The Scientific Temper (2024) Vol. 15 (3): 2672-2682
Doi: 10.58414/SCIENTIFICTEMPER.2024.15.3.35

E-ISSN: 2231-6396, ISSN: 0976-8653
https://scientifictemper.com/

RESEARCH ARTICLE

A secured routing algorithm for cluster-based networks, integrating trust-aware authentication mechanisms for energy-efficient and efficient data delivery

Annalakshmi D*., C. Jayanthi

Abstract

Secure routing in wireless sensor networks (WSNs) is vital for preserving data veracity and privacy in the face of possible threats. Traditional routing protocols lack robust security mechanisms, making WSNs vulnerable to attacks. Secure routing protocols in WSNs aim to address these vulnerabilities by implementing authentication, encryption, and intrusion detection techniques to ensure secure and reliable data transmission while minimizing energy consumption. This paper proposes a novel secured routing algorithm tailored for cluster-based networks aimed at enhancing energy efficiency and data delivery security by integrating trust-based authentication mechanisms. The approach begins with the design of a clustering algorithm, which organizes network nodes into clusters based on proximity or network topology. Subsequently, a trust-based authentication mechanism is developed to evaluate the reliability and integrity of both nodes and links within the network. Building upon these foundational elements, a secured routing protocol is devised to capitalize on cluster-based organization and trust-based authentication, thereby facilitating energy-efficient and secure data transmission. The proposed algorithm and authentication mechanism cluster and optimal routing assisted cryptograph (CORAC) are implemented within a simulated network environment to validate their efficacy. Performance evaluation is conducted through simulation studies, focusing on key metrics such as packet delivery ratio, energy consumption, and security effectiveness. This comprehensive approach aims to address the dual challenges of energy efficiency and data security in cluster-based networks, offering a promising solution for future deployments in various applications.

Keywords: Clustering, Optimal routing, Secured WSN, ECC, Data transmission, Energy consumption.

Introduction

In the contemporary digital context, the proliferation of data communications *via* the internet has reached unprecedented levels. A myriad of applications, ranging from online banking to mobile communication and smartcards, underscores the paramount importance of security in environments where resources are limited. Among the technologies driving this paradigm shift are wireless sensor networks (WSNs), which play a pivotal role in collecting data from the environment to power essential services across various domains, including healthcare, agriculture, and military operations. However, the security of data transmitted through WSNs poses a significant challenge that must be addressed for the successful deployment of sensor network-based applications Qazi, R., (2021); Tropea, M., (2022).

One promising cryptographic technique that addresses the security needs of WSNs is elliptic curve cryptography (ECC). ECC offers several advantages over traditional public key algorithms, including shorter key sizes and greater computational efficiency. Given the resource constraints inherent in WSNs, ECC emerges as a suitable choice for ensuring secure communication while minimizing the computational burden on sensor nodes. However, to meet the ever-increasing demand for speed in modern applications, it is essential to develop effective key management techniques that can accommodate hardware acceleration without compromising overall network performance Mallick, B. B., (2021).

Effective key management involves reducing the key size of public key cryptographic algorithms to facilitate rapid key computation and encryption. Additionally,

PG & PG and Research Department of Computer Science, Government Arts College (Autonomous) Affiliated to Bharathidasan University, Tiruchirappalli), Tamil Nadu, India.

***Corresponding Author:** Annalakshmi D, PG & PG and Research Department of Computer Science, Government Arts College (Autonomous) Affiliated to Bharathidasan University, Tiruchirappalli), Tamil Nadu, India., E-Mail: poorna23.priya@gmail.com

How to cite this article: Annalakshmi, D., Jayanthi, C. (2024). A secured routing algorithm for cluster-based networks, integrating trust-aware authentication mechanisms for energy-efficient and efficient data delivery. The Scientific Temper, **15**(3):2672-2682.

Doi: 10.58414/SCIENTIFICTEMPER.2024.15.3.35

Source of support: Nil

Conflict of interest: None.

Received: 10/07/2024 **Accepted:** 10/08/2024 **Published** : 10/09/2024

integrating additional mathematical functions into the key generation process can enhance security without unduly burdening the network. By leveraging ECC and optimizing key management techniques, researchers aim to strike a balance between security and performance in WSNs, thereby enabling secure and efficient data transmission Mohindru, V., (2020); Gupta, S. C., (2021).

Wireless network security encompasses safeguarding wireless networks from unauthorized usage and malicious access. In the context of WSNs, secure routing algorithms play a crucial role in ensuring the integrity and confidentiality of transmitted data. Congestion-aware routing algorithms are particularly important for optimizing network performance and resource utilization. By dynamically adapting routing decisions based on network conditions, congestion-aware routing algorithms can mitigate bottlenecks and enhance overall efficiency Pooja & Chauhan, R. K. (2022).

WSNs consist of spatially dispersed sensor nodes tasked with monitoring and recording the physical environment. These nodes communicate data to a central storage and processing location, facilitating real-time analysis and decision-making. Despite their small size, sensor nodes are multifunctional and can communicate effectively over short distances, making them ideal for a wide range of applications, including patient health monitoring, environmental observation, military surveillance, and forest fire monitoring Ravi, K., (2020), Mirvaziri, H., (2020), Gulen, U., (2020).

The holistic view of security algorithms in WSNs encompasses several key categories, including cryptographic algorithms, key management techniques, secure routing algorithms, secure data aggregation methods, intrusion detection systems, and trust management techniques. By addressing these aspects comprehensively, researchers aim to fortify WSNs against a wide range of security threats, including eavesdropping, data tampering, and denial-of-service attacks. The design requirements for security in WSNs are multifaceted and encompass various dimensions, including confidentiality, integrity, availability, self-organization of nodes, secured localization of data, time synchronization, and authentication. These requirements are essential for ensuring the robustness and reliability of WSNs in the face of evolving security challenges. Vivek, K. (2021, November), Morales-Sandoval, M. (2021), Shah, P., (2020), Ganesan Sangeetha., (2020), Xiao, Y, (2007).

The security of data communicated through WSNs is a critical issue that must be addressed to enable the successful implementation of sensor network-based applications across diverse domains. By leveraging technologies such as ECC and developing innovative key management techniques, researchers aim to enhance the security and efficiency of WSNs, thereby unlocking their full potential for transformative applications.

Related Works

Wireless network security safeguards a wireless network against unauthorized use and unwanted access. Routing is crucial in computer networks, and congestion-aware routing algorithms are essential for improving network performance (Ganesan Sangeetha *et al*., 2020). Wireless network security is usually ensured by wireless networking equipment like routers and switches, which use encryption to protect all wireless network traffic conducted over ad hoc and sensor networks. A WSN is comprised of a collection of geographically distributed and specialized devices known as sensors, which are used for efficient monitoring and data collection of the physical environment. Furthermore, WSN is responsible for arranging the collected data in a centralized way of storage and processing site. WSNs are comprised of versatile sensor nodes that are compact and capable of more efficient communication over limited distances. The WSN is used in many applications, such as patient health monitoring in the medical industry, environmental observation via intrusion detection systems, military surveillance, and forest fire monitoring. The comprehensive perspective on security algorithms may be categorized into six main groups: cryptographic algorithms, key management techniques, secure routing algorithms, secure data aggregation methods, intrusion detection systems, and trust management approaches. The security measures in WSN safeguard communication by protecting transmitted information, safeguarding resources from assaults, and preventing hostile node behavior. Key security design needs in WSN include confidentiality, data integrity, availability, self-organization of nodes, safe data localization, time synchronization, and authentication. Recently, three primary forms of public-key cryptosystems have been prominent for their security and efficiency.The first practical implementation, termed RSA after the creators, was developed at the Massachusetts Institute of Technology (MIT) labs. This historical progression shows the importance of RSA and ECC in contemporary cryptographic applications Xiao *et al*., (2007).

Key algorithms in public key cryptography include ElGamal cryptography, RSA algorithm, and the Diffie-Hellman key exchange algorithm. The ECC was later refined and has now become a standard for implementing security systems employing public key algorithms. Elliptic key cryptography is widely used in modern applications because of its compact key size and enhanced security features. This research introduces a comprehensive security key distribution strategy using asymmetric cryptography technology for WSNs to overcome the resource constraints of sensors. The system guarantees reciprocal authentication using a challenge-response method with modest complexity, enhancing security while minimizing storage overhead and key exposure hazards.

Energy consumption and security efficiency are still major obstacles in WSNs. This paper examines the dynamic cluster head (DynCH) approach to mitigate power consumption in mobile WSN nodes. DynCH enhances network longevity by 45% in comparison to constant clustering methods, as shown by the results. The paper assesses lightweight systematic block encryption techniques, including Speck128, FlexenTech, tiny encryption algorithm (TEA), and advanced encryption standard (AES), showcasing their effects on energy consumption and network longevity in WSNs Cheng, Y., (2023), Abu-Ain, T., (2021).

Security is essential in WSNs because of their limited hardware resources. A technique for safe text encryption that is lightweight, energy-efficient, and utilizes a dynamic salt key is suggested. This paradigm improves data security using limited communication and computing resources, creating a secure environment for sensors to effectively safeguard data before transmitting it across wireless networks Elamurugu, V., (2021).

Encryption is crucial for security in WSNs, including both asymmetric and symmetric encryption techniques. This study assesses the KCMA technique for creating chain keys in ECC, RSA, and ElGamal algorithms, together with SHA2 and XOR hash functions. Diehard tests are used to evaluate the unpredictability of produced secret keys, with SHA2 demonstrating superiority. A performance assessment is carried out based on system time and network throughput Hamza, A. H., (2021).

Research into wireless body-area sensor networks (WBASN) has gained significant importance in medical applications, particularly in patient monitoring. However, routing remains a resource-intensive activity in WBANs, necessitating the development of energy-efficient routing systems. Existing routing algorithms prioritize energy efficiency over security, potentially leading to increased energy consumption due to security attacks. To address energy efficiency, reliability, and security concerns in WBANs, a new cluster-based secure routing protocol called secure optimal path-routing (SOPR) has been proposed. Similarly, in the context of the wireless sensor network in the internet of things (WSN-IoT), security remains a critical concern. While WSN-IoT networks are transforming various aspects of life, security threats such as sniffing, spoofing, and intruders persist. However, limited research has been conducted on security methods for WSN-IoT networks Dass, R., (2023), Hussain, M. Z., (2023).

Recently, a study evaluated the security mechanisms of the Routing protocol for low power & lossy network (RPL) using a partial implementation in the Contiki operating system. This critical analysis underscores the need for robust security mechanisms in WSN-IoT infrastructures and highlights the potential of machine learning for network management. This model integrates intrusion detection, decision-making mechanisms, and energy-aware routing algorithms to achieve superior detection accuracy and reduced energy consumption compared to existing intrusion detection models. Overall, these research endeavors emphasize the importance of developing secure and energy-efficient routing protocols to safeguard IoT-enabled networks from security threats while optimizing performance Aruchamy, P., (2023).

In the realm of wireless network security, several research gaps persist, particularly concerning WSNs and IoT-enabled systems. One significant gap lies in the limited exploration of security methods tailored specifically for WSN-IoT networks despite their transformative impact across various domains. These networks face persistent security threats such as sniffing, spoofing, and intrusions, necessitating tailored security measures. Additionally, energy consumption and security efficiency remain considerable challenges in WSNs, requiring lightweight encryption techniques and energy-aware security mechanisms to mitigate resource constraints. Moreover, efficient and secure key distribution strategies are essential but inadequately addressed, further compounded by the need to integrate machine learning for enhanced network security. Cluster-based secure routing, such as the proposed SOPR protocol, aims to address these challenges by enhancing network performance while mitigating security risks. By organizing nodes into clusters and employing secure routing mechanisms, cluster-based approaches optimize energy usage, improve reliability, and safeguard data transmission, thereby rectifying the identified research gaps and advancing the security and efficiency of wireless networks.

Proposed Methodology

A secured routing algorithm for cluster-based networks, integrating trust-aware authentication mechanisms

The work presents a secure routing system that utilizes clusters and trust to guarantee the safe transmission of data from sensor nodes to the base station. The decision manager chooses an encryption technique from the research, such as ECC with beta and gamma functions, ECC with gamma functions, and ECC with unicode, to encrypt the collected data. The encrypted data is sent to the base station using the energy-efficient, trust-based secure routing mechanism described in this paper.

Cluster Construction

Initially, cluster formulation is initiated during routing, and the D (cluster head) is allocated. The cluster formulation process has started, and the ensuing cluster formulation is as outlined:

$$\sum_{D\in\alpha} b_{snd} = 1, \forall\ sn \in SN \text{---------(1)}$$

Within this context, the cluster head formation is denoted as b_{snd}, the transmission link is denoted as SN, and the sensor node is denoted as SN.

The intra-routing route is created after the cluster is established, linking the source location to the CH. The routing establishment is defined as in Equation 2:

$$b_{DSD} = \sum_{cp \in CP_{DSD}} t_{cp}\,, \forall ds \in DS; D \in \alpha \text{ ---------(2)}$$

In the intra-routing route, TCP indicates the dependent variable. All potential pathways CP_{DSD} provide the path cp inside CP_{DSD} for each pair from the DS to D. During the solution phase, a specific route is identified within the network, and the appropriate path is selected from a predefined list. The data originating from various sources is collected and aggregated at the sink node. Inter-routing connections are then established to link the sink node with the cluster head (designated as D) of each source node in the network.

$$b_{DSD} = \sum_{p \in P_D} \eta_p\,, \forall ds \in DS; D \in \alpha \text{ ---------(3)}$$

The choice variables, η_q, are assigned a value of 1 when the inter-routing route, represented as p, η_q is chosen for transmission with the value 1, and 0 otherwise.

Figure 1 displays G1 to G6 as cluster heads, with k serving as the sink and a total of 21 nodes organized into groups.

Intra-routing and its constraints

Each cluster head verifies the conjunction in the constraint and also has a unique intra-routing path to the DS from each cluster. Equation 4 details the formation of the connection, whereas equation 5 specifies the transmission restriction.

$$\sum_{D \in \alpha} \sum_{cp \in CP_{DSD}} t_{cp}\,, \forall ds \in DS \text{---------(4)}$$

$$\sum_{D \in \alpha} \sum_{cp \in CP_{DSD}} t_{cp}\, \delta_{cp(x,y)} \leq J_{(x,y)}, \forall ds \in DS; (x,y) \in M \text{---------(5)}$$

where J(x,y) denotes the link (x,y) on the tree, with the intra-routing decision variable set to 1.

Inter-routing and its constraints

Sensor node D is chosen as a cluster head, and an intra-routing path with a cost of 0 is established from the source to the sink node. The identified route in the network is assigned a value of 1 to denote its potential. If there is no route, then the value of $_p$ is 0.

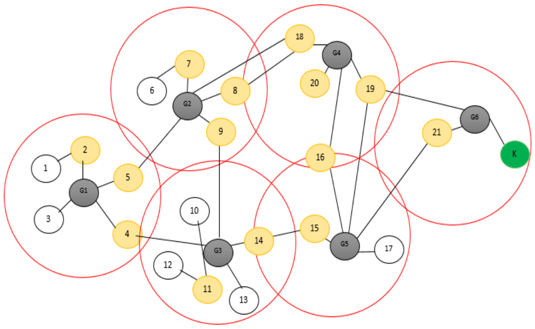

Figure 1: Cluster formulation

$$\sum_{p \in P_D} \eta_p \leq 1, \forall D \in \alpha \text{ ---------(6)}$$

The decision variable J(x,y) is set to 1 to indicate the selection of a route in the network and the cluster head indication is represented by $\delta_{q(x,y)}$ when a link is chosen in the network, equation 7 outlines the constraints on these two pathways. The decision variable J(x,y) is set to 1 when a route on the network is chosen and the cluster head indication is $\delta_{q(x,y)}$, while a link is selected in the network. Equation 7 describes the restrictions on these two pathways.

$$\sum_{p \in P_D} \eta_p \delta_{p(x,y)} \leq J_{(x,y)}, \forall D \in \alpha, (x,y) \in M \text{---------(7)}$$

If the cluster head is situated along the route, any sensor node within that route can serve as an intermediary node for the inter-cluster routing path. The selection process for a node is detailed in equation 8.

$$\sum_{p \in P_D} \eta_p . \delta_{psn} \leq \frac{\sum_{p \in P_D} \eta_p . \delta_{pD} + b_{snd} + L_1(1 - b_{snd})}{2}, \forall sn \in SN, D \in \alpha \text{ ---------(8)}$$

Link Contraints

All data is directed toward the sink node, and each node collects information regarding the connection restrictions via the appropriate channel. A sensor node is selected to collect data from a specific source DS, ensuring that it has an out-degree value of at least 1, indicating that the connection originates from the source node.

$$\sum_{sn \in SN} J_{(ds,sn)} \geq 1, \forall\ sn \in SN \text{---------(9)}$$

Each node is limited to a maximum out-degree of 1, meaning that each node can have at most one outgoing connection. Within the inter-routing region, only one connection from sensor node x to y is permitted. These constraints characterize this scenario.

$$\sum_{sn \in SN} J_{(x,y)} \leq 1, \forall x \in SN \text{---------(10)}$$

The intermediate nodes pass data to the cluster head, which then establishes a connection to the sink for data transmission. One relay node, denoted as x, provides coverage to a cluster whose members satisfy the requirement. $b_{dsd} \geq 1$, $ds \in DS$. The total of in-degree links to relay node x is set to 1.

$$\sum_{y \in SN} J_{(x,D)} \geq b_{dsd}\,, \forall ds \in DS, \forall d \in D \text{ ---------(11)}$$

After the data forwarding procedure is finalized, the data is transmitted to the sink node, ensuring the availability of at least one node to maintain coverage to the sink node. The in-degree link summation is set to 1, indicating that each node has at least one incoming connection.

$$\sum_{x\in SN} J_{(x,S)} \geq 1 \text{---------(12)}$$

Constraints in node-to-node data transmission

The data flow between nodes x and y is represented as d (x, y) in the following equation.

$$d_{(x,y)} = \frac{(g^{-\lambda} \sum_{n\in SN} a(n.x)^{(DIFS)}(RTS+SIFS+CTS+\bar{B})+DIFS+\bar{N})}{g^{-\lambda} \sum_{n\in SN} a(n.x)^{(DIFS)-(RTS+SIFS+2\theta)} g^{-\lambda} \sum_{n\in SN} a(n,y)-\bar{N},\forall(x,y)\in M} \text{---------(13)}$$

In wireless communication protocols, various terms play crucial roles in ensuring efficient data transmission. The random backoff time, represented as $\overline{B}$, determines the average duration a device waits before initiating data transmission to mitigate potential collisions. Similarly, the average time spent in the network allocation vector (NAV), denoted by $\overline{N}$, reflects the period during which the channel remains occupied due to ongoing transmissions. The decision link symbolized as j, signifies a connection established based on predefined criteria or decisions within the network, influencing routing or data forwarding. Data transmission involves the exchange of control to send (CTS), acknowledgment (ACK), and request to send (RTS) frames, essential components facilitating communication between devices. Additionally, short inter-frame space (SIFS) and dispersed inter-frame space (DIFS) represent specific time intervals between frames, regulating access to the wireless medium and ensuring orderly transmission in wireless networks. These terms collectively contribute to the efficient operation and management of wireless communication systems, optimizing data throughput and minimizing interference and collisions. Data transmission between nodes is represented as Ø(x, y), with a range of 1 for active connections. The minimal energy usage TU is acquired by the function l(x,y) for transmission.

$$l_{(x,y)} - K_3\left(l - V_{(x,y)}\right) \leq \emptyset_{(x,y)}, \forall(x, y)\epsilon M \text{---------(14)}$$

An energy-efficient routing strategy, including trust values and one of three ECC-based encryption algorithms, has been proposed to improve security and efficiency in data transmission inside WSN. The proposed routing approach has three phases: trust score evaluation, clustering, and the implementation of the energy efficient trust-based secure routing technique. Trust scores for each sensor node are determined during the trust score evaluation phase by calculating the first trust score followed by subsequent trust ratings. Following two methods. Authentic nodes are those that really transmit received packets to their neighboring nodes.

Initially, the score for trust can be calculated with the weighted sum of diverse node behavior, namely responsiveness (P_i) and reliability in packet forwarding (R_i). The score computation is given in Equation 15.

$$T_i = \alpha R_i + (1-\alpha)P_i \text{---------(15)}$$

where the balance between responsiveness and reliability is accomplished by the weight factor α, which ranges between 0 and 1.

Subsequently, scores of the trust are updated based on the factors namely mobility of node, patterns of communication, and packet integrity. Mi indicates the mobility of node, patterns of communication is indicated by C_i, and packet integrity is indicated by I_i.

$$T_i' = \beta T_i + \gamma I_i + \delta M_i + \epsilon C_i \text{---------(16)}$$

where the significance of each weight factor is given by β, γ, δ, and ϵ.

The trust scores are evaluated and assigned for clusters. The C_i indicates the cluster j^th trust score. The cluster trust score is estimated by equation 17.

$$C_j = \frac{\sum_{i\in N_j} T_i}{|N_j|} \text{---------(17)}$$

where Nj indicates the set of nodes in the jth cluster.

The routing technique, based on the clustered network structure and trust values, should select routes considering both energy efficiency and node trust scores. The average trust value of the member node is estimated using equation 18.

$$Cost_{ij} = \alpha E_i + (1-\alpha)D_{ij}$$

where the energy and trust can be balanced with the weight factor α.

Incorporate ECC-based encryption algorithms into the routing protocol for secure data transmission. Utilize mathematical expressions for encryption and decryption processes based on ECC algorithms, such as elliptic curve Diffie-Hellman (ECDH) for key exchange and elliptic curve digital signature algorithm (ECDSA) for authentication and integrity. The procedure is given in Algorithm 1.

To initiate routing in the proposed model with ECC-based encryption, secure communication channels between nodes must be established, and optimal routes for data transmission determined. This begins with the key exchange initialization phase, where each node generates its public-private key pair using ECC and broadcasts its public key to neighboring nodes. Nodes then utilize elliptic curve Diffie-Hellman (ECDH) to establish shared secret keys for secure communication. Once established, nodes can encrypt and decrypt messages using symmetric encryption algorithms. Following this, the route discovery phase commences when a node requires sending data to a destination. The source node broadcasts a route request message, and intermediate nodes forward it while appending

Algorithm 1: ECC-based routing protocol

```
1. Key exchange initialization:
  - Each node generates its public-private key pair using ECC.
  - BroadcastPublicKey() // Nodes broadcast their public keys to neighboring nodes
2. Route discovery:
  - InitiateRouteDiscovery(destinationNode):
    nonce = GenerateRandomNonce()
    routeRequestMessage = {
      sourceNodeId,
      destinationNodeId,
      nonce,
      publicKeyOfSender
    }
    Broadcast(routeRequestMessage) // Source node broadcasts route request message
  - HandleRouteRequest(routeRequestMessage):
    VerifyMessageIntegrity(routeRequestMessage) // Verify message integrity using ECDSA
    VerifyMessageAuthenticity(routeRequestMessage) // Verify message authenticity using public key
    ForwardRouteRequest(routeRequestMessage) // Forward route request to neighbors
  - HandleRouteReply(routeReplyMessage):
    VerifyMessageIntegrity(routeReplyMessage) // Verify message integrity using ECDSA
    VerifyMessageAuthenticity(routeReplyMessage) // Verify message authenticity using public key
    EstablishRoute(routeReplyMessage) // Establish route to destination node
3. Route Establishment:
  - EstablishRoute(routeReplyMessage):
    // Once route reply is received, establish route to destination node
    // Encrypt data using shared secret key established with next hop node
    // Decrypt data at each intermediate node and forward to next hop node
4. Data Transmission:
  - TransmitData(destinationNode, data):
    sharedSecretKey = GenerateSharedSecretKey(destinationNode)
    encryptedData = EncryptData(data, sharedSecretKey)
    TransmitEncryptedData(encryptedData) // Transmit encrypted data along the established route
  - ReceiveAndDecryptData(encryptedData):
    sharedSecretKey = RetrieveSharedSecretKey()
    decryptedData = DecryptData(encryptedData, sharedSecretKey)
    VerifyDataIntegrity(decryptedData) // Verify data integrity using ECDSA signatures
5. Route Maintenance:
  - MaintainRoute():
    PeriodicallyUpdateSharedSecretKeysWithNeighbors() // Update shared secret keys using ECDH
    HandleRouteFailures() // Handle route failures and initiate route repair process if necessary
```

cryptographic information. Nodes verify message integrity using ECDSA and the authenticity of the sender. Upon receiving the request, the destination node replies with a route reply message signed using ECDSA. Subsequently, the route establishment phase begins, where the source node establishes the route to the destination using received route information. Data encryption occurs using shared secret keys, and intermediate nodes decrypt and forward packets. Finally, the route maintenance phase ensures ongoing secure communication, with nodes updating shared secret keys and initiating route repair if necessary due to node failure or network changes. By integrating ECC-based encryption, this model ensures both secure data transmission and efficient routing within WSNs.

Result and Discussion

This research work on ECC-based encryption and decryption encompasses key generation, encryption, and decryption processes implemented using Python programming. The key sizes for RSA and ECC are carefully chosen to ensure better security, utilizing different algorithms within a subgroup. Simulations are conducted in a node deployment area of m, accommodating up to 500 nodes, with LEACH serving as the basic routing protocol. Each sensor node is initialized with an energy level of 2 Joules. The simulation compares the performance of routing protocols LEACH, HEED, and the proposed CORAC. Notably, security comparisons are made against existing RSA, DynCH, TEA, SOPR, and AES algorithms. The transport layer protocol employed in simulations is the transmission control protocol (TCP). During simulations, data are collected, encrypted by the nodes themselves, and then routed to the base station through their respective clusters, as well as the cluster heads of other clusters. This comprehensive evaluation aims to provide insights into the performance and security implications of the proposed GBECC algorithm compared to established encryption standards and routing protocols in wireless sensor networks.

Key Computation Time Analysis

Key computation time analysis involves assessing the time it takes to generate cryptographic keys, which is crucial for encryption and decryption processes. This analysis typically considers the complexity of key generation algorithms and their efficiency in computing keys of sufficient strength. The time complexity of key generation algorithms is often expressed using Big O notation, indicating the worst-case time complexity in relation to the input size. For example, if a key generation algorithm has a time complexity of O(n^2), it means that the time required to generate keys increases quadratically with the size of the input. This analysis helps determine the computational overhead associated with key generation and its impact on overall system performance.

The simulation performance key computation time analysis of the proposed approach and existing technique is compared in Table 1 and illustrated in Figure 2.

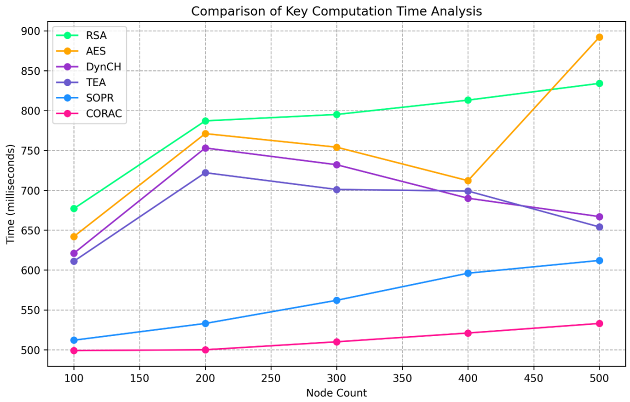

Figure 2: Key computation time analysis

Encryption Time Analysis

Encryption time analysis involves evaluating the time required to encrypt plaintext data using cryptographic algorithms and keys. The efficiency of encryption algorithms plays a significant role in determining encryption time, with faster algorithms reducing the computational burden on the system. The encryption time is influenced by factors such as the complexity of the encryption algorithm, the size of the plaintext data, and the strength of the cryptographic keys. By assessing encryption time, researchers can identify bottlenecks in the encryption process and optimize algorithms or hardware implementations to improve performance. The encryption time is estimated using below equation.

$$T_{enc} = f(P, K, A)$$

where the size of the plain text is P, the strength or size of the key is given as K, and A indicates the additional factors influencing encryption time. The simulation performance encryption time analysis of the proposed approach and existing technique is compared in Table 2 and illustrated in Figure 3.

Decryption Time Analysis

Decryption time analysis focuses on assessing the time it takes to decrypt ciphertext data using cryptographic keys. Similar to encryption time analysis, decryption time is influenced by factors such as the complexity of the decryption algorithm, the size of the ciphertext data, and the strength of the cryptographic keys. Analyzing decryption time helps evaluate the efficiency of decryption algorithms and their impact on overall system performance. By optimizing decryption algorithms or hardware implementations, researchers can reduce decryption time and enhance system responsiveness. The decryption time is estimated using the below equation.

$$T_{dec} = g(C, K, B)$$

Where C is the ciphertext data size, the size or strength of the decryption key is K, and B indicates the additional factors influencing decryption time. The simulation performance

Table 1: Key computation time analysis

Node count	*RSA*	*AES*	*DynCH*	*TEA*	*SOPR*	*CORAC*
100	677	642	621	611	512	499
200	787	771	753	722	533	500
300	795	754	732	701	562	510
400	813	712	690	699	596	521
500	834	892	667	654	612	533

decryption time analysis of the proposed approach and existing technique is compared in Table 3 and illustrated in Figure 4.

PDR Analysis

Packet delivery ratio (PDR) analysis involves evaluating the ratio of successfully delivered packets to the total number of packets sent in a wireless network. A high PDR indicates reliable communication and efficient packet delivery, while a low PDR may indicate network congestion, packet loss, or other communication issues. PDR analysis helps assess the effectiveness of routing protocols, congestion control mechanisms, and error recovery techniques in ensuring reliable data transmission in wireless networks.

$$PDR = \left(\frac{D}{S}\right) \times 100\%$$

where the count of the successfully delivered packets is D and the sent data packet count is S. The simulation performance packet delivery ratio (PDR) analysis of the proposed approach and existing technique is compared in Table 4 and illustrated in Figure 5.

Energy Consumption Analysis

Energy consumption analysis involves evaluating the amount of energy consumed by network devices during data transmission, processing, and other operations. In wireless sensor networks and other resource-constrained environments, energy efficiency is critical for prolonging network lifetime and ensuring reliable operation. By analyzing energy consumption patterns, researchers can identify energy-intensive tasks, optimize algorithms and protocols to reduce energy consumption, and design energy-efficient hardware solutions. Energy consumption (E) can be calculated based on factors such as the energy consumed per unit of data transmission (E_{tx}), the energy consumed per unit of data processing (E_{proc}), and the total amount of data transmitted or processed (D). The formula for energy consumption is

$$E = E_{tx} \times D_{tx} + E_{pro} \times D_{pro}$$

where: E_{tx} is the energy consumed per unit of data transmission, D_{tx} is the total amount of data transmitted,

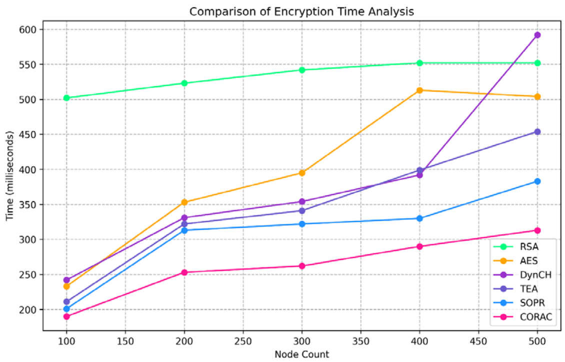

Figure 3: Encryption time analysis

E_{proc} is the energy consumed per unit of data processing, and D_{proc} is the total amount of data processed. The simulation performance energy consumption analysis of the proposed approach and existing technique is compared in Table 5 and illustrated in Figure 6.

Packet Loss Analysis

Packet loss analysis focuses on assessing the rate at which packets are lost or dropped during transmission in a network. Packet loss can occur due to various reasons, including network congestion, errors, collisions, and link failures. Analyzing packet loss helps evaluate the reliability of communication channels, identify potential causes of packet loss, and implement mechanisms to mitigate its impact on data transmission quality. Packet loss rate (PLR) is calculated as the ratio of lost packets (L) to the total number of packets sent (S). The formula for packet loss rate is.

$$PLR = \left(\frac{L}{S}\right) \times 100$$

where L is the number of lost packets, and S is the total number of packets sent. The simulation performance PLR analysis of the proposed approach and existing technique is compared in Table 6 and illustrated in Figure 7.

The comparative analysis of key computation time reveals notable differences among the cryptographic algorithms evaluated. At a node count of 100, the proposed CORAC algorithm exhibits the lowest key computation time of 499 ms, followed closely by SOPR with 512 ms. RSA, AES, DynCH, and TEA algorithms demonstrate higher key computation times, with RSA being the slowest at 677 ms. This trend

Table 2: Encryption time analysis

Node count	*RSA*	*AES*	*DynCH*	*TEA*	*SOPR*	*CORAC*
100	502	233	242	211	201	190
200	523	353	331	322	313	253
300	542	395	354	341	322	262
400	552	513	392	399	330	290
500	552	504	592	454	383	313

Table 3: Decryption time analysis

Node count	*RSA*	*AES*	*DynCH*	*TEA*	*SOPR*	*CORAC*
100	333	333	337	353	483	544
200	357	333	358	333	517	578
300	333	356	395	355	553	669
400	490	399	538	593	603	689
500	433	555	505	593	613	712

Table 4: Packet delivery ratio

Node count	*RSA*	*AES*	*DynCH*	*TEA*	*SOPR*	*CORAC*
100	76	78	80	81	86	92
200	78	79	83	84	87	95
300	79	81	85	86	89	96
400	80	82	86	87	91	97
500	81	84	88	89	92	98

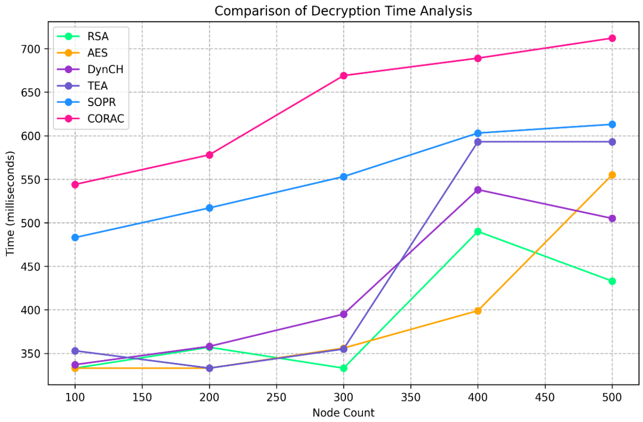

Figure 4: Decryption time analysis

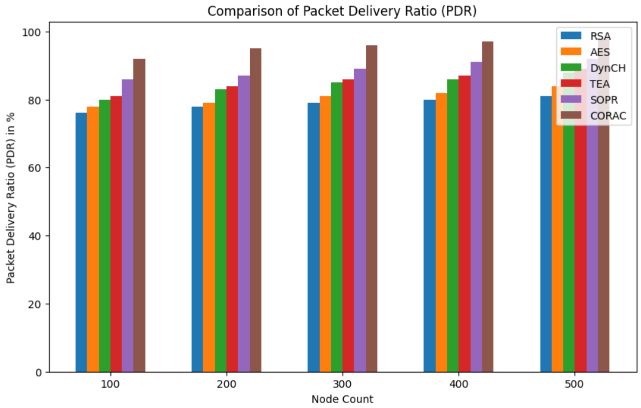

Figure 5: Packet delivery ratio

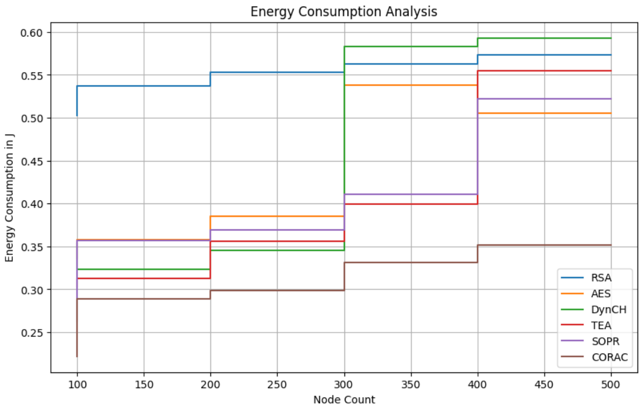

Figure 6: Energy consumption analysis

persists across increasing node counts, with CORAC consistently outperforming other algorithms in terms of key computation time. Similarly, encryption time analysis shows significant variations in the performance of cryptographic algorithms. At node count 100, CORAC demonstrates the fastest encryption time of 190 ms, followed by SOPR with 201 ms. In contrast, RSA and AES algorithms exhibit longer encryption times, with RSA taking 502 ms and AES 233 ms. The decryption time analysis further corroborates these findings, with CORAC consistently outperforming other algorithms across different node counts. PDR analysis indicates that the proposed CORAC algorithm achieves higher PDR values compared to other algorithms, reflecting its effectiveness in ensuring reliable communication and efficient packet delivery. Additionally, energy consumption analysis reveals that CORAC consumes less energy per unit of data transmission compared to other algorithms, highlighting its energy efficiency. Lastly, packet loss analysis demonstrates that CORAC consistently exhibits lower packet loss ratios, indicating its reliability in data transmission. Overall, the comprehensive evaluation suggests that the CORAC algorithm offers superior performance in terms of key computation time, encryption time, decryption time, packet delivery ratio, energy consumption, and packet loss ratio compared to existing cryptographic algorithms.

Table 5: Energy consumption analysis

Node count	*RSA*	*AES*	*DynCH*	*TEA*	*SOPR*	*CORAC*
100	0.503	0.337	0.353	0.303	0.263	0.222
200	0.537	0.358	0.323	0.313	0.357	0.289
300	0.553	0.385	0.345	0.356	0.369	0.299
400	0.563	0.538	0.583	0.399	0.411	0.331
500	0.573	0.505	0.593	0.555	0.522	0.352

Table 6: Packet loss ratio

Node count	*RSA*	*AES*	*DynCH*	*TEA*	*SOPR*	*CORAC*
100	24	22	20	19	14	8
200	22	21	17	16	13	5
300	21	19	15	14	11	4
400	20	18	14	13	9	3
500	19	16	12	11	8	2

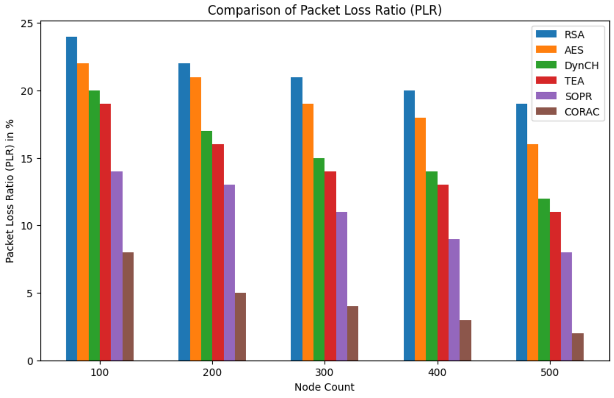

Figure 7: Packet loss ratio

Conclusion

The proposed methodology introduces a secured routing algorithm tailored for cluster-based networks, integrating trust-aware authentication mechanisms to enhance both energy efficiency and data delivery security in WSNs. The approach begins with clustering nodes based on proximity or network topology and then implements a trust-based authentication mechanism to evaluate the reliability and integrity of nodes and links within the network. Building upon these foundations, a secured routing protocol is devised to capitalize on cluster-based organization and trust-based authentication, thereby facilitating energy-efficient and secure data transmission. By integrating ECC-based encryption algorithms into the routing protocol, the proposed system ensures secure communication channels between nodes and optimizes routes for data transmission. Performance evaluation through simulation studies demonstrates the efficacy of the proposed approach in terms of packet delivery ratio, energy consumption, and security effectiveness. The comprehensive analysis highlights the superiority of the proposed CORAC algorithm over existing techniques, showcasing its potential for improving security and efficiency in wireless sensor networks.

References

Abu-Ain, T., Ahmad, R., & Sundararajan, E. A. (2021). Analysis of the effect of dynamic clustering and lightweight symmetric encryption approaches on network lifetime in WSNs.

Aruchamy, P., Gnanaselvi, S., Sowndarya, D., & Naveenkumar, P. (2023). An artificial intelligence approach for energy-aware intrusion detection and secure routing in internet of things-enabled wireless sensor networks. Concurrency and Computation: Practice and Experience, 35(23), e7818.

Cheng, Y., Liu, Y., Zhang, Z., & Li, Y. (2023). An asymmetric encryption-based key distribution method for wireless sensor networks.

Dass, R., Narayanan, M., Ananthakrishnan, G., Kathirvel Murugan, T., Nallakaruppan, M. K., Somayaji, S. R. K., ... & Almusharraf, A. (2023). A cluster-based energy-efficient secure optimal path-routing protocol for wireless body-area sensor networks. Sensors, 23(14), 6274.

Elamurugu, V., & Evanjaline, D. J. (2021). An efficient and secure text encryption scheme for wireless sensor network (WSN) using dynamic key approach. International Journal of Computer Networks and Applications, 8(6), 788-794.

Ganesan Sangeetha, M., Vijayalakshmi, S., Ganapathy, S., & Kannan, A. (2020). An improved congestion-aware routing mechanism in sensor networks using fuzzy rule sets. Peer-to-Peer Networking and Applications, 13(3), 890-904.

Gulen, U., & Baktir, S. (2020). Elliptic curve cryptography for wireless sensor networks using the number theoretic transform. Sensors, 20(5), 1507.

Gupta, S. C., Singh, B., Amjad, M., Gopianand, M., & Bhuvaneswari, E. (2021, March). Security enhancement using quantum cryptography in WSN. In 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS) (Vol. 1). IEEE.

Hamza, A. H., & Al-Alak, S. M. K. (2021, February). Evaluation of key generator of multiple asymmetric methods in wireless sensor networks (WSNs). In Journal of Physics: Conference Series (Vol. 1804, No. 1, p. 012096). IOP Publishing.

Hussain, M. Z., & Hanapi, Z. M. (2023). Efficient secure routing mechanisms for the low-powered IoT network: A literature review. Electronics, 12(3), 482.

Mallick, B. B., & Bhatia, A. (2021). Comparative analysis of the impact of cryptography algorithms on wireless sensor networks. arXiv preprint arXiv:2107.01810.

Mirvaziri, H., & Hosseini, R. (2020). A novel method for key establishment based on symmetric cryptography in hierarchical wireless sensor networks. Wireless Personal Communications, 112, 2373-2391.

Mohindru, V., Singh, Y., & Bhatt, R. (2020). Hybrid cryptography algorithm for securing wireless sensor networks from node clone attack. Recent Advances in Electrical & Electronic Engineering (Formerly Recent Patents on Electrical & Electronic Engineering), 13(2), 251-259.

Morales-Sandoval, M., Flores, L. A. R., Cumplido, R., Garcia-Hernandez, J. J., Feregrino, C., & Algredo, I. (2021). A compact FPGA-based accelerator for curve-based cryptography in wireless sensor networks. Journal of Sensors, 2021, 1-13.

Pooja, & Chauhan, R. K. (2022). Triple phase hybrid cryptography technique in a wireless sensor network. International Journal of Computers and Applications, 44(2), 148-153.

Qazi, R., Qureshi, K. N., Bashir, F., Islam, N. U., Iqbal, S., & Arshad, A. (2021). Security protocol using elliptic curve cryptography algorithm for wireless sensor networks. Journal of Ambient Intelligence and Humanized Computing, 12, 547-566.

Ravi, K., Khanai, R., & Praveen, K. (2020). A secure key and data exchange mechanism using elliptic curve cryptography on WSN. In Emerging Trends in Electrical, Communications, and Information Technologies: Proceedings of ICECIT-2018 (pp. 529-541). Springer Singapore.

Shah, P., Arora, M., & Adhvaryu, K. (2020, October). Lightweight cryptography algorithms in IoT: A study. In 2020 Fourth International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud)(I-SMAC) (pp. 332-336). IEEE.

Tropea, M., Spina, M. G., De Rango, F., & Gentile, A. F. (2022). Security in wireless sensor networks: A cryptography performance analysis at the MAC layer. Future Internet, 14(5), 145.

Vivek, K., Kale, M. R., Thotakura, V. S. K., & Sushma, K. (2021, November). An efficient triple-layered and double secured cryptography technique in wireless sensor networks. In 2021 IEEE International Conference on Distributed Computing, VLSI, Electrical Circuits and Robotics (DISCOVER) (pp. 117-122). IEEE.

Xiao, Y., Rayi, V. K., Sun, B., Du, X., Hu, F., & Galloway, M. (2007). A survey of key management schemes in wireless sensor networks. Computer Communications, 30, 2314-2341.