Journal of Theoretical and Applied Information Technology
31st December 2022. Vol.100. No 24

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

SECURE INTEGRATION OF WIRELESS SENSOR NETWORK WITH CLOUD USING CODED PROBABLE BLUEFISH CRYPTOSYSTEM

NV RAVINDHAR[1], S SASIKUMAR[2], N BHARATHIRAJA[3], M VINOTH KUMAR [4]

[#1]Research Scholar, Department of Computer Science and Engineering, Saveetha School of Engineering,
Saveetha Institute of Medical and Technical Sciences,
Chennai, Tamil Nadu, India.
[#2]Professor, Department of Computer Science and Engineering, Saveetha Engineering College,
Chennai, Tamil Nadu, India.
[#3]Associate Professor, Chitkara University Institute of Engineering and Technology,
Chitkara University, Punjab, India.
[#4]Lecturer, Department of Computer Science and Engineering,
DMI-St.Eugene University, Chibombo District, Zambia

E-mail: ravindharnv1986@gmail.com, ssasiin@gmail.com, rajamesoft@gmail.com, mvinothcs@gmail.com

ABSTRACT

Wireless sensor networks (WSNs) connect hundreds or thousands of physically separate sensor nodes (or "motes") to gather data from all over a given area. The medical monitoring sector, the weather service, government and military applications, and many more all use WSNs for data collecting. The limitations of WSNs in terms of processing speed, data storage capacity, and power availability all present challenges. Due to the rise of Cloud computing and all the advantages it brings, the idea of combining the two systems has been proposed. The proposed system provides a fresh approach to how WSNs might link up with the Cloud. Using Cloud computing, this research provides a unified WSN for environmental surveillance. The primary goal is to provide a robust and astute infrastructure for the constant flow of sensor data. To do this, the WSN will collect data from its immediate vicinity, compress it, and then relay it via a gateway, which will then expand it and store it in the Cloud server. Both the opportunities and limitations of the WSN, including security concerns and Cloud computing services, will be discussed in this article. Therefore, a hierarchical structure of coded probable blue fish cryptosystem has been constructed in the design of data aggregation in WSN and the Distributed optimized LEACH (DO LEACH) protocol can deal with the restricted resources of the nodes and the route selection necessary for Cloud integration. The bio-inspired compressed cuckoo optimization technique was utilized to fine-tune the protocols. It should be evident by the completion of the article how well the proposed security protocols may maintain the network's total energy by decreasing the energy consumption of individual nodes, all while enabling reliable secure transmission between Wireless Sensor Networks and Cloud servers.

Keywords: *Wireless Sensor Network; Security; Transmission; Energy Consumption; Coded Probable Blue Fish; Cloud.*

1. INTRODUCTION

WSNs are becoming more relevant in the contemporary world because of the abundance of inexpensive sensor devices now available. It is expected that WSNs will be widely implemented in the not-too-distant future. We may do more in-depth analyzes of data, increasing our understanding and enlightenment as a result. Thanks to Cloud computing and sensor networks, this is now feasible. Currently, WSNs are being developed for a wide range of purposes, including monitoring the environment, providing surveillance, providing healthcare, monitoring habitats, monitoring industries, keeping tabs on traffic on highways, and facilitating the development of smart cities. Because of its ad hoc ancestry, WSNs have nodes that are scattered over

large regions and engage in data processing, resource sharing, and other activities.

Ad hoc networks do not have a standardized setup or a central command and control mechanism, and their configurations and settings are determined on the fly. Verification of each node in such a network is required before any form of communication takes place inside the network. However, authentication between nodes is necessary for ad hoc networks before any communication can occur since there is neither a stable configuration nor a centralized organization. Also, safety concerns in WSNs aren't given enough thought, despite their prevalence. This is because sensor nodes have limited resources and network nodes are often in a state of instability.

Secure communication using WSNs is required to guarantee the privacy of the node's sensed data while yet allowing the node's data to be shared with other nodes in the Cloud. Because of this, it's important to reevaluate how private the information should be. Data validation is the duty of confirming that the data came from the purported sender, while data integrity is the assurance that the received data was not tampered with in transit. And last, data confidentiality is the process of checking that the data has not been tampered with by the purported sender. This is done to guarantee that the data came from the purported sender and has not been tampered with in transit. As a bonus, this protocol verifies that the incoming information has not been assumptions have already been made about the implementation of asymmetric cryptography in WSNs.

It is widely agreed that the limited power available to sensor nodes in WSNs makes PKC (Public Key Cryptography) impractical to use there. However, many studies have come out showing that these assumptions are wrong and that it is possible to construct a more secure connection under certain circumstances. Data confidentiality between WSNs and Cloud settings is protected by a mix of coded probable bluefish cryptography and the proposed DO_ LEACH protocol, which aims to preserve overall network energy by saving the energy of its nodes. The DO_ LEACH protocol's end goal is to achieve all of these aims.

2. RELATED WORKS

Several authors have shown different approaches to integrating Cloud WSN across various applications; some of them are presented as examples below. Based on virtualization, the author of [1] offers a large variety of applications that may be run on a single sensor node. In this case, virtualization might potentially shorten the necessary time. Many of the benefits of using a sensor Cloud are outlined in their research. In the existing approach [2], an energy consumption design for WSN and Cloud date projecting is introduced Autoregressive Integrated Moving Average (ARIMA). Typically, the Cloud is used to relay all user requests to the WSN. The standard approach has users making requests every fifteen minutes, requiring constant communication between the sensor and the Cloud. In the current configuration, the WSN's sensors report their status to the Cloud every two hours. In Cloud, the ARIMA data forecasting approach can fulfill the vast majority of customers' needs. Having fewer data to relay means the sensor's battery will last longer. The ARIMA-based forecasting algorithm can create more accurate forecasts using more consistent temperature data for the next two hours.

In the explored system [3], the writer examines how WSN functions in IoT and Cloud environments. The goal of data collection proposed [4] is to disseminate data in mobile sensor networks with as little data cost and energy consumption as possible, and it is based on secure mobile sensor networks. It also ensures that collected data in the Cloud will always be accessible over the network and enhances authoritative routing. In the delivered method [5], the benefits and downsides of using Cloud-based framework engineering to monitor and analyze actual company performance are examined in detail. The proposed Cloud connecting architecture for WSNs carries out RESTful frameworks for efficient communications. The whole structure may be broken down into several different kinds of staggered monitoring and control systems, and it offers advantages in the areas of perception, information storage, and preparation for distributed computations.

The viability of the proposed framework is further examined, and an illustrated and reenacted way for

enabling flexible IPv6-based modern remote sensor organizations is shown. In [6], it is shown that a Cloud-based event-driven backbone may be utilized to cheaply and efficiently process data from several applications in real-time. The data center's Apache Kafka cluster and Zookeeper are split over three availability zones to provide the highest possible throughput and the lowest possible latency. The author of [6] investigates the potential of two-tier WSNs for use in the growth of a Cloud-based, IoT-based surveillance framework.

A ZigBee network connects the individual sensor nodes to the coordinators, while a separate Wireless RF network connects the coordinators to the base station. Information analyzed using fuzzy logic is stored and tracked utilizing a Cloud-based platform. [7] presents the IoT applications with SASC, which allows for greater network scalability without sacrificing data processing efficiency or security. In [8], the author proposes using the fuzzy Analytical Hierarchy Process to bring more order to the process of analyzing information security controls (AHP). Its purpose was to rate the importance of various information security procedures based on the priorities of a given firm. They assert that a more comprehensive and cost-effective examination of such controls is possible when using fuzzy AHP for the prioritization of information security measures.

In [9], the author aims to offer a model of security measures for Smart Cities via the use of the Green Internet of Things and Cloud Integrated Data Management (M-SMDM). As a first step, it employs self-balancing trees to spread load factors in green communication networks, resulting in a connection that is both resilient and resource-conscious. Second, dispersing secret keys among peer nodes eliminates the trust problem in both direct and indirect communication. Security is tightened up through the transmission chain, from mobile gateways to application users, while costs are cut down. In [10], the author developed a security model based on Elliptic Curve Encryption (ECC) and characteristics to ensure the full security of sensor data, including its privacy and integrity. Moreover, it enables fine-grained management of who has access to what.

The existing system [11] was explored to illuminate potential applications of cryptography in WSN to enhance data security. The goal of this essay is to educate the reader on a different method of data center security that relies on cryptography. The authors of [12] propose a solution to the security issue by recommending that data be encrypted and obfuscated at IoT devices before Cloud storage. Data from IoT devices is evaluated for its importance and treated accordingly, whether that be via encryption or obfuscation. The efficiency of the suggested method is measured by timing cryptographic operations on Internet-of-Things devices. To protect the privacy of its users

An enhanced identity-based encryption approach is provided by [13], in which a safe key is generated by employing part of an identifying bit string, making the encrypted data unreadable even if the key is decoded by an opponent or attacker. Experimental results demonstrate that when compared to a rival approach known as an efficient selective-ID secure identity-based encryption strategy, their method is noticeably quicker when it comes to both encrypting and decrypting data. The employment of the Lagrange coefficient, a polynomial interpolation function, to protect the privacy of the user makes this approach stand out. The computational difficulty of solving the bilinear Diffie-Hellman problem is a critical aspect of the system's security. In this research paper, we introduce the RSFSA [14] approach, which is a real-time service-centric feature sensitivity analysis. The RSFSA model measures how having access to different features at different levels of granularity affects the quality of any given service. This article introduces an efficient MSDAM, or multi-party secure data access management, that may be utilized by a wide range of parties for safe data sharing and access. The major goal of this technique is to improve coordination between the many organizations in charge of verifying requests to make changes to data. In addition, user-centric attributes are used to structure the distribution of the key set and scheme set for each user throughout each session using this method. In this method, the user's key sets and scheme sets are randomly assigned using a least-fact randomization algorithm. The technique uses a randomized

mechanism to choose the key and the methods used to encrypt the user's data using a system called User Centric Block Chain Encryption (UCBE) [15].

The hash code was similarly built using the Distancing method. The purpose of this research is to [16] propose and create a secure cryptographic protocol called CoopECC, which leverages the clustering of IoT nodes to distribute the responsibilities of being the cluster head (CH) among the nodes in the cluster. It has been shown that using this technique may lessen the computation and power requirements of IoT nodes. After running simulations using the TOSSIM program, it became clear that the proposed protocol, CoopECC, outperforms the original ECC algorithm in terms of computing efficiency, energy consumption, and network longevity.

The work [17] proposed a new energy-aware routing secured approach based on clustering, encryption, and trust modeling for secure data transmission in WSNs. To achieve this, it proposes an Elliptic Curve Cryptography-based encryption scheme. Before transmission throughout the network, the messages are encrypted using the approach they propose for secure routing. Starting on page [18], the author describes a WSN that utilizes Cloud computing to gather and analyze data from an array of sensors dispersed across the environment. The result is supposed to be a quick and secure network for sensor node data. To do this, the WSN will collect data from its immediate area, compress it, and then relay it back to the Cloud gateway, where it will be decompressed and stored in an extremely large Cloud server. In this article, we'll not only go over the WSN's capabilities but also the dangers it poses and the restrictions imposed by Cloud-computing services.

3. PROPOSED SYSTEM

The flexibility of the Cloud and wireless sensor networks have prompted many to herald them as a revolutionary stride forward in the area of computing. Many companies and organizations have considered the risks and drawbacks associated with adopting Cloud computing. In contrast to popular belief, wireless sensors are not autonomous nodes in the layer of wireless communication, but may instead be managed and kept up to date remotely via the use of Cloud services. As a result, an effective strategy was essential for resolving the current problems. The Cloud Hub (CH) requires information from Cloud-based apps like node ID, application location, and packet creation time before releasing any data. CH transmits requests from cluster nodes to the Cloud server.

The Cloud server should check the node's identity and location to make sure they are accurate. This method enables mutual authentication between the sink node and the Cloud server, in addition to the establishment of a communication key between the two. Figure 1 shows how the Cloud sends encoded mess messages pass to the sensor sink node, which in turn delivers the data to its intended recipient.

3.1 DO_LEACH

A distributed variant of LEACH uses a clustering hierarchical design in which nodes are separated into groups with their own CH to achieve the tasks normally handled by the nodes, and to promise a more dynamic form. With LEACH's data fusion technology and its capacity to form clusters locally, a network may be constructed on the fly via the dynamic selection of CHs, with every given node having an equal chance of being chosen as a CH at any given moment.

Figure 1. Representation of Proposed Methodology

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

CH compresses the information transmitted during the joining procedure between the nodes and the base BS to lower the total amount of data sent. Each of the numerous clusters in a WSN typically consists of a CH and sometimes additional nodes. If the base station (BS) is geographic if all the nodes are in the same network, then LEACH assumes that the sensor nodes will be able to interact with the sink nodes. It is important to remember that the DO LEACH model was developed so that the energy used by nodes while sending an n-bit data packet across a distance of d could be determined using the formula:

$$G_{Node}(m,a) = m[G_{elec} + \in_{Cons} a^2] \quad (1)$$

Where $G_{elec}$was the group of nodes chosen to create a CH and $\in_{Cons}$was the amount of power required to transmit a single bit of data. After CHs are established, nodes retain the option to act as CHs in later rounds, with the likelihood of doing so proportional to the number of rounds in which they have determined using equation (2), the node becomes a CH for the current round. "Since $1/x_i$rounds is less than or equal to the % Of CHs, the node will become a CH in less than or equal to the current round. When $x_i$is in the set of nodes $x_i \in K$that were not CHs in the previous $1/x$ round, then the corresponding equation holds".Elected nodes are represented as follows to form a CH and

$$TH(m_i) = \frac{x_i}{1-x_i\left(r \bmod 1/x_i\right)} \quad if\ x_i \in K \quad (2)$$

If a node is selected as a CH, it will begin broadcasting advertisements around the network to let other nodes know about its new role and sends it out to everyone in the cluster.

3.2 Path Scheduling

In order to ensure that data traveling from the SN to the DN has the greatest possible connection quality, it is recommended that we use an optimal routing path discovery to determine the most direct way. In this study, we employ the BICCO technique to prioritize the selected paths. The BICCO algorithm is a novel population-centered meta-heuristic that takes cues from the cuckoo's herding behavior. The essential features of the cuckoo-host system may be idealized as a population of n cuckoos with n nests, allowing for this streamlined presentation of the cuckoo search algorithm. Each cuckoo in a natural cuckoo-host system is able to effectively attack and lay its eggs in a vast number of host nests since each nest typically contains three to four eggs. Because we assume that each cuckoo may only affect a single host nest by laying a single egg, the total number of eggs is equal to the total number of nests and cuckoos. Therefore, in the answer to an optimization problem, the coordinates of an egg might be written as the vector x. These methods eliminate the need to differentiate between eggs, cuckoos, and nests. As a result, the adage "egg = cuckoo = nest" is accurate.

• Cuckoos only produce one egg, which they then deposit in the nest of an unselected host bird. For convenience, let's assume that the total number of eggs, nests, and cuckoos is the same.

• Each cuckoo egg has a p=0.0001 chance of being found and abandoned. This is the same as changing a fixed percentage pa of the population with each cycle t.

• A solution's or an egg's viability is measured by its objective value. The most optimal strategies are those with the smallest goal values (for minimization). Best answers are preserved in this way.

The cuckoo builds its nest in an area where the animals have a high chance of finding suitable grazing. Horses and sheep are picked at random in the BICCO algorithm's step size calculating procedure. The optimization suffers as a consequence of this random selection process. Levy flight selection, as opposed to a random nest selection, is used in the proposed work to circumvent this problem. That being said, these Initial steps include generating a cuckoo-filled collection of possible solutions. Equation 3 describes the resulting collection of nest solutions.

$$\varphi_j = \{\varphi_1, \varphi_2, \varphi_3, \ldots\ldots \varphi_m\} \quad (3)$$

"Herein $\varphi_j(m = h \times s)$stands for the number of cuckoo birds, while m represents the nest's solution set.; h represents the total number of individuals in a given group, while s stands for the total number of nests.. Then, the equation describing the starting location of a cuckoo in f-dimensional space"

$$\varphi_j^o = \varphi_{min} + \hat{l}o(\varphi_{max} - \varphi_{min}) \quad (4)$$

"Herein, φ_j^orepresented j^{th} cuckoo's initial solution φ_{max} and $\varphi_{min}$are the bound of design variables, o implies the element-by-element multiplications and $\hat{l}$ is the random vector $\hat{l} \in [0,1]$". After that, a total is made of how well each solution fits the problem, and all viable options are given due consideration. Taking into account the costs and benefits of each solution's path trust, EC, and path distance is done using the equation below (5).

$$F_{opt} = \begin{cases} max\left(\sum_{l=1}^{d} QN(L_l, L_{l+1})\right); & path\ trust \\ max\left(\frac{1}{L_l}\sum C_{res_l}\right); & residual\ energy\ of\ each\ node \\ min\left(\sum_{l=1}^{d} dist(L_l, L_{l+1})\right); & path\ distance\ between\ lsource\ and\ destination \end{cases}$$

(5)

"Herein QN(L_s, L_{s+1}) and $dist(L_l, L_{l+1})$ signify the paths trust and path's distance betwixt node L_l and L_{l+1}correspondingly; $L_l$implies SN; d signifies the DN; C_{res_l}implies the nodes residual energy".

Solution values are ranked from highest to lowest based on their fitness values. Assimilate the cuckoo into the group you're trying to form. It's standard practice for each group to choose their nests in descending order of preference. The chosen eggs are known as standard eggs, and the nest that includes the efficient fitness common is known as a herd. As in L'evy flight equations, the size of each cuckoo's nest, S_{s_j} is determined by randomly selecting one of the cuckoos and their nests.

$$S_{s_j} = \delta^* L^y o(\varphi_d - \varphi_j) + \bar{\delta}^* L^y o(\varphi_k - \varphi_j)$$

(6)

$$L^y = t(-y), \quad 1 < y < 3$$

(7)

L´evy ~ u=$L^y = L^y$

(8)

Herein $\varphi_d$and $\varphi_j$represent the cuckoo and the selected nestL^yw and v stand for aspects of exploration and exploitation management; they also imply the levy flight distribution. The steps here constitute a random walk process whose step-length distribution follows a power law and whose tail is rather long. Some of the new solutions may be generated quickly via a local search by doing an L'evy walk around the greatest solution identified so far. Distant field randomization should be used to generate a large fraction of the new solutions, with locations sufficiently far from the present best solution, to avoid the system from being trapped in a local optimum.

$$\delta = \delta_0 + \frac{\delta_{MAX} - \delta_0}{C_{MAX}} \times C$$

(9)

$$\bar{\delta} = \bar{\delta}_0 + \frac{\bar{\delta}_0}{C_{MAX}} \times C$$

(10)

"Herein, I imply the iteration; C_{MAX} means the highest possible repetition count. To achieve effective optimization, this algorithm first performs high-quality exploration in the first stages and then improves upon exploration with higher-quality exploitation in the later stages. Next φ_j's new position is enumerated as in the equation"

$$\varphi_j^{"} = \varphi_j^0 + S_{s_j}$$

(11)

"If the fitness value of $\varphi_j^{"}$ isn't worse analogized to φ_j^0 fitness value φ_j position is updated. Similar to how each new route is compared to its predecessor, fitness is considered". This procedure will be repeated until the best possible solution is found. The pseudo-code for the BICCO algorithm's optimum route selection is shown in the following algorithm 1.

3.3 Data Security

Data packets (DPs) are encrypted to increase their security against assaults before being sent from a sink node (SN) to a destination node (DN). The primary goal of transforming DPs into encrypted information is to prevent unwanted access. The suggested method uses the blowfish (BF) algorithm. The CPBFC method is used to provide multi-level security in WSN communication. The CPBFC technique is only one of several that may be used to trick brute-force hackers. It will prevent intruders by providing false information at each failed attempt to crack the key combination. The approach generates a single key, the CPBFC key generation methodology. The CPBFC key is used to generate a second, secret key, which is intended to increase the system's already high degree of security. Primarily, '2' prime numbers are chosen at random for use in BFA's CPBFC key generation process. The product q may be found by thinking of the prime integers 2 as z and y and solving the resulting equation.

$$R = z \times y$$

(12)

"Next, choose '1' integer g that must be greater than '1' and less than (z-1) and (y-1). Then, the definite pair of integers P and g forms the CPBFC

Journal of Theoretical and Applied Information Technology
31st December 2022. Vol.100. No 24

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

key, which is denoted as g'. Then, the CPBFC secretkey $g'_s$is enumerated as of the integers z,y and $g_s$as in equation",

$$g_s = g^{-1}(z-1)(y-1) \quad (13)$$

When the key has been generated, the CPBFC procedure begins. Let's assume Q n is the input DP.

$$\delta \rightarrow \alpha DTE(q_n) \quad (14)$$

Here, α implying a uniform random selection was used. Using the CPBFC key and secret key, the seed is encoded using a traditional encryption method to generate the cipher text matching to the input as NQ^{Q_n} in equation (15)

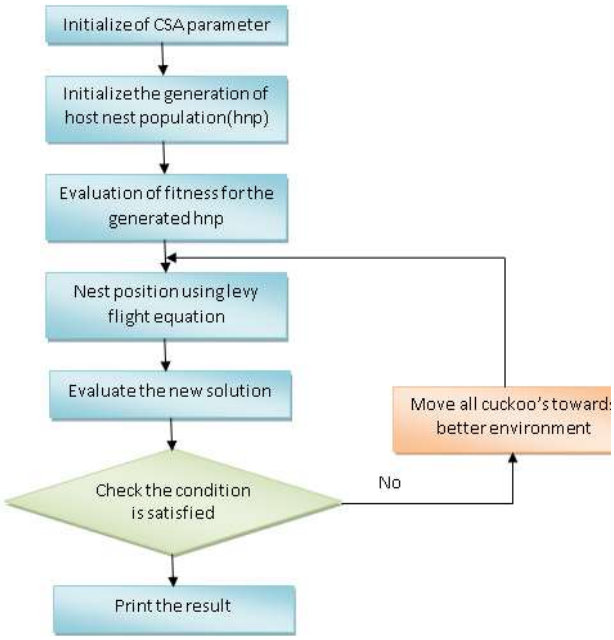

Figure 2 Process Of CPBFC

$$\hat{\delta} = F(q, g) \quad (15)$$

"Here $\hat{\delta}$signifies the encrypted seed; ρimplies the random string; F symbolizes the cryptographic hash function. Lastly, the cipher-text is attained utilizing seed and encrypted seed as in equation",

$$X(Q_n) = \left(\hat{\delta} \oplus \delta\right) + g_s (16)$$

Herein, $X(Q_n)$ signify the DP Q_n's ciphertext. During encryption, the sender side (authorized users) is given access to the random string, encrypted seed, CPBFC key, and secret key. As long as the random string, encrypted seed, CPBFC key, and secret key are kept secure, the original DP may be rebuilt. The SN passes the cipher text onto the next node in the data path, where the DP is expected to be sent. However, before the cipher text can be delivered securely, a secured routing path must be determined between SN and DN. Finally, after the data gets encrypted the data can be transmitted to the destination node.

Algorithm 1 DO LEACH

Input: LO_LEACH

Output: Secured integration based

Begin

Clustering ()

Data compression ()

Path scheduling ()

Initialize the candidate solution φ_j randomly

Determine the initial position using,

$$\boldsymbol{\varphi_j^o = \varphi_{min} + \hat{l}o(\varphi_{max} - \varphi_{min})}$$

Evaluate fitness for each solution using,

$$F_{opt} = \begin{cases} max\left(\sum_{l=1}^{d} QN(L_l, L_{l+1})\right); & path\ trust \\ max\left(\frac{1}{L_l}\sum C_{res_l}\right); & residual\ energy\ of\ each\ node \\ min\left(\sum_{l=1}^{d} dist(L_l, L_{l+1})\right); & path\ distance\ between\ lsource\ and\ destination \end{cases}$$

While I=0 to C_{MAX} do

Arrange the answers from highest to lowest. based on F_{opt} and form group

A step size calculator for each shepherd may be found at

$$S_{s_j} = \delta^* L^y o(\varphi_d - \varphi_j) + \bar{\delta}^* L^y o(\varphi_k - \varphi_j)$$

Generate new element using,

$\varphi_j^{"} = \varphi_j^0 + S_{s_j}$

if $\left(F_{opt}(\varphi_j^l) \geq F_{opt}(\varphi_j^0)\right)$

Nest position =φ_j^n

Else

Nest position =φ_j^0

End

While

Return optimal routing path

Data encryption phase(CPBFC);

Data transmission phase

End.

4. PERFORMANCE ANALYSIS

The novel optimization approach is used in MATLAB simulations. More testing is needed before we can have faith in the results. As a result, determining whether or whether the strategy proposed here is effective is crucial. The feasibility of the solution and the estimated results are tested against the parameters of the computation. Table 1 displays the simulation settings used in testing the proposed system.

Table 1. Simulation Parameters

Parameter	Value
"No. of Nodes"	200
"No. of round cycle"	9000.00
"Initial energy"	00.1
"Energy transmission"	80*0.000000001
"Energy sampling"	50*0.000000001
"Energy amplitude"	0.0018*0.000000000004
"Energy aggregate"	4*0.000000001
"Size of the packet"	512 Bytes
"Position of the sink"	Random

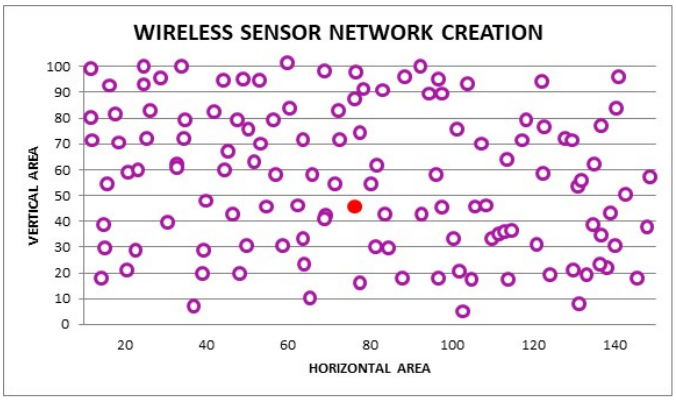

Figure 3 Implemented System Model.

Figure 3 depicts a WSN network model that may be used to start off the process. Here, we include a broad range of novel approaches to current studies. Total transmission time, packet delivery rate, encryption and decryption times, and key generation times were all measured to see how well the approach performed in comparison to others. The following sections elaborate on a number of indicators of success.

End-to-End Delay (ETED): ETED is the total amount of time spent on the network to move a DP from SNs to DNS. Formulation of the ETED

$$ETED = U'_{sP} - U'_{rP} \quad (17)$$

"Here, U'_{sP} implies the packet generation time in SN; $\boldsymbol{U'_{rP}}$implies the packet received time in the DN".

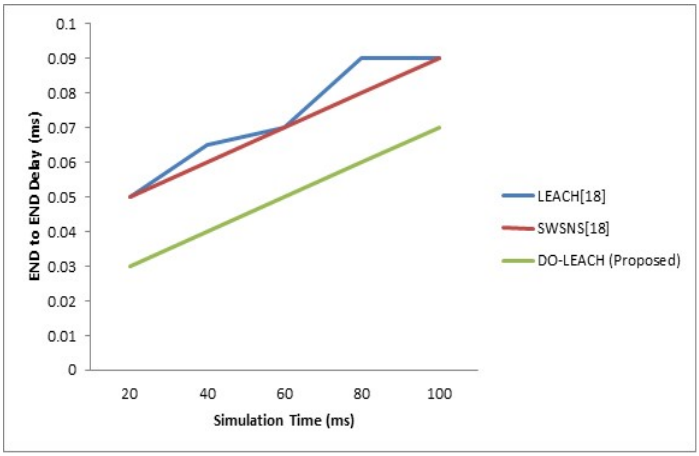

Figure 4 Simulation Time Vs. End-To-End Delay

An increase in nodes, from 20 to 200, often has a linear effect on this latency. The buffering that occurs at the nodes' communication levels also contributes to the latency. The end-to-end latency for the DO LEACH protocol ranges from around 0.07 milliseconds for 20 nodes to about 2.03 milliseconds for 200 nodes. EESRA protocol causes an increase in end-to-end latency of nearly 25 percent, from 2.17 milliseconds to 3.45 milliseconds for the same number of nodes. If you have up to 200 nodes, the LEACH protocol's end-to-end latency is just 0.0564 ms. Therefore, it becomes clear that the proposed safe routing protocol may reduce energy consumption at nodes without jeopardizing confidentiality.

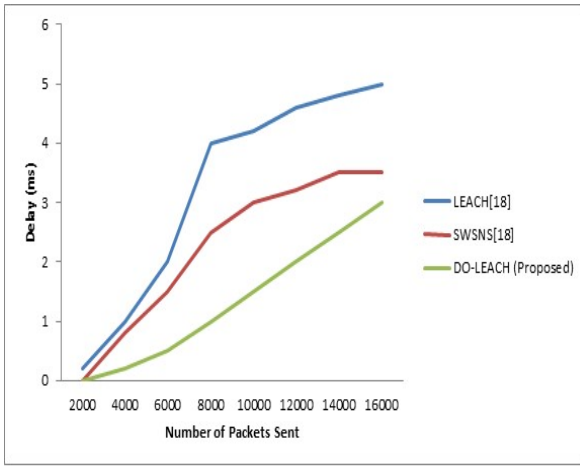

Figure 5 Number Of Packets Sent Vs. Delay

Figure 4 depicts the delay analysis done on the proposed DO LEACH in contrast to the existing routing algorithms known as LEACH and the EESRA by delivering between 2,000 and 16,000 packets from the sender to the receiver. Figure 8 shows that the suggested DO LEACH algorithm has worse delay reduction performance than the state-of-the-art energy efficient routing approaches.

Packet delivery ratio (PDR) :"It defines the ratio betwixt the number of DPs supplied to the DN $L_{S(SP)}$

And the DP's transmitted by the SN ($L_{S(SP)}$) as in equation (18)"

$$P_{DR} = \frac{L_{r(DP)}}{L_{S(SP)}} \times 100 \quad (18)$$

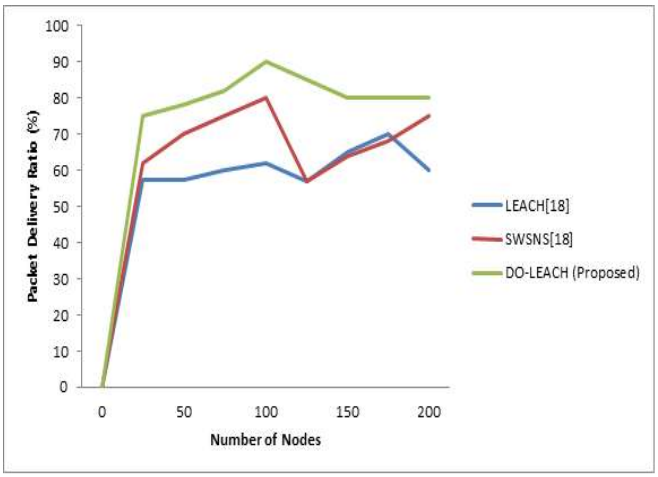

Figure 6. Packet Delivery Ratio Vs Number Of Nodes

Figure 6 depicts the percentage of successfully delivered packets based on the number of nodes involved while utilizing the DO LEACH routing protocol without any kind of cryptosystem model. When the suggested model was applied, however, it became apparent that the batteries of some of the nodes were being depleted at an alarming rate. This led to a delivery ratio of around 87.5 packets, which was lower than the ratios achieved by the previous SWSNS and LEACH for 200 nodes. It's worth noting that extra power is needed for data compilation and transmission to the BS. The suggested model features a reduced packet delivery ratio and an acceptable number of active nodes. DO-LEACH was put to the test, and it was shown to enhance things by roughly 90% if a security protocol already existed.

Figure 7 shows the results of research comparing the times required to generate keys for both the unencrypted and encrypted versions of a given set of data. The time needed to generate keys for the proposed CPBFC encryption method is much shorter than that needed for the present method. This is because both shorter-lived keys and keys that can only be used once have been developed, increasing security without increasing key size.

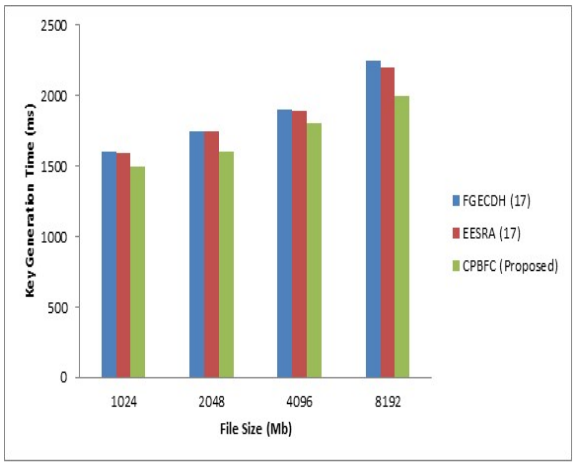

Figure 7 File Size Vs. Key Generation Time

Decryption Time (U_d): It denotes the amount of time that is spent with the intention of deciphering the cipher text in order to retrieve the original message.

$$U_d = d(U)_{end} - d(U)_{start}$$
(19)

"Here, $d(U)_{end}$ signifies the decryption ending time; $d(U)_{start}$ implies the decryption starting time".

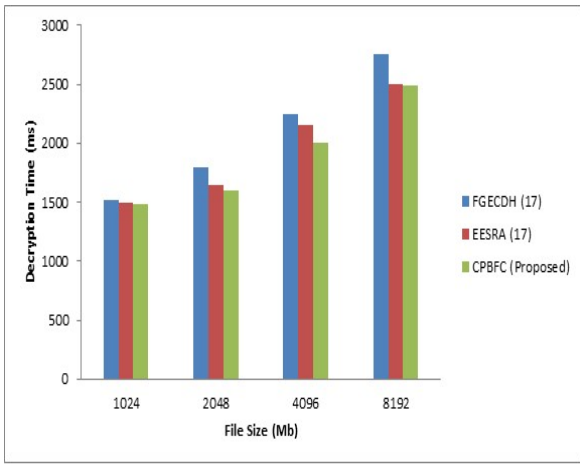

Figure 8 File Size Vs. Decryption Time

Evidence of this is seen in the fact that the suggested model clocked in at a lower total time for encrypting and decrypting than the previous best method. Decoding a message using the suggested method took much less time than using the conventional methods.

Encryption Time (U_e): Specifically, it represents the effort put into transforming the plaintext message (DPs) into cipher text (using the equation above).

$$U_e = e(U)_{end} - e(U)_{start}$$
(20)

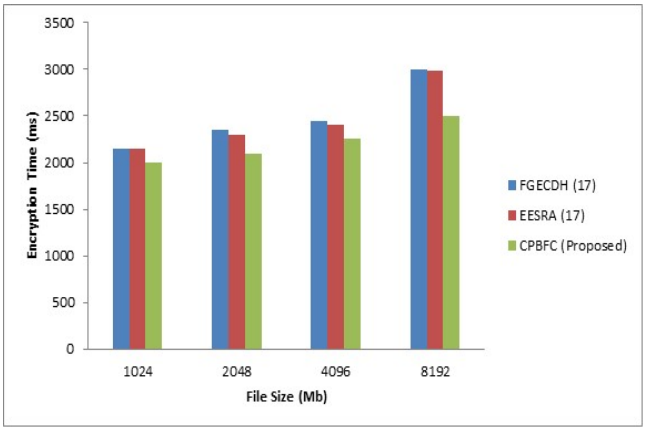

Figure 9 File Size Vs. Encryption Time

It has been shown, with the help of figure 9, that the CBBFC-based encryption model that was developed takes much less time to carry out the encryption procedures than the symmetric key technique that is now in use. From the result obtained it was revealed that the suggested mechanism outperforms well than the existing mechanisms.

5. CONCLUSION

Constraints on WSNs and Clouds come in the form of attacks, limited storage space, insufficient computing power, and high energy costs. Therefore, it is crucial to develop a multipath protocol that is both low-power and resistant to common attacks. Therefore, the proposed approach, a WSN and Cloud integration was accomplished by using the DO LEACH protocol, which comprises BICCO and CPBFC to provide efficient and secure communication inside the WSN. The suggested method makes use of a BICCO in order to find an optimum routing route that is designed to be efficient in terms of energy consumption and provide secured routing. In order to increase the DPs' degree of protection against assaults and to

facilitate the integration of Cloud and WSN infrastructure, a CPBFC has been modified. As a result of the performance evaluation, it has become abundantly clear that the methods that have been suggested are very efficient and safe when compared to the other methods that are now in use. As a result, the suggested routing strategy was determined to be very efficient in terms of energy use and secure enough to withstand a variety of assaults. In the future, work on this system may include the incorporation of intrusion detection systems (IDS), which are capable of being installed at network nodes and used to improve secure communication in order to facilitate the effective identification and isolation of attackers.

REFERENCES

[1] R. K. Dwivedi and R. Kumar, "Sensor Cloud: Integrating wireless sensor networks with Cloud computing," in *2018 5th IEEE Uttar Pradesh section international conference on electri*cal, electronics and computer engineering (UPCON), 2018, pp. 1-6.

[2] K. Das and S. Das, "Energy-Efficient Cloud-Integrated Sensor Network Model Based on Data Forecasting Through ARIMA," International Journal of e-Collaboration (IJeC), vol. 18, pp. 1-17, 2022.

[3] M. J. Therese, P. Dharanyadevi, and K. Harshithaa, "Integrating IoT and Cloud Computing for Wireless Sensor Network Applications," Cloud and IoT-Based Vehicular Ad Hoc Networks, pp. 125-143, 2021.

[4] K. Haseeb, Z. Jan, F. A. Alzahrani, and G. Jeon, "A Secure Mobile Wireless Sensor Networks based Protocol for Smart Data Gathering with Cloud," Computers & Electrical Engineering, vol. 97, p. 107584, 2022.

[5] Bharathiraja, N., & Kumar, P. S. (2016). Service Oriented Architecture for an Efficient Automation of Sensor Networks Data on Cloud with Internet. Asian Journal of Research in Social Sciences and Humanities, 6(12), 1192-1203.

[6] S. Khriji, Y. Benbelgacem, R. Chéour, D. E. Houssaini, and O. Kanoun, "Design and implementation of a Cloud-based event-driven architecture for real-time data processing in wireless sensor networks," The Journal of Supercomputing, vol. 78, pp. 3374-3401, 2022.

[7] K. Haseeb, A. Almogren, I. Ud Din, N. Islam, and A. Altameem, "SASC: Secure and authentication-based sensor Cloud architecture for intelligent Internet of Things," Sensors, vol. 20, p. 2468, 2020.

[8] M. I. Tariq, S. Ahmed, N. A. Memon, S. Tayyaba, M. W. Ashraf, M. Nazir, et al., "Prioritization of information security controls through fuzzy AHP for Cloud computing networks and wireless sensor networks," Sensors, vol. 20, p. 1310, 2020.

[9] A. Rehman, K. Haseeb, T. Saba, and H. Kolivand, "M-SMDM: a model of security measures using Green Internet of Things with Cloud integrated data management for smart cities," Environmental Technology & Innovation, vol. 24, p. 101802, 2021.

[10] M. Saran, R. K. Dwivedi, and R. Kumar, "Attribute-Based Elliptic Curve Encryption for Security in Sensor Cloud," in Advances in Data and Information Sciences, ed: Springer, 2020, pp. 441-453.

[11] Bharathiraja, N., Padmaja, P., Rajeshwari, S. B., Kallimani, J. S., Buttar, A. M., & Lingaiah, T. B. (2022). Elite Oppositional Farmland Fertility Optimization Based Node Localization Technique for Wireless Networks. Wireless Communications and Mobile Computing, 2022. https://doi.org/10.1155/2022/5290028.

[12] D. Anurekha, N. Thirugnanasambandan, and D. Rajivkannan, "Enhancing Security of Neurological Health Information Using Cryptography in Wireless Sensor Network," NeuroQuantology, vol. 20, p. 96, 2022.

[13] R. K. Gupta, K. K. Almuzaini, R. Pateriya, K. Shah, P. K. Shukla, and R. Akwafo, "An improved secure key generation using enhanced identity-based encryption for Cloud computing in large-scale 5G," Wireless Communications and Mobile Computing, vol. 2022, 2022.

[14] Bharathiraja, N., Shobana, M., Manokar, S., Kathiravan, M., Irumporai, A., & Kavitha, S. (2023). The Smart Automotive Webshop Using High End Programming Technologies. In Intelligent Communication Technologies and Virtual Mobile Networks (pp. 811-822). Springer, Singapore. https://doi.org/10.1007/978-981-19-1844-5_64

[15] T. Arulananth, M. Baskar, V. Anbarasu, R. Thiagarajan, T. Rajendran, and A. Balaji, "Multi party secure data access management in Cloud using user centric block chain data

encryption," Pattern Recognition Letters, vol. 152, pp. 295-301, 2021.

[16] W. Jerbi, A. Guermazi, O. Cheikhrouhou, and H. Trabelsi, "CoopECC: a collaborative cryptographic mechanism for the internet of things," Journal of Sensors, vol. 2021, 2021.

[17] S. Viswanathan, R. Bhuvaneswaran, S. Ganapathy, and A. Kannan, "Euler Phi Function and Gamma Function Based Elliptic Curve Encryption for Secured Group Communication," Wireless Personal Communications, pp. 1-31, 2022.

[18] Sasikumar, B., Naveenraju, D., Anand, K., Hariharan, S., Sudhakaran, P., & Bharathiraja, N. (2020). Diabetes prediction using sensors by analysing skin temperature. J. Eng. Sci. Technol, 15(2), 1357-1370. .