the Spanish Ministry of Science and Innovation for the support under the project PID2020-117954RB-C21 and the European Regional Development Fund and Junta de Andalucía for projects PY20-00870 and UPO-13851.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Chakraborty, C.; Othman, S.B.; Almalki, F.A.; Sakli, H. FC-SEEDA: Fog computing-based secure and energy efficient data aggregation scheme for Internet of healthcare Things. *Neural Comput. Appl.* **2024**, *36*, 241–257. [CrossRef]
2. Alnaim, A.K.; Alwakeel, A.M. Machine-Learning-Based IoT-Edge Computing Healthcare Solutions. *Electronics* **2023**, *12*, 1027. [CrossRef]
3. Huang, Y.T.; Chen, T.S.; Wang, S.D. Authenticated key agreement scheme for fog computing in a health-care environment. *IEEE Access* **2023**, *11*, 46871–46881. [CrossRef]
4. Challa, R.; Kothamasu, K.K. A decentralized public-permissioned blockchain framework for enhanced security of health records in fog computing. *Comput. Methods Biomech. Biomed. Eng.* **2023**, 1–13. [CrossRef] [PubMed]
5. Almas, A.; Iqbal, W.; Altaf, A.; Saleem, K.; Mussiraliyeva, S.; Iqbal, M.W. Context-based adaptive Fog computing trust solution for time-critical smart healthcare systems. *IEEE Internet Things J.* **2023**, *10*, 10575–10586. [CrossRef]
6. Alsaeed, N.; Nadeem, F.; Albalwy, F. A scalable and lightweight group authentication framework for Internet of Medical Things using integrated blockchain and fog computing. *Future Gener. Comput. Syst.* **2024**, *151*, 162–181. [CrossRef]
7. Li, J.; Li, D.; Zhang, X. A secure blockchain-assisted access control scheme for smart healthcare system in fog computing. *IEEE Internet Things J.* **2023**, *10*, 15980–15989. [CrossRef]
8. Elayan, H.; Aloqaily, M.; Guizani, M. Digital twin for intelligent context-aware IoT healthcare systems. *IEEE Internet Things J.* **2021**, *8*, 16749–16757. [CrossRef]
9. Sengan, S.; Khalaf, O.I.; Sagar, P.V.; Sharma, D.K.; Prabhu, L.A.J.; Hamad, A.A. Secured and privacy-based IDS for healthcare systems on E-medical data using machine learning approach. *Int. J. Reliab. Qual. E-Healthc. (IJRQEH)* **2022**, *11*, 1–11. [CrossRef]
10. Khan, N.; Zhang, J.; Mallah, G.A.; Chaudhry, S.A. A Secure and Efficient Information Authentication Scheme for E-Healthcare System. *Comput. Mater. Contin.* **2023**, *76*, 3877–3896. [CrossRef]
11. Martinson, E.O. An Efficient Secure Sharing of Electronic Health Records Using IoT-Based Hyperledger Blockchain. *Int. J. Intell. Syst.* **2024**, *2024*, 6995202.
12. Salazar, L.H.A.; Leithardt, V.R.Q.; Parreira, W.D.; Fernandes, A.M.R.; Barbosa, J.L.V.; Correia, S.D. Application of machine learning techniques to predict a patient's no-show in the healthcare sector. *Future Internet* **2022**, *14*, 3. [CrossRef]

13. Makina, H.; Letaifa, A.B.; Rachedi, A. Survey on security and privacy in Internet of Things-based eHealth applications: Challenges, architectures, and future directions. *Secur. Priv.* **2024**, *7*, e346. [CrossRef]
14. Alam, S.; Bhatia, S.; Shuaib, M.; Khubrani, M.M.; Alfayez, F.; Malibari, A.A.; Ahmad, S. An overview of blockchain and IoT integration for secure and reliable health records monitoring. *Sustainability* **2023**, *15*, 5660. [CrossRef]
15. Rajavel, R.; Ravichandran, S.K.; Harimoorthy, K.; Nagappan, P.; Gobichettipalayam, K.R. IoT-based smart healthcare video surveillance system using edge computing. *J. Ambient Intell. Hum. Comput.* **2022**, *13*, 3195–3207. [CrossRef]
16. Tuli, S.; Basumatary, N.; Gill, S.S.; Kahani, M.; Arya, R.C.; Wander, G.S.; Buyya, R. HealthFog: An ensemble deep learning based Smart Healthcare System for Automatic Diagnosis of Heart Diseases in integrated IoT and fog computing environments. *Future Gener. Comput. Syst.* **2020**, *104*, 187–200. [CrossRef]
17. Das, S.; Namasudra, S. A novel hybrid encryption method to secure healthcare data in IoT-enabled healthcare infrastructure. *Comput. Electr. Eng.* **2022**, *101*, 107991. [CrossRef]
18. Geetha, B.T.; Mohan, P.; Mayuri, A.V.R.; Jackulin, T.; Stalin, J.L.A.; Anitha, V. Pigeon inspired optimization with encryption based secure medical image management system. *Comput. Intell. Neurosci.* **2022**, *2022*, 2243827. [CrossRef] [PubMed]
19. Park, Y.-H.; Kim, Y.; Lee, S.-O.; Ko, K. Secure outsourced blockchain-based medical data sharing system using proxy re-encryption. *Appl. Sci.* **2021**, *11*, 9422. [CrossRef]
20. Sutradhar, S.; Karforma, S.; Bose, R.; Roy, S. A Dynamic Step-wise Tiny Encryption Algorithm with Fruit Fly Optimization for Quality of Service improvement in healthcare. *Healthc. Anal.* **2023**, *3*, 100177. [CrossRef]
21. Ali, A.; Almaiah, M.A.; Hajjej, F.; Pasha, M.F.; Fang, O.H.; Khan, R.; Teo, J.; Zakarya, M. An industrial IoT-based blockchain-enabled secure searchable encryption approach for healthcare systems using neural network. *Sensors* **2022**, *22*, 572. [CrossRef]
22. Verma, P.; Tiwari, R.; Hong, W.-C.; Upadhyay, S.; Yeh, Y.-H. FETCH: A deep learning-based fog computing and IoT Integrated environment for healthcare monitoring and diagnosis. *IEEE Access* **2022**, *10*, 12548–12563. [CrossRef]
23. Ahmad, I.; Singh, Y.; Ahamad, J. Machine learning based transformer health monitoring using IoT Edge computing. In Proceedings of the 2020 5th International Conference on Computing, Communication and Security (ICCCS), Patna, India, 14–16 October 2020; pp. 1–5.
24. Sundas, A.; Badotra, S.; Bharany, S.; Almogren, A.; Tag-ElDin, E.M.; Rehman, A.U. HealthGuard: An Intelligent Healthcare System Security Framework Based on Machine Learning. *Sustainability* **2022**, *14*, 11934. [CrossRef]
25. Qureshi, K.N.; Din, S.; Jeon, G.; Piccialli, F. An accurate and dynamic predictive model for a smart M-Health system using machine learning. *Inf. Sci.* **2020**, *538*, 486–502. [CrossRef]
26. Ahmad, S.; Mehfuz, S.; Beg, J. Hybrid cryptographic approach to enhance the mode of key management system in cloud environment. *J. Supercomput.* **2023**, *79*, 7377–7413. [CrossRef]
27. Kondaka, L.S.; Thenmozhi, M.; Vijayakumar, K.; Kohli, R. An intensive healthcare monitoring paradigm by using IoT based machine learning strategies. *Multimed. Tools Appl.* **2022**, *81*, 36891–36905. [CrossRef]
28. Shukla, S.; Thakur, S.; Hussain, S.; Breslin, J.G.; Jameel, S.M. Identification and authentication in healthcare internet-of-things using integrated fog computing based blockchain model. *Internet Things* **2021**, *15*, 100422. [CrossRef]
29. Riad, K.; Hamza, R.; Yan, H. Sensitive and energetic IoT access control for managing cloud electronic health records. *IEEE Access* **2019**, *7*, 86384–86393. [CrossRef]
30. Ray, P.P.; Chowhan, B.; Kumar, N.; Almogren, A. BIoTHR: Electronic health record servicing scheme in IoT-blockchain ecosystem. *IEEE Internet Things J.* **2021**, *8*, 10857–10872. [CrossRef]
31. Alamri, B.; Javed, I.T.; Margaria, T. A GDPR-compliant framework for IoT-based personal health records using blockchain. In Proceedings of the 2021 11th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Paris, France, 19–21 April 2021; pp. 1–5.
32. Alhayani, B.S.A.; Hamid, N.; Almukhtar, F.H.; Alkawak, O.A.; Mahajan, H.B.; Kwekha-Rashid, A.S.; İlhan, H.; Marhoon, H.A.; Mohammed, H.J.; Chaloob, I.Z.; et al. Optimized video internet of things using elliptic curve cryptography based encryption and decryption. *Comput. Electr. Eng.* **2022**, *101*, 108022. [CrossRef]
33. Ibrahim, S.; Alharbi, A. Efficient image encryption scheme using Henon map, dynamic S-boxes and elliptic curve cryptography. *IEEE Access* **2020**, *8*, 194289–194302. [CrossRef]
34. Lara-Nino, C.A.; Diaz-Perez, A.; Morales-Sandoval, M. Lightweight elliptic curve cryptography accelerator for Internet of Things applications. *Ad Hoc Netw.* **2020**, *103*, 102159. [CrossRef]
35. Sadhukhan, D.; Ray, S.; Biswas, G.P.; Khan, M.K.; Dasgupta, M. A lightweight remote user authentication scheme for IoT communication using elliptic curve cryptography. *J. Supercomput.* **2021**, *77*, 1114–1151. [CrossRef]
36. Roy, S.; Khatwani, C. Cryptanalysis and improvement of ECC based authentication and key exchanging protocols. *Cryptography* **2017**, *1*, 9. [CrossRef]
37. Agyekum, K.O.B.O.; Xia, Q.; Sifah, E.B.; Cobblah, C.N.A.; Xia, H.; Gao, J. A proxy re-encryption approach to secure data sharing in the internet of things based on blockchain. *IEEE Syst. J.* **2022**, *16*, 1685–1696. [CrossRef]
38. Manzoor, A.; Braeken, A.; Kanhere, S.S.; Ylianttila, M.; Liyanage, M. Proxy re-encryption enabled secure and anonymous IoT data sharing platform based on blockchain. *J. Netw. Comput. Appl.* **2021**, *176*, 102917. [CrossRef]
39. Liu, J.; Liu, Z.; Sun, C.; Zhuang, J. A data transmission approach based on ant colony optimization and threshold proxy re-encryption in WSNs. *J. Artif. Intell. Technol.* **2022**, *2*, 23–31. [CrossRef]

40. Khashan, O.A. Hybrid lightweight proxy re-encryption scheme for secure fog-to-things environment. *IEEE Access* **2020**, *8*, 66878–66887. [CrossRef]
41. Ibrahim, R.A.; Ewees, A.A.; Oliva, D.; Abd Elaziz, M.; Lu, S. Improved salp swarm algorithm based on particle swarm optimization for feature selection. *J. Ambient. Intell. Humaniz. Comput.* **2019**, *10*, 3155–3169. [CrossRef]
42. Hichri, A.; Hajji, M.; Mansouri, M.; Nounou, H.; Bouzrara, K. Supervised machine learning-based salp swarm algorithm for fault diagnosis of photovoltaic systems. *J. Eng. Appl. Sci.* **2024**, *71*, 12. [CrossRef]
43. Nadimi-Shahraki, M.H.; Zamani, H.; Asghari Varzaneh, Z.; Mirjalili, S. A systematic review of the whale optimization algorithm: Theoretical foundation, improvements, and hybridizations. *Arch. Comput. Methods Eng.* **2023**, *30*, 4113–4159. [CrossRef] [PubMed]
44. Hegazy, A.E.; Makhlouf, M.A.; El-Tawel, G.S. Improved salp swarm algorithm for feature selection. *J. King Saud Univ. Comput. Inf. Sci.* **2020**, *32*, 335–344. [CrossRef]