An Efficient Fog-Assisted Framework For Wireless Sensor Networks By Exploiting Binary Edwards Curves Cryptography

Ethar Sabah Mohammad Ali, Karrar Ahmed Kareem

1College of Engineering, Wasit University, Ministry of Higher Education and Scientific Research, Iraq.

2Ministry of Oil, Thiqar Oil Company, Iraq.

Abstract

Intelligent Sensing and IoT applications have joined forces in critical missions since the advent of Internet technology. Wireless Sensor Networks (WSNs), for example, serve as the primary activation technology in IoT architectures and expand their range of intelligent applications. This network should use dependable and cost-effective network techniques while adhering to the constraints of low power consumption and transmission. However, this technology is resource constrained and suffers from a number of vulnerabilities and security flaws. The goal of this proposal is to use the BEC encryption method to improve the efficiency of the system's security structure and to adapt it to IoT network characteristics such as resource constraints and specific attacks on these networks. To simulate, the Ifogsim simulator is used. By examining the simulation results, we concluded that the proposed solution has a better status compared to the basic article method, both in terms of encryption time and in terms of authentication time. In the case of the network consumption factor, the number of packages that should have been exchanged for various operations, such as authentication through gateways, has been reduced by using the fog element. Data recovery and security-related operations have also decreased. In terms of privacy protection, because gateway encryption was done in groups, data owners' privacy was preserved, and this shortcoming was actually compensated for in the basic article method.

Keywords: Internet of Things (IoT), Wireless Sensor Networks (WSN), BEC encryption, security.

Introduction

WSNs are increasingly being used for data communication and processing. A WSN infrastructure is composed of a large number of independent sensor nodes and a base station, with the base station
serving as a gateway to another network. A sink node typically performs the function of a base station; this could be a laptop or a computer system that collects and analyzes data in order to make appropriate decisions. A WSN can be made up of various types of sensor nodes, such as low sampling rate magnetic, thermal, visual, infrared, and acoustic nodes. Each node's sensor can detect phenomena such as light, pressure, heat, and so on. The sensor is powered by a small battery, as a source of power, which means that network performance is highly dependent on the rate of energy consumption. [1].

WSNs are made up of one sink node and a large number of sensor nodes spread across a large area (sensing field). Data is transmitted from nodes to base stations via single-hop or multi-hop communication, and then to users via the internet. [2].

Due to the nature of broadcast communication, there is the possibility of hooligans, intrusion, and data packet alteration in wireless sensor networks. To protect the network from these threats, it is critical to provide security in this field for securely routing data in the network [3].

Cloud computing has emerged as a new computing paradigm over the last decade. Its vision is to centralize computing, storage, and network management in Clouds, which refer to data centers, backbone IP networks, and cellular core networks. In recent years, a new trend in computing has emerged, with the function of Clouds increasingly moving to the network edges [4].

Edge devices, such as routers, gateways and even sensor nodes interact with cloud servers to provide services on the fog computing platform. These fog nodes are equipped with storage and a little amount of compute power. There are instances in the fog network when several edge servers, referred to as "cloudlets," engage in a distributed computing environment. Fog devices can provide consumers with real-time responses to latency-sensitive applications. Fig. 1-2 depicts a typical fog computing platform's design. In addition to routers and gateways, all the end devices are connected to them. The routers and gateways are then connected to the remote cloud server. A multi-tiered fog architecture means that the fog spreads from edge devices to cloud servers [5].
Challenges

The basic article had the following challenges in the following areas:

- In paper method [8], ECC algorithm is used for cryptographic operations. However, this method has disadvantages as well.
- Weierstrass form accounts are not compatible with low resource IoT devices.
- In addition, these curves are not designed and selected with the additional constraint of resistance to lateral channel attacks and error attacks.
- In addition, paper [8] only uses a local server that performs functions such as real-time processing and response, authentication operations without the need for public cloud, and does not have storing sensor data received at shorter intervals than the cloud. These capabilities will cause speed up network acceleration, reduce network traffic and increase performance.
- The firewall server, which is responsible for verifying the data sent by the sensor, can also become a Single point of failure. In addition, the use of such a centralized server in the system, it will make scalability in the system with difficult.

Research goals

Given the challenge, in the present study, we intend to present an algorithm to solve the main article method problems such as non-compliance with resource constraints in the IoT network as well as resistance to some attacks by applying an optimal encryption method. Resolves. Also, the proposed method should solve the problems of the basic article method which results from performing all the processes in the public cloud. Improving the design of the firewall server to suit the network conditions is another objective of the proposed method.
Therefore the aims of the research are as follows:

- Replacing the previous encryption method with an optimized one to match the resource constraints of the IoT and resist all attacks in the domain.
- Find a solution to respond to real-time system requests and perform authentication without the need for public cloud
- Improve the design of distributed firewall server and pay attention to system scalability

DEFINITIONS OF BASIC CONCEPTS

Wireless sensor network
WSNs are made up of many miniaturized embedded devices with wireless communication, processing and sensing capabilities, as shown in Figure 2. Reduced cost, simple deployment and small size, make WSN attractive for military use [11].

![Figure 2 WSN node architecture](http://www.webology.org)

Specific constraints in WSNs
WSN nodes have very limited energy, memory and processing capabilities, and constrained communication bandwidth. The main characteristics of WSNs, are [12]:

- Memory limitations: The available embedded memory in a sensor node is too small. Besides, it is shared by the operating system and the added value routines.
• Energy constraints: The most important constraint in WSN is the energy. In fact, nodes have limited energy autonomy, and this energy is not usually renewable. Any processing overhead will affect their availability.

• Hostile deployment environment: Their deployment in open and hostile environment makes them exposed to vandalism and security breaches such us signal jamming, eavesdropping and spoofing.

• Wireless ad-hoc nature: The use of ad-hoc wireless communication adds new security challenges to secure the interconnection between nodes, while providing a redundancy in the path of connectivity.

• Topology changes: WSNs have no static topology. Nodes can join the network at any time, others can leave it voluntarily or by being forced to quit, due to a vandalism act or energy disruption. Topology control is also important in WSNs to preserve energy and network connectivity without affecting throughput and security.

A) Precision Agriculture

As part of its mission to reduce environmental impact while increasing resource efficiency, precision agriculture seeks to improve cultural operations. Sensor data is utilized to determine the best sowing density, estimate fertilizer and other input requirements, and more accurately predict crop yields. In agriculture, WSN is an unavoidable factor. To gather and monitor data like temperature, humidity and carbon dioxide gas concentrations, the field is equipped with a number of wireless sensor nodes. A human expert receives the sensed and collected data via cloud computing or the Internet [13].

B) Environmental Monitoring

Coal mining, earthquake and tsunami detection, flood detection, forest fire forecasting, gas leakage, cyclones, rainfall range, water quality, and volcanic eruptions, among other things, can benefit from the deployment of wireless sensor networks in the environment. Due to its ability to detect and predict environmental disasters early on, the network aids in the implementation of safety precautions. The sensors collect the data, which is then sent via the Internet to the central station. As a result, people are more likely to take preventative measures and become aware of the impending calamity. This section focuses on the use of wireless sensor networks in environmental monitoring.

C) Vehicle Tracking

WSN can also be used in the field of smart transportation. Traffic flow is monitored with a network of cameras and sensors to alleviate congestion, track vehicles in the city to catch unlawful activity and monitor important infrastructure like airports and train stations for illicit activity.
D) Health care Monitoring

The employment of WSN in the medical industry has become standard procedure in the modern world. Sensors are used to monitor a variety of bodily functions. After that, the data is sent to the doctor, who can use it to make a diagnosis.

E) Smart Buildings

In an intelligent building, all of the building's systems can be monitored and controlled by the building itself. To what extent may a building's features be used depends on its overall scope.

F) Military Applications

In armed C4ISRT systems, WSN is a critical component. It allows the systems to communicate, coordinate, compute, gather intelligence, monitor, detect, and target threats. Sensor networks, because of their rapid exploitation, self-organization, and error-tolerance, are a particularly promising sensing approach for military C4ISRT. Due to the fact that sensor networks rely on the extensive use of disposable and low-cost nodes, the destruction of a few nodes does not have the same impact on military operations as the eradication of a standard sensor.

G) Animal Tracking

In animal tracking, a sensor is connected to the animal's body so that its movement and location may be tracked using wireless sensor networks. It is possible to track zebras in the wild using a wireless sensor network called Zebra Net. In order to monitor the animal's position, location and food intake, sensors are inserted into the animal's body.

Fog computing

Cloud computing and IoT nodes are connected by fog computing, which allows computation, storage, networking, and data management on network nodes within close proximity of IoT devices. Fog computing is a bridge between cloud computing and IoT devices. Consequentially, the cloud's computing capabilities are not limited to the IoT-to-Cloud route; they are also present in the IoT-to-Cloud path itself (preferably close to the IoT devices). Intelligent Transportation Systems (ITS) can, for example, compress GPS data before it is transmitted to the cloud (ITS). The Open Fog Consortium defines fog computing as "a horizontal system-level architecture that distributes processing, storage, control, and networking services closer to the users along a cloud-to-thing continuum." Fog computing uses a "horizontal" platform, which allows computer tasks to be disseminated across many platforms and industries, whereas a vertical platform encourages silos applications. Even though a vertical platform may be well-suited to supporting a specific class of application (a "silo"), it neglects to take into account cross-platform communication in other vertical platforms. Fog computing provides a flexible platform for operators and users to meet their data-driven objectives in addition to facilitating a horizontal architecture. To support the Internet of Things, fog computing has been designed and implemented.
Figure 4 depicts a three-tiered fog system. The cloud stratum, the fog stratum, and the IoT/end-user stratum make up the three layers of the system. If there are multiple fog domains, they can all be controlled by the same or distinct service providers. As a result, each of these domains is established by a group of fog nodes that can be anything from edge routers to access points to PCs to cellphones. The IoT/end-user stratum is made up of two domains, the first of which includes end-user devices, and the second of which includes IoT devices. It's important to keep in mind that the stratum may lack one of these two domains. Fog systems, for example, use this technology to convey content. There is no Internet of Things (IoT) domain. Local Area Networks (LANs) are used to communicate between the IoT/end-users stratum and the fog stratum (LAN). IoT/end-users and the cloud stratum must instead communicate via WAN, whether or not the fog is present, to exchange data. Fog can be depicted in literature in a variety of ways. End consumers are mostly ignored in favor of the Internet of Things (IoT) as a whole. Thus, non-IoT applications like CDNs are out of the question (CDNs). There is no mention of the possibility of many fog domains in any of them.

![The Fog System](image)

**Figure 4 The Fog System [15]**

**Authentication**

Users and devices are authenticated in order to grant access to those who are not authorized or who have not been tampered with. IoT systems can be protected from attacks like response attacks, middle man attacks, impersonation attacks, and Sybil attacks by implementing authentication mechanisms.

Using the Internet of Things (IoT), Ruhul and colleagues [18] have devised a distributed cloud architecture in which a private cloud holds secret data. For secure access to private cloud servers in the distributed system, this article develops a standard authentication protocol that resists all
forms of security assaults and provides crucial features like user anonymity, a shared sense of belonging. The security of the protocol has been proven using BAN logic and AVSIPA simulation findings. For the security assessment, they also used the Pro-verif simulator. Because of the hash function's hardness assumption, informal cryptanalysis ensures the protocol's security against security attacks. The protocol performance study outperforms other efforts in terms of compute, storage, and communication costs. It's a major accomplishment. There is no password verifier table in the proposed protocol, which makes it possible for authorized users to make changes to their passwords and identities.

For capacity-based security access authentication, Ming et al. introduce Access Auth, a lightweight protocol. Each V2G network domain is given an adaptive number of admissible access requests based on overload probability and system capacity limits, as well as the mobility of electric vehicles, to actively achieve capacity-based access admission control. This is followed by a high-level authentication model with specific authentication procedures to enforce strict access authentication so that only authorized requesters can conduct sessions, taking into account whether there is prior knowledge of the trust relationship between the relevant V2G network domains. This ensures the privacy of admitted sessions. Additionally, revocation of a session with forward security and recovery of a session.

Finally, Access Auth's performance is demonstrated by analyzing and evaluating its outcomes.

Xianjiao et al. [20] developed an Efficient Anonymous User Authentication (E-AUA) protocol between users and servers based on multi-server architectures, which contain several servers, to address the issue of network congestion in mobile IoT. As an added benefit, the E-AUA protocol was created with an attack-resistant dual-message technique that further reduces communication and computation costs. The E-AUA protocol reduced both communication and calculation costs when compared to competing protocols. To demonstrate our E-AUA protocol's robust security in a mobile IoT setting, the researchers also conducted an in-depth security analysis.

**Cryptography**

Nodes are encrypted for end-to-end security. However, because IoT systems are so diverse, certain nodes may be able to use all-purpose microprocessors. However, in devices with restricted resources, only application-specific ICs can be used. Traditional primary encryption is not ideal for low-resource smart devices because of their lack of computational power, battery life, size, memory, and power supply. As a result, these devices may benefit from using a more efficient kind of encryption, such as lightweight encryption. For IoT encryption, lightweight symmetric and asymmetric algorithms are designed to address the demand for efficient end-to-end communications with minimal power consumption.

Chaudhry [21] explored all of the IoT's limitations in an incredibly thorough manner. Various requirements for IoT security have been outlined in this study, which are critical to achieving the
goal of security. An initial step toward solving this security problem with salt-based encryption is depicted in this calculation. This is made possible by the small piece design, which speeds up the entire process at the same time. Some IoT are vulnerable to security attacks because there is no genuine countermeasure in place to protect them from these attacks, according to Propel's investigation of security measures.

ANU-II, proposed by Dahiphale and et al. [22], is a novel lightweight encryption scheme that supports 64-bit blocks and 80/128-bit keys. As the name suggests, this new lightweight encryption is an improved variant of the well-known ANU. Memory capacity, latency, throughput, and power are the parameters that make the ANU-II more efficient than the original ANU. On all relevant design metrics, ANU-II outperforms the cipher ANU. 20 percent less RAM is needed for ANU II's cipher design than for ANU's original version. As opposed to the ANU design, the ANU-II cipher design requires only 146.35 us to encrypt an entire 64-bit block of plaintext. On the same hardware, ANU cipher design consumes 27mW of dynamic power, whereas ANU-II only uses 24 mW. A single S-Box, a few shift operators, and a few XOR gates make up the ANU-II design. In terms of memory requirements, execution time, and power consumption, they believe the ANUII design to be the most energy-efficient and least heavy currently available.

OTFEP, a new scalable one-time file encryption approach presented by Wei and coworkers [23], uses a combination of dependable cryptographic techniques to meet specific security needs. OTFEP provides a technique to guard database files from arbitrary access by system administrators or third-party auditors. When dealing with tiny files and streams, OTFEP employs a two-pronged strategy. As a result, OTFEP provides excellent scalability for nodes and a secure key distribution method. OTFEP can be explored for use in specific IoT devices that require one-time file encryption due to its practical security and performance.

**Trust Management**

On the subject of device trust management, there are numerous publications. The goal of IoT trust management is to detect and remove rogue nodes while also ensuring that only authorized users have access to data. Some of the newest work in trust management involves automating and dynamically validating trust calculations for the nodes participating in an IoT network. Many researchers are focused on discovering bad nodes, although this is not the case. Trust-based access control is limited to just a few options available. An urgent requirement for an automated and transparent access control management system is due to the scalability and high number of intelligent objects that store sensitive data. This allows different nodes and users to have differing levels of permission.

Robust trust, a system proposed by AWAN and et al. [24], is a cross-domain distributed trust management system that enables devices to be trusted locally by analyzing their trustworthiness towards other devices. There are three parts of security in this system that assist IoT nodes stand up to hacked and hostile devices/nodes. Proposed mechanisms include a scalable trust mechanism,
various assessment components to strengthen robustness against assaults and recommendations along with the feedback to generate knowledge. The originality of these aspects can be stated as follows: A key feature of the proposed method is that it is event-driven, which makes it easier for nodes to assess trust and hence improves overall system performance. It is compared to existing trust evaluation techniques by focusing on several aspects, such as trustworthiness, usability and accuracy among others, in this proposed work In-depth simulations of the Robust Trust's performance, accuracy of trust estimation, and several potential assaults are used to validate it.

There is a new approach to managing trust in the IoT called Holi Trust, developed by AWAN and co-authors [25]. Many mechanisms for managing trust have been offered in previous studies, but the lack of a mechanism for managing trust across domains has been overlooked for years. The Holi Trust relies on various central authority, such as the community, domain, and trust servers, to maintain its accuracy and efficiency. IoT nodes benefit from these authorities, which improve accuracy and minimize computing weight. Reduced compute weight reduces susceptibility, increases resilience to assaults, and provides acceptable security. The Holi Trust uses community trust and domain trust, as well as trust parameters, to effectively secure cross-domain communication.

CATMEC, a clustering technique based on trust management and edge computing, was proposed by Kammoun and et al [26]. Defending the IoT environment against malicious node attacks necessitates the establishment of a trust connection between nodes. The trustworthiness of a node is determined by the extent to which it cooperates and is honest. As a result, they developed a one-hop clustering technique based on Trust Management, energy, and density in order to address the issue of restricted node capacity and energy. On order to move data processing and storage to the BSs in the network's perimeter, they used edge computing technology. Inter-cluster and external communication is made possible by the usage of BSs as a TTP.

**Safe Routing**

The Internet of Things relies heavily on sensors and actuators. The self-organization and information sharing of these devices, despite their resource limitations and high power consumption, make them ideal for the Internet of Things (IoT). At the same time, they are also data storage devices and computers. As a result, any routing solution must have the potential to scale, be automated, and be energy efficient. In order to link a low-power, lossy network (LLN) to the Internet or a neighboring local area network, some of these sensor nodes act as boundary routers (LAN). The IP addresses of IoT devices are based on IPv6 due to the large scope of these networks. IPv6 in Low Power Wireless Networks (6LoWPAN) is an IETF IPv6 adaptive layer that permits IP connections across a low power network with no loss. 6LoWPAN's lack of authentication means that security breaches are quite possible.

For secure routing in ad-hoc networks, Hatzivasilis and colleagues [27] offer SCOTRES – a trust-based system that utilizes five innovative measures to enhance the intelligence of network
components. When using the energy metric, each node's resource usage is taken into account. This increases the network's lifespan by requiring similar levels of participation. By keeping track of the nodes' locations, the topology metric makes it easier to distribute the load. Channel health metric provides tolerance for intermittent malfunctions due to unfavorable channel conditions and protects the network against jamming assaults. Individual collaboration is evaluated using a reputation metric, whereas trust measures the total level of compliance. The system's security is confirmed by theoretical study. The NS2 simulator is used to assess the performance and effectiveness of SCOTRES and the DSR routing protocol. To ensure a fair comparison, similar methods are executed on the same platform. And in addition, SCOTRES is used in a rural application to monitor environmental parameters on olive trees on two embedded system platforms. The aforementioned evaluations show that the system offers the maximum level of security while still being efficient for real application deployments, as can be shown. An IoT multicast routing system, REMI, has been proposed by Conti and et al. [28]. REMI's primary goal is to enable efficient communication in low-power and lossy networks like the Internet of Things (IoT) by assuring that a message will be received by all its intended recipients, regardless of the network size or the presence of misbehaving nodes. Using cluster-based routing, REMI is able to distribute messages more quickly inside the network. An operating system commonly used by industry to create energy- and memory-constrained wireless networks, Contiki, was used to implement REMI. They undertake a large number of simulations to gauge REMI's efficacy and efficiency. In terms of network speed, propagation latency, and scalability, results reveal that the protocol outperforms state-of-the-art protocols at a minimal cost in terms of energy consumption and memory utilization.

LASeR, a secure onboarding and routing system for NDN-based IoT networks, has been suggested by Mick and et al. [29]. Through a hierarchical network design and minimal cryptographic or computational burden, scalability is accomplished. LASeR's simulation results show that it uses very little network overhead and provides acceptable onboarding convergence times. However, an addition is planned to permit the marketing of any name prefix in the LASeR routes currently based on node IDs. The development of a low-overhead method for dealing with node mobility is also underway. In the future, they plan to use LASeR on real IoT devices to conduct a live testbed deployment.

**PROPOSED METHOD**

**Proposed method**

- The ECC technique was utilized for cryptographic operations in the paper [8]. Many networks employ this technology as a common cryptography strategy. Although this technique has been widely adopted, there have been studies that have uncovered several problems in this cryptographic algorithm and have questioned the system's performance in certain circumstances. As a result, ideas for addressing the shortages were floated. The
Edward Curve binary method (BEC) was one of these methods. These two algorithms will be briefly described below so that you are familiar with their features so that you can better comprehend the differences between them and the improvements that BEC has made to the ECC approach.

● There is just one local server utilized in the study [8] that does not have the capability of real-time processing and response, completing authentication activities without the requirement for public cloud, and storing data received from sensors at more limited intervals than the cloud. In response to these capabilities, the network’s reaction time will be faster, network traffic will be reduced, and network efficiency will be improved.

● Another possible weak spot is the firewall server, which is in charge of determining the accuracy of the data sent from the sensor. In addition, the system's scalability will be hampered by the use of a single central server.

● We plan to use a cryptographic technique to address the issues with the published method [8], such as its inability to adhere to IoT network resource limits and its vulnerability to certain assaults. As a result, it is expected that the method provided here will address some of the issues of the method in [8]. Another purpose of the suggested method is to improve the architecture of the firewall server that is appropriate for the network conditions.

● For the suggested method, instead of employing the ECC algorithm for cryptographic operations that are incompatible with the device structure and resource limits in IoT networks, we will use the BEC method proposed in the article [7]. We can boost the method's efficiency by doing this. It is possible to add a private cloud layer to our paper structure, which would allow us to do things like authenticate users using the fog element [8]. More privacy is provided by this layer than by the public cloud, as detailed in our study [8].

● To avoid problems with data gathering and virtualization, we use this framework to build a private cloud server. In the paper [8], just one local server, which lacks the capabilities of fog, is used. With these capabilities (such as real-time processing and response, authentication without the need for a public cloud, and storage of data received from sensors at more limited intervals than the cloud), network acceleration in response, reduced network traffic and increased efficiency are all possible. Networks such as healthcare will find this topic of great interest and importance. It's also possible to integrate the firewall server, which is responsible for determining the sensor's data precision and scalability, in the fog while without becoming a single point of failure.

As a result, the following scenarios are covered by the suggested approach:

● In order to increase system security performance and conformity with IoT network aspects such as resource limits and particular assaults on these networks, the BEC encryption method is being implemented in the structure [8].

● Build in a layer of fog enabling real-time and more secure computations at the network edge, without the requirement for a public cloud or local storage of sensor data in order to execute authentication processes.
- Redesign the firewall server in the network to allow scalability by designing it in the fog layer, resulting in a shift from a central to a distributed structure.

**Modeling the problem, assumptions and limitations**
By using a private cloud server, we can avoid issues with data collecting and virtualization that would otherwise arise.

Figure 6 illustrates the framework's primary components: In addition to the user, there is a central server (CS), sensor nodes (SN), a firewall, an intrusion prevention system (IPS), and a private cloud server (PCS).

Local servers receive the data from the sensor nodes, which in turn relay the data to the sensors. The firewall manages the process of connecting nodes to the central server, as well as the flow of data between them. Data flow is controlled, analyzed, and compared to prevent external threats by the IPS. Authentication of users, encryption of data, and communication between the mobile device and cloud server are all handled by the local server. Encrypted data is sent to and stored on the cloud server. The encrypted data may be downloaded, decrypted, and then the secure data can be accessed by the user after registration and authentication. Thus, the security plan is broken down into three primary sections:

1. user authentication step,
2. PCS download step, and
3. the step of data sharing between users.

![Figure 6 General architecture for the proposed framework](http://www.webology.org)
The proposed algorithm
The proposed algorithm consists of three distinct parts and we intend to describe in detail in the following:

The usage of BEC cryptography method
Since the cryptographic method had the problems mentioned in the previous section in order to fit into the structure of wireless sensor networks, we decided to use an improved cryptographic method called the Edward Binary curve, which solves the problems of the ECC method. However, the time and computational overhead of the BEC method is not so different from that of the ECC.

RESULT EVALUATION

Simulation environment
Due to the fact that the IoT platform we are looking for is developed in Java, we need a library that can be used in this environment. A very powerful simulator, Ifogsim Simulator can simulate all wireless networks and use various nodes such as fog, clouds and smart gateways to link this equipment with end users, which was designed based on cloudsim simulator and provides evaluation and measurement for different criteria such as latency, throughput, energy consumption and so on. [30].

The ifogsim simulator is used to model the proposed procedure. It's first necessary to go to the following URL and download the file associated with IFog Sim's simulation environment. https://github.com/harshitgupta1337/fogsim
Use the Eclipse software to create a new project and the instructions in the Readme.txt file to access IFog Sim's classes and graphical user interface in order to run simulations of the required topologies. Version 8 of the Java Development Kit (JDK) is required, as is the most recent version of the Java Runtime Environment (JRE).

The evaluation of algorithm efficiency
In this section, the experiments that are performed to evaluate the performance of the proposed algorithm are described. Among the evaluation criteria of the proposed solution that can be compared with the method of the paper [8], the following three items can be mentioned:

- The time of user authentication
- Encryption and decryption time
- The communication rate and traffic consumption in the network

In order to evaluate the proposed solution, since the comparison is done with the structure of the paper [8], the health network scenario has been used. This means that the sensors are actually sensors exist in the health control network, which intend to collect data about patients from the
environment and send it to the fog element through gateways for encryption, analysis and the usage of users.

**The time of user authentication**

By examining the simulation results (Table 1), we concluded that the proposed solution is in a better condition in comparison to the paper method [8] on encryption time. The reason for better authentication time is that the authentication process of users and sensors is done centrally in the fog element at the network edge and with high processing power. The proposed method has shown a better result than the base paper method, which this process in gateways are performed with low processing power. Figure 1 shows a comparison of user authentication time between the proposed method and the paper method [8].

Table 1- The results of process time for user authentication

<table>
<thead>
<tr>
<th>Size key (bit)</th>
<th>Paper in [8]</th>
<th>Proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>160</td>
<td>0.54</td>
<td>0.49</td>
</tr>
<tr>
<td>224</td>
<td>1.03</td>
<td>0.89</td>
</tr>
<tr>
<td>256</td>
<td>1.34</td>
<td>1.24</td>
</tr>
<tr>
<td>390</td>
<td>1.7</td>
<td>1.55</td>
</tr>
</tbody>
</table>
Figure 7- The comparison of user authentication time between the proposed method and the method.

**Encryption and decryption time**

As you can see from the results in Table 2, the proposed solution is in a better condition than the paper method in case of encryption time. The reason for better encryption time is related to use the BEC encryption method instead of the ECC method, which has adapted itself to the conditions of sensor networks. Figure 2. shows a comparison of encryption and decryption times between the proposed method and the method in [8].

Table 2- The results of encryption and decryption time

<table>
<thead>
<tr>
<th>Size key (bit)</th>
<th>Paper [8]</th>
<th>Proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>160</td>
<td>3.5</td>
<td>2.8</td>
</tr>
<tr>
<td>224</td>
<td>4.1</td>
<td>3.4</td>
</tr>
<tr>
<td>256</td>
<td>5.1</td>
<td>4.1</td>
</tr>
<tr>
<td>390</td>
<td>7.55</td>
<td>6.5</td>
</tr>
</tbody>
</table>

Figure 8- The Comparison of encryption and decryption time between the proposed method and the method in [8]
5-2-3- The communication rate and traffic consumption in the network

Regarding the factor of network consumption or the use of the network (Table 3), for this reason, by applying the fog element, the number of packages that had to be exchanged for various operations, including authentication through gateways, was reduced. The amount of traffic load that is streamed in the network for data collection and related security operations have also been reduced. About the privacy factor, because the encryption operation at the gateways was done in groups, the privacy of the data owners was preserved and in fact this shortcoming was compensated in the method of the paper [8]. Figure 3. shows a comparison of the communication rate and traffic consumption in the network between the proposed method and the method in [8].

Table 3- The results of communication rate and traffic consumption in the network

<table>
<thead>
<tr>
<th></th>
<th>Paper [8]</th>
<th>Proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Usage network</td>
<td>2521</td>
<td>1950</td>
</tr>
</tbody>
</table>

Figure 9- The comparison of communication rate and traffic consumption in the network between the proposed method and the method in [8]

Conclusion

To apply the proposed method, initially instead of using the ECC algorithm for cryptographic operations that is not sufficiently compatible with the device structure and resource constraints in IoT networks, we use the BEC method in the structure presented in the paper [8], which can increase the method efficiency. Moreover, we can add the fog element as a layer in the structure of the paper [8] and assign operations such as user authentication to this layer because given that this layer is a private cloud as it is stated in the paper [8], this layer has more privacy than the public cloud. It should be noted that in the paper [8], only one local server is used and it does not
have the capabilities of fog. (Such as real-time processing and response, authentication without the need for a public cloud, and storage of data received from sensors at more limited intervals than the cloud) These capabilities will accelerate response, reduce network traffic, and increase efficiency and this subject is important for networks such as healthcare. In addition, the firewall server, which is responsible for detecting the precision of the data sent from the sensor, can be implemented in fog as an application based on the SDN structure, so that, while not converting into a single point of failure, it will also support scalability.

By examining the simulation results, we concluded that the proposed solution is in a better condition in terms of both encryption time and authentication time compared to the paper. About the network consumption factor, for this reason, by using the fog element, the number of packages that had to be exchanged for various operations, including authentication through gateways has been reduced, the amount of traffic load flowing through the network to add data collection and related security operations have been also decreased. About the privacy factor, because the encryption operation at the gateways has been performed in groups, the privacy of the data owners has been preserved and in fact these flaws has been compensated in the method of the paper.

**Future works**

In next works, our goal is to improve our confidential framework in terms of complexity and access time for a better experience quality. In addition, we intend to implement our platform in other fields related to cloud in practice for large-scale applications.

**References**


[24]. Awan, Kamran Ahmad, Ikram Ud Din, Ahmad Almogren, Mohsen Guizani, Ayman Altameem, and Sultan Ullah Jadoon. "RobustTrust–A Pro-Privacy Robust Distributed


