IIETA International Information and Engineering Technology Association
Advancing the World of Information and Engineering

Mathematical Modelling of Engineering Problems
Vol. 11, No. 4, April, 2024, pp. 1060-1070
Journal homepage: http://iieta.org/journals/mmep

Key Agreement Scheme for Authorization and Authentication of WSN in IoT-5G Using Elliptic Curve Cryptography

Hemantaraj M. Kelagadi[1], M. R. Prasad[2], B. T. Ramesh[3*], Arun Kumar Bongale[3], Satish Kumar[4]

[1] School of Electronics & Communication Engineering, KLE Technological University, Hubballi 580031, India
[2] Department of Computer Science & Engineering, Vidyavardhaka College of Engineering, Gokulam 3 Stage, Karnataka 570002, India
[3] Symbiosis Institute of Technology (SIT), Symbiosis International (Deemed University) (SIU), Lavale, Pune 412115, Maharashtra, India
[4] Symbiosis Centre for Applied Artificial Intelligence, Symbiosis International (Deemed University) (SIU), Pune 412115, Maharashtra, India

Corresponding Author Email: rameshbt049@gmail.com

Special issue: Emerging Trends in Computational Intelligence, Networks Technologies, and Wireless Communication Systems

https://doi.org/10.18280/mmep.110424

Received: 14 November 2023
Revised: 16 January 2024
Accepted: 30 January 2024
Available online: 26 April 2024

Keywords:
Geography and Energy Aware Routing (*GEAR*), *Internet of Things* (*IoT*), *One Sample Median Vigenere Cipher based Diffie Hellman* (*OSMVC-DH*), *Public Private and Session-based Elliptic Curve cryptography* (*PPSECC*), *Separate Chaining based Secure Hash Algorithm 512* (*SC-SHA-512*), *Triangle walk strategy based coati optimization algorithm* (*TW-COA*), *Wireless Sensor Networks* (*WSN*)

ABSTRACT

The successful deployment of the Internet of Things (IoT) heavily relies on the integration of Wireless Sensor Networks (WSN) with 5th Generation (5G). However, this integration presents data security challenges during continuous data transactions in WSN. Thus, to provide secured data transfer from any location in WSN, a secured data transmission framework using Public Private and Session-based Elliptic Curve Cryptography (PPSECC) and One Sample Median Vigenere Cipher-based Diffie-Hellman (OSMVC-DH) is proposed. First, the node is registered and then authenticated regarding the node's checksum. Subsequently, Geography and Energy Aware Routing (GEAR) is employed for routing, and the optimal routes are selected using the Triangle Walk strategy-based Coati Optimization Algorithm (TW-COA). The data from sensed nodes are encrypted using PPSECC, based on a Session Key (SK) generated using the OSMVC-DH technique. The encrypted data that transmits through the selected paths is changed into a hashcode using Separate Chaining-based Secure Hash Algorithm 512 (SC-SHA-512). At the receiver end, the hashcode-matched data is decrypted in the server. Hence, the proposed model authorized the user by generating the hashcode in 313ms and secured the data with 98% Security Level and 1137ms Encryption Time, thus showing better performance than existing models.

1. INTRODUCTION

One of the latest technologies, which permits remote control of heterogeneous networks, is IoT. IoT makes remote control possible for certain applications through the use of technologies like WSNs across heterogeneous networks [1]. Networks of sensor nodes with constrained energy as well as memory resources, which communicate with one another wirelessly, are termed WSNs; also, they support to collect and store data regarding the monitored objects' state that enables further evaluation, which benefits stakeholders [2]. In several applications like (A) smart home, (B) smart city, (C) smart health, and (D) smart grid, 5G is becoming an active candidate. Since 5G is becoming commercially viable, IoT has become more ingrained in daily life [3]. The potential of incorporating WSNs as well as 5G in the IoT is predictable to permit IoT to offer a variety of handy services, but simultaneously, it would also lead to novel security risks [4].

Typical WSNs are subjected to various problems associated with maintenance, scalability, and vulnerability to security threats as they depend on cloud infrastructures for tackling huge gathered data. Thus, for IoT security, systems for authorization, access control, and privacy protection are important [5]. The security solutions might be traditional cyber security approaches, namely encryption, antimalware, password protection, firewall, network segmentation, or else advanced techniques [6]. Device authentication is one of the elementary security services for securing the IoT in smart cities. For offering suitable access control for those IoT devices, it verifies the IoT devices' identity encompassed in the communication [7].

Hash function-based authentication schemes for WSNs were introduced recently. However, they were prone to impersonation attacks, smart card loss attacks, along with SK disclosure attacks [8]. Since anyone can use the public key to confirm the authenticity and integrity of a message, the existing models resolved the issue by generating a public key. However, the authentication token, which is publicly verified by deploying the sender's public key, public-key operations, is particularly computationally costly. So, some of the other

existing models used the Diffie–Hellman key agreement algorithm, Chebychev chaotic map, together with ECC for building a secured user authentication as well as key agreement protocol [9, 10]. Even though the computational cost during authentication is reduced by the prevailing models, the security of the key is not concentrated, which could lead to the hacking of information. Also, the existing models could not retrieve the original data after encryption. Thus, to address these mitigations and to improve the secured data transfer, a novel framework for data security, authentication and authorization along with key agreement using PPSECC and OSMVC-DH is proposed.

The main disadvantages of the prevailing research methodologies are:

- Most of the existing research methodologies did not concentrate the data security for continuous transactions in a network at any location. The data authorization process is affected once the generated conventional keys are hacked.
- Owing to inequality in data sensing along with distance as of Base Station, each sensor node has a diverse energy consumption rate in WSNs, which causes energy disparity amongst sensor nodes in the network which in turn shortens the network lifetime.
- Existing methodologies concentrated on secure data retrieval but they failed to focus on the integrity of data and their loss.
- Existing hashing-based node authentication techniques couldn't retrieve the original message and produced the same hash values in some cases.
- Existing Works securely transmitted data by using crypto analysis, but they failed to protect the information sent by the node to remain unreadable for unauthorized users.

The objective of this research is detailed as follows.

- To provide data security during continuous transmission in WSN, a Session Key is generated using OSMVC-DH for each transmission.
- The lifetime of the model is improved by generating an energy-efficient routing using the GEAR method and then selecting the optimal path using the TW-COA technique.
- To prevent the data integrity and loss of information, parity bits of the data are added, and then, on the receiver side, the data is reconstructed.
- To retrieve the original data and to produce different hash values, the node authentication is done using Left Shift 2's (LS2) complement checksum method.
- The information is protected by encrypting the data using the PPSECC model.

The rest of the article is organized as follows: Section 2 provides an overview of related research pertinent to the proposed model. Section 3 elaborates on the details of our proposed technique. Section 4 presents the results of our proposed system along with a comprehensive discussion of performance metrics. Finally, Section 5 concludes the work

2. RELATED WORK

An ECC-centric Privacy-Preserving Authentication, Authorization, along with a Key Agreement System by taking the incorporation of WSNs and 5G for IoT is recommended by the authors [11]. To exhibit the system's capability for withstanding several security attacks, formal and informal security analysis was done. In the end, the performance analysis with related schemes indicated it is efficient and secure. But they still faced security problems and lacked node authentication.

An ECC-centric key management together with flexible authentication for augmenting WSN's system robustness in 5G integrated IoT is developed [12]. For user authentication along with authorization, ECC was employed. The system's efficiency was revealed by comparing it with the existing methods on the basis of various performance metrics. However, the model did not concentrate on energy constraints.

A Lightweight Anonymous Privacy-Preserving Three-factor Authentication Scheme for WSN-centric IoT (LAPTAS) is explored by Abdi Nasib Far et al. [13]. Sensor node dynamic registration, password change, and revocation phase were supported by the LAPTAS. The scheme was simulated by the Protocol Verifier tool as well as analogized with other similar systems, which ensured the superiority of LAPTAS. Yet, the scheme was threatened by user impersonation along with tracking attacks.

An improved privacy-preserving remote user authentication scheme for WSN is developed by Rangwani et al. [6]. By employing the probabilistic Random-Oracle Model (ROR), it was evaluated to prove its robustness. As per evaluation, the scheme was more efficient when weighed against the available techniques. Nevertheless, the system lacked password verification, an improper SK agreement, and improper authentication.

A lightweight, secure Identity-Based Online/Offline Signature Technique (IBOOST) in 5G-WSNs is explored by employing fractional chaotic maps [14]. Multi-time usage of offline storage was attained at a less processing time. Moreover, the pre-registration process was enabled with a secret key; also, no secret key was essential in the offline stage. This system offered superior security together with improved computational overhead when weighed against the primary methodologies. Yet, the system aimed at the usual public key-centric situation, causing several data attacks.

A privacy-preserving implicit authentication system for IoT is presented [15]. Primarily, the security together with privacy requirements for mobile intelligent terminal's security authentication was summarized. Next, a privacy-preserving implicit authentication and a partial homomorphic public key encryption scheme were presented. When weighed against other associated protocols, this technique's communication and computation were more effectual. However, the accuracy rate was not satisfying.

A Continuous Hybrid and Energy-efficient Secure Data Aggregation (CHESDA) for WSN is flaunted [16]. By the slice-mixing technique, privacy-preserving was maintained. For choosing optimal slicing in every sub-tree, fuzzy logic was implemented. For verifying the key authentication scheme, the logic of Gong, Needham, and Yahalom (GNY) was wielded. As per the evaluation, the CHESDA was more energy-efficient as well as highly secure with low communication overhead. Yet, a high end-to-end delay was caused.

A lightweight authentication for the Narrowband Internet of Things (NB-IoT) is presented [17]. A dynamic key generation, which regarded entropy and performed several operations for constantly engendering huge unique keys, was presented. For diverse categories of IoT applications in 5G networks, the dynamic key scheme ensured an enhanced security level and reduced communication overhead. But, while handling huge data, accuracy problems arose.

A cryptographic-centric clustering structure to preserve data

privacy deploying Optimal Privacy-Multihop Dynamic Clustering Routing Protocol (OP-MDCRP) is explored [18]. High data privacy was offered by employing the Elliptic Curve Integrated Encryption-Key Provisioning Method (ECIES-KPM) and a small key size. The ECIES-KPM verification process restricted data security-centric attacks. As per the experimental outcomes, when analogized to the prevailing techniques, this system offered more data security. Nevertheless, insufficient memory and low communication capability were faced by this system.

A Secure and Lightweight three-factor-centric User Authentication protocol for WSN (SLUA-WSN) is developed [19, 20]. Via informal and formal analysis like the ROR model, together with AVISPA simulation, the SLUA-WSN's security was analyzed. Security threats were prevented; also, anonymity and untraceability were ensured. However, the model was unable to produce an SK between each entity and enable strong mutual authentication.

The literature review reveals several critical issues in the domain of secure authentication and privacy preservation in the context of integrating WSN with 5G for IoT applications. These issues encompass security vulnerabilities, inadequate node authentication, neglect of energy constraints, incomplete features like missing password verification and improper SK agreement, susceptibility to data attacks, accuracy problems, resource limitations affecting memory and communication capabilities, and a lack of strong mutual authentication in certain models. Addressing these multifaceted challenges is paramount for the development of robust and comprehensive security solutions in WSNs integrated with 5G for ensuring the privacy and integrity of data transmission in IoT environments.

Thus, to alleviate the mentioned issues, a secure authentication, authorization, and key agreement scheme is proposed for WSN in 5G-integrated IoT using PPSECC and OSMVC-DH, and its main contributions are detailed as follows:

- An SK is generated for each transmission using OSMVC-DH to enhance the data security for continuous transmission.
- Energy-efficient routing is done and optimal routes are selected using TW-COA to increase the lifetime.
- Parity bits are added and data reconstruction is done to prevent data integrity and loss.
- Checksum-based connection establishment is proposed to perform efficient node authentication.
- A key agreement scheme-based PPSECC is introduced to protect the information.

3. PROPOSED SECURED DATA TRANSMISSION FRAMEWORK

In this paper, a secure authentication, authorization, along with a key agreement scheme for WSN in 5G-integrated IoT using PPSECC and OSMVC-DH are proposed. An SK is utilized along with the public and private keys in the PPSECC to enhance data security. In Figure 1, the structural design of the proposed system is modeled.

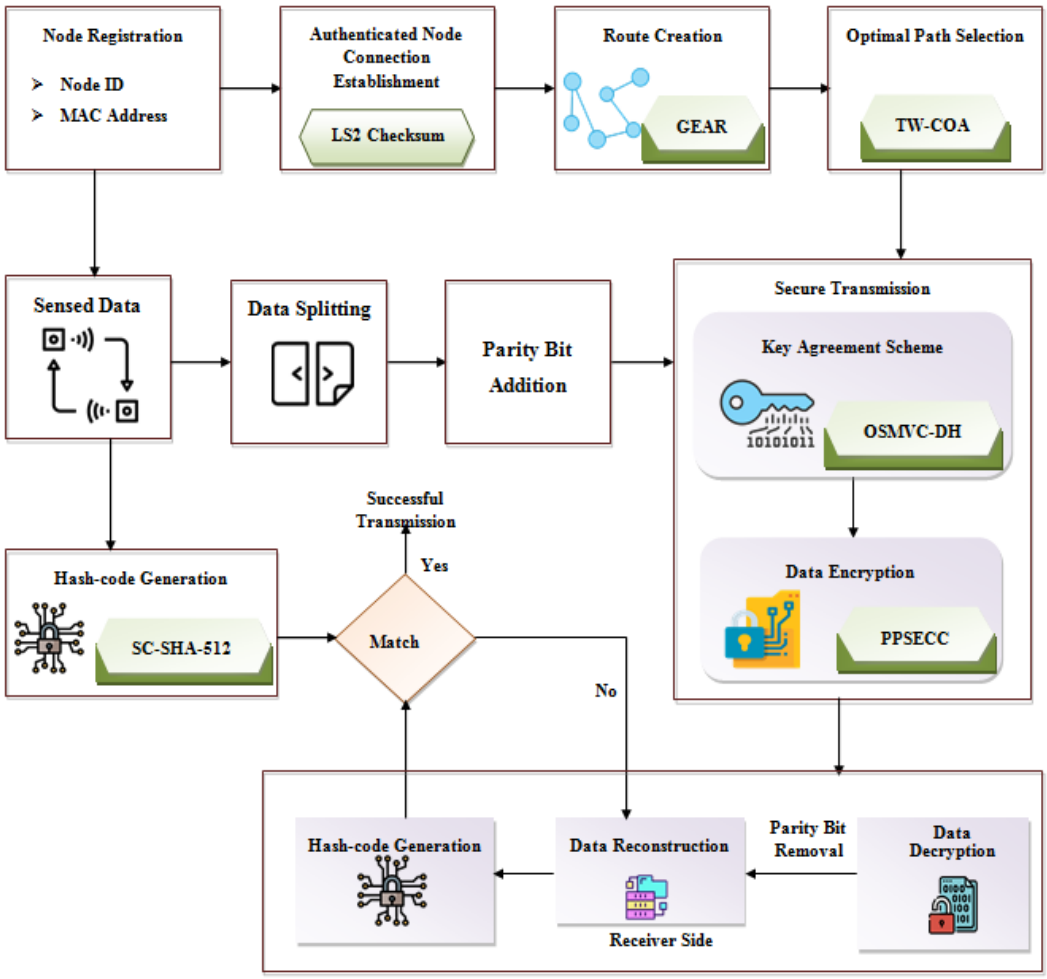

Figure 1. Structural design of the proposed system

Experimental Setup:

The performance of the proposed work depends on the sensor nodes of the WSN. The sensor nodes are chosen for efficient data transfer, and in this work, the sensor nodes in the range of (100 to 500) are taken for processing. For each node, the node Identification (ID) and Media Access Control (MAC) address are provided, and along with that, the power consumed by the node and the location of the node are also considered for this model. The dynamic nodes are chosen because the fluctuations present in the nodes of WSN can be identified by the dynamic consideration of nodes. Other parameter details included for the effective secured data transfer in WSN are as follows: the network size of 100*100 square meters, base station location of 50*50 meters, data packet size of 4000 bits, energy of 0.5 Joules and Data Aggregation Energy cost of 50 nJ/bit (nano Joules per bit) is taken for analysis. In the experiment, the input data are passed through the optimal path. Then, the data present in the path is encrypted and transferred to the receiver side.

Experiment Dataset:

The Intel Lab data dataset is taken for processing the proposed model. This dataset contains 14400 data collected from 54 sensors deployed in the Intel Berkeley Research Lab. A total of 52 attributes are available in the dataset and these attributes are used for processing. The proposed framework is explained in detail as follows.

3.1 Node registration

Primarily, the proposed methodology starts with the node registration phase. The nodes entering the network are registered using their Node ID and Media Access Control (MAC) address. The registered nodes ($\Re$) are expressed as,

$$\Re = \{\Re_1, \Re_2, \Re_3, ..., \Re_n\} \quad (1)$$

Here, $\Re_n$ implies the n^{th} number of nodes.

3.2 Authenticated node connection establishment

Here, for the registered nodes ($\Re$), authenticated node connection establishment is performed. The proposed works consider the checksum value from the MAC address of the corresponding nodes to verify the authentication of nodes. The checksum value represents the number of bits in a transmission message. The conventional checksum is prone to hacking. Thus, the LS2 complement checksum is utilized here to enhance security. The authentication process based on the LS2 checksum is explained as follows,

- Primarily, the MAC address of the node is divided into the q number of blocks with data bits in each block.
- Next, all the κ data blocks are added. The additional output is complemented using the 2's complement and finally left shift is performed.
- The computed output is the LS2 checksum value of the corresponding node. Likewise, the checksum values are computed for the nodes $\Re$.
- The LS2 checksum value is integrated and transmitted to the receiver node to perform authenticated node connection establishment.
- The received value is divided into the κ number of blocks on the receiver node.

The divided data blocks along with the LS2 checksum value of the receiver node are added and the output is complemented using 2's compliment and left shift.

If the obtained result is 0, then there is no error and the connection between the nodes is established.

If the obtained result is not 0, the node is declined.

In the end, the nodes are verified by establishing connections centered on the LS2 checksum. The authenticated nodes are denoted as $\Re_{auth}$.

3.3 Route creation

To perform data transmission, suitable routes between the source as well as destination node are obtained. For secure routing, the GEAR algorithm is utilized. For routing a packet to the target region, GEAR deploys energy-aware as well as geographically-informed node selection techniques. This algorithm comprises three assumptions:

i. Every packet is destined for a target region.
ii. Every node is aware of its location and remaining energy. level, along with its neighbors' location and remaining energy level.
iii. The links between the nodes are bidirectional.

For transferring the packets towards the target region, GEAR selects the closest node as the next-hop when a neighbor that is closer to the destination exists. When every neighbor is farther away, the next-hop node chosen by GEAR minimizes the cost value of this neighbor.

For propagating the packet within the region, the Recursive Geographic Forwarding algorithm is used. In some cases, restricted flooding is wielded by low-density networks to prevent routing loops.

The energy-efficient neighbor is selected based on the cost estimated the estimated cost is based on the distance to the target region from a node$dist(\Re_{auth}, \Im)$ and residual energy at the node. The estimated cost $est(\Re_{auth}, \Im)$ is described as,

$$est(\Re_{auth}, \Im) = \mu \cdot dist(\Re_{auth}, \Im) + (1-\mu)e(\Re_{auth}) \quad (2)$$

Here, $\Im$ implies the target region, μ signifies the tunable weight, and $e(\Re_{auth})$ denotes the consumed energy by the node. While receiving the packet, the node routes toward the destination, and in the meantime, it ensures that all of its neighbors consume the same amount of energy. This trade-off is accomplished by the nodes by reducing the learning cost. The learned cost $learn(\Re_{auth}, \Im)$ is defined as,

$$learn(\Re_{auth}, \Im) = learn(\Re_{auth}^{\min}, \Im) + est(\Re_{auth}, \Re_{auth}^{\min}) \quad (3)$$

Here, $\Re_{auth}^{min}$ implies the cost-minimized node. GEAR selects the neighbors, and the next hop node, and performs routing by minimizing the cost. Hence, the possible routes between the source and target nodes are created. The created routes are expressed as,

$$H = \{h_1, h_2, h_3, ..., h_m\} \quad (4)$$

Here, H implies the total number of routes created and h_m signifies the m^{th} number of the routes created.

3.4 Optimal path selection

After creating the routes H, by employing the TW-COA, the optimal path is selected. The Coati Optimization Algorithm (COA) is based on the hunting strategy of the coatis. The conventional COA is selected for its evident superiority in optimal path selection and it has high convergence speed. However, in the exploration phase, the coati attacked the iguana in a random manner, which led to a problem of falling into local optima. Thus, the Triangle Walk (TW) strategy that positions the iguana in search of food is introduced in the exploration phase of COA to enhance the prey search approach.

The TW-COA starts by randomly initializing the coati's position in the search space. Here, the routes H are considered as the coati population. The initialization is expressed as,

$$H_{i,j} = \ell_j + r*\left(\upsilon_j - \ell_j\right),\ i = 1,2,....,N,\ j = 1,2,..,K \tag{5}$$

Here, ℓ_j and υ_j signifies the lower and upper bounds, r implies the random number of range [0, 1], N denotes the total number of coatis, and K delineates the number of decision variables. Now, the initialized members are identified using a population matrix, which is expressed:

$$H = \begin{bmatrix} h_1 \\ \vdots \\ h_i \\ \vdots \\ h_N \end{bmatrix}_{N\times K} = \begin{bmatrix} h_{1,1} & \cdots & h_{1,j} & \cdots & h_{1,K} \\ \vdots & \ddots & \vdots & \iddots & \vdots \\ h_{i,1} & \cdots & h_{i,j} & \cdots & h_{i,K} \\ \vdots & \iddots & \vdots & \ddots & \vdots \\ h_{N,1} & \cdots & h_{N,j} & \cdots & h_{N,K} \end{bmatrix}_{N\times K} \tag{6}$$

Here, h_i implies the i^{th} candidate of the population. The fitness (F) is calculated for every candidate solution and is described as,

$$F(H) = \langle \min(R), \max(E) \rangle \tag{7}$$

where, fitness is based on the minimum response time (R) and maximum energy (E). The member with the best fitness value is regarded to be the best solution. The TW-COA works under two phases explained as follows:

Hunting strategy on iguana: The coatis reach trees to attack the iguana in the hunting phase. Half of the members climb the tree and their position is described by the expression,

$$H_{i,j}^{(1)} = H_{i,j} + r\left(I_j - \Lambda \cdot H_{i,j}\right),\ i = 1,2,....,\left\lceil \frac{N}{2} \right\rceil \tag{8}$$

$$\Lambda = \frac{\psi}{\wp} \tag{9}$$

Here, $H_{i,j}^{(1)}$ implies the position in the first phase, I_j signifies the position of the iguana, Λ delineates the triangle walk strategy, ψ elucidates the angular velocity, and $\wp$ implies the tangential velocity of the path. In search of the iguana, the remaining half of the population on the ground moves in a triangular trajectory. It is expressed as,

$$H_{i,j}^{(1)} = \begin{cases} H_{i,j} + r*\left(I_j - \Lambda \cdot H_{i,j}\right), & F_I < F_i; \\ H_{i,j} + r*\left(H_{i,j} - I_j\right), & else \end{cases} \tag{10}$$

Here, F_I implies the iguana's fitness function. Now, the position update based on fitness in phase is explained by the following expression,

$$H_i = \begin{cases} H_i^{(1)}, & F_i^{(1)} < F_i \\ H_i, & else \end{cases} \tag{11}$$

Escaping from Predators: To escape from the predators, the coatis move to a safer position. The position of the coatis based on the escaping behavior is expressed as,

$$H_{i,j}^{(2)} = H_{i,j} + (1 - 2\times r)*\left(\ell_j^{local} + r.\left(\upsilon_j^{local} - \ell_j^{local}\right)\right) \tag{12}$$

$$\ell_j^{local} = \frac{\ell_j}{t},\ \upsilon_j^{local} = \frac{\upsilon_j}{t},\quad t = 1,2,...,\tau \tag{13}$$

Here, $H_{i,j}^{(2)}$ implies the newer position of the coatis in phase (2), ℓ_j^{local} and υ_j^{local} signifies the local lower and upper bounds, t implies the iteration counter, and τ elucidates the total number of iterations. The position update in this phase (2) is defined by the following expression,

$$H_i = \begin{cases} H_i^{(2)}, & F_i^{(2)} < F_i; \\ H_i, & else \end{cases} \tag{14}$$

Here, $F_i^{(2)}$ implies the fitness value obtained in phase 2. Until the termination criterion is met, the steps are repeated and the position is updated. In the end, the best solution (H_{opt}) with the best fitness is obtained as output by the TW-COA. The procedure of TW-COA is explained in Algorithm 1.

Algorithm 1. TW-COA Technique

Input: Routes H
Output: Optimal Paths H_{opt}

Begin
Initialize the parameters of the problem and number of iterations
Generate the initial position using the equation,
$H_{ij} = \ell_j + r*\left(v_j - \ell_j\right), i = 1,2,\ldots,N,\ j = 1,2,\ldots,K$
Evaluate the Fitness(F)
For t=1 to τ **do**
Update I_j
# phase 1
For i=1 to N/2 **do**
Calculate the newpositions using equation
$$H_{i,j}^{(1)} = H_{i,j} + r\left(I_j - \Lambda \cdot H_{i,j}\right)$$
Update the position using the equation,
$$H_i = \begin{cases} H_i^{(1)}, & F_i^{(1)} < F_i \\ H_i, & \text{else} \end{cases}$$
End For
For i=1 + N/2: N **do**
Calculate the newpositions using the equation,
$$H_{i,j}^{(1)} = \begin{cases} H_{i,j} + r*\left(I_j - \Lambda \cdot H_{i,j}\right), & F_I < F_i; \\ H_{i,j} + r*\left(H_{i,j} - I_j\right), & \text{else} \end{cases}$$

Update the position

End For

phase 2

Calculate new status based on the local bounds using the equation,

$$H_{i,j}^{(2)} = H_{i,j} + (1-2\times r)*\left(\ell_j^{\text{local}} + r\cdot\left(v_j^{\text{local}} - \ell_j^{\text{local}}\right)\right)$$

Update the position using the expression,

$$H_i = \begin{cases} H_i^{(2)}, & F_i^{(2)} < F_i; \\ H_i, & \text{else} \end{cases}$$

Save the best candidate solution

End for

Return H_{opt}

End

Hence, by using TW-COA, the optimal paths (H_{opt}) with maximum energy and minimum response time are selected.

3.5 Data splitting

Now, to protect the data from security breaches, the data sensed (d) from the registered nodes $(\Re)$ are split into n number of shares. Next, each share (γ) is given a parity bit (ρ), which is a check bit added for error detection. It is implemented to verify the data accuracy and allows the target to evaluate whether the data was accurately received. The parity bit added data (P) is expressed as,

$$P = \sum_{i=1}^{n} \gamma_i + \rho \tag{15}$$

The parity-added data is carried out for the encryption process.

3.6 Key agreement scheme-based encryption

Here, by employing the PPSECC algorithm, the parity bit added data P is encrypted. The key-based Elliptic Curve Cryptography (ECC) is selected for data encryption since it encrypts and decrypts the data faster and satisfies high security. But, the ECC can be vulnerable to key attacks, which leads to information loss. Therefore, to mitigate this issue and enhance data security, the ECC aims to pair public and private keys along with SK for every data transmission. The SK is created using a key agreement scheme, which is explained as follows.

3.6.1 Session key creation

Here, by utilizing the OSMVC-DH technique, the SK is created. To add perfect forward secrecy to secure transmission, the conventional Diffie Hellman (DH) method that securely exchanges the key during data transfer is used. However, the DH process does not contain an encryption step, which affects data security. Thus, the Vigenere Cipher (VC) text that alters the plain text into different cipher text at the exchanging phase is used along with DH. Moreover, to avoid the repetition of the key in VC, the One Sample Median (OSM), which interprets the median value of the ciphered text for encryption is used for altering the length of the key at each transmission.

Primarily, the private keys for both sender (Φ_1^{priv}) and receiver (Φ_2^{pri}) are chosen randomly over the curve. Public keys for both sender and receiver are calculated by considering the private keys. The public keys are generated as,

$$\Phi_1^{pub} = \Phi_1^{priv} * \text{mod}(\eta) \tag{16}$$

$$\Phi_2^{pub} = \Phi_2^{priv} * \text{mod}(\eta) \tag{17}$$

Here, Φ_1^{pub} and Φ_2^{pub} imply the public keys of the sender and receiver, respectively, and η signify the point on the curve. Now, both of them exchange their public keys. The exchanged key is converted into cipher text using VC. It applies a transferring mechanism that shifts every character using the Vigenere Table, which is a matrix of 26 rows and 26 columns. The VC encryption process is defined as,

$$\varsigma = \left(\Phi^{pub} + u\right) \text{mod } 26 \tag{18}$$

Here, ς is the converted cipher and u implies the key length, which is altered by the OSM as,

$$u = \frac{\Phi_1^{pub}/v_1 - \Phi_2^{pub}/v_2}{\sqrt{z(1-z)}\ 1/v_1 + 1/v_2} \tag{19}$$

Here, v_1 and v_2 implies the sample size, and z signifies the median parameter. Based on the cipher text, the SK (δ) is generated as,

$$\delta = (\varsigma) \text{mod}(\eta) \tag{20}$$

To increase the security of transmission, this SK is used in the encryption and decryption process of the PPSECC.

3.6.2 Data encryption

In data encryption using PPSECC, an elliptic curve is used for generating points, and the curve is defined as,

$$x^3 = a^3 + ay + b \tag{21}$$

Here, a, b implies integers, and x, y signifies the parameters that define the function. The sender encrypts the data with the receiver's public key and the receiver decrypts the data employing their private key. The private (λ_{priv}) and public keys (λ_{pub}) are generated to obtain encryption and decryption. The equation used to generate the public key is,

$$\lambda_{pub} = e * \lambda_{priv} \tag{22}$$

Here, e implies the point on the curve. Now, the input data is encrypted. The SK (δ) is multiplied by the encryption formula during the encryption process. The encrypted data consists of two cipher texts (χ), which are defined as,

$$\chi_1 = (e * \sigma) * \delta \tag{23}$$

$$\chi_2 = \left(P + \lambda_{pub} * \sigma\right) * \delta \tag{24}$$

Here, σ implies a random number of ranges (1, n-1). On the receiver side, the data can be decrypted using the equation,

$$P = \frac{\chi_2 - \hat{\lambda}_{priv} * \chi_1}{\delta} \tag{25}$$

In the end, the sensed data is encrypted using PPSECC and transmitted to the receiver through the optimal paths H_{opt}.

3.7 Hashcode generation

Here, for the purpose of data authorization, the sensed data, the sender and receiver's public keys, and the SK are converted into hashcodes using SC-SHA-512. Existing SHA-512 is a faster hashing technique. However, it has a collision problem. To overcome this problem, the Separate Chaining (SC) strategy is wielded in SHA-512.

Text of any length is converted into a fixed-size string by the SC-SHA-512. It works on a message in 1024-bit blocks and produces a 512-bit message. Primarily, the input is padded to a length of 896 mod 1024 and the message length is joined as a 128-bit binary number. The message is established into $n \times 1024 - bit$ blocks of data. Next, '8' working variables S, T, U, V, W, X, Y, Z are initialized as 64-bit words. 80 rounds of the SC-SHA-512 compression function are performed on the first 1024-bit block data. In Figure 2, the operation involved in the compression function is shown.

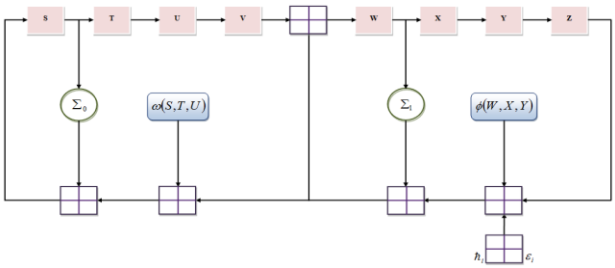

Figure 2. A round of the SC-SHA-512 compression function

The compression function performs logical functions AND, XOR. A round of the compression functions is described as follows,

$$\sum_0^{512}(S) = (S >>>> 28) \oplus (S >>>> 34) \oplus (S >>>> 39) \tag{26}$$

$$\omega(S,T,U) = (S \wedge T) \oplus (S \wedge U) \oplus (T \wedge U) \tag{27}$$

$$\phi_{2,i} = \sum_0^{512}(S) + \omega(S,T,U) \tag{28}$$

$$\sum_1^{512}(W) = (W >>>> 14) \oplus (W >>>> 18) \oplus (W >>>> 41) \tag{29}$$

$$\varphi(W,X,Y) = (W \wedge X) \oplus (\overline{W} \wedge Y) \tag{30}$$

$$\phi_{1,i} = Z + \sum_1^{512}(W) + \varphi(W,X,Y) + \varepsilon_i + \hbar_i \tag{31}$$

$$(Z_{i+1}, Y_{i+1}, X_{i+1}, W_{i+1}) = (Y_i, X_i, W_i, V_i + \phi_{1,i}) \tag{32}$$

$$(V_{i+1}, U_{i+1}, T_{i+1}, S_{i+1}) = (U_i, T_i, S_i, \phi_{1,i} + \phi_{2,i}) \tag{33}$$

Here, $\omega(S,T,U)$ and $\varphi(W,X,Y)$ imply the majority and choice of the bit-wise operations, respectively, ϕ signifies the summation of the functions performed, the term >>>> represents the right shifting, ε_i implies one among the 64-bit words, and $\hbar_i$ denotes the message schedule of 64-bit values. Each message block is passed through the message schedule, which is represented by 16 words and is expressed as,

$$\hbar_i = \begin{cases} \Phi_i^{pub} + \delta_i + d_i & 0 \le i \le 15 \\ \left(g_{1,i}^{512} + \hbar_{i-7} + g_{0,i}^{512} + \hbar_{i-16}\right)\Omega & 16 \le i \le 79 \end{cases} \tag{34}$$

Here, $g_{0,i}^{512}$ and $g_{1,i}^{512}$ signifies hash parameters, which are computed by the following expressions,

$$g_{0,i}^{512} = (\hbar_{i-15} >>>> 1) \oplus (\hbar_{i-15} >>>> 8) \oplus (\hbar_{i-15} >> 7) \tag{35}$$

$$g_{1,i}^{512} = (\hbar_{i-2} >>>> 19) \oplus (\hbar_{i-2} >>>> 61) \oplus (\hbar_{i-2} >> 6) \tag{36}$$

The term Ω implies the SC strategy. Instead of utilizing a list or linked list to chain a colliding bit, this strategy uses a binary search tree. SC is defined as,

$$\Omega(1+n) \to \Omega(\log n) \tag{37}$$

By using the 512-bit output from the first round, the eight variables of the subsequent data block are now initialized. The set of eight variables is rotated word-wise in each round. After all data blocks have been completed, the eight intermediate hash values are combined to produce the final 512-bit digest. The SC-SHA-512 procedure is elucidated in Algorithm 2.

Algorithm 2. SC-SHA-512

Input: sensed data, sender and receiver's public keys, session key
Output: Hashcode

Begin
  Initialize the eight working variables S, T, U, V, W, X, Y, Z
  For i=1 to n
  Prepare message schedule $\hbar_i$ using the equation,

$$\hbar_i = \begin{cases} \Phi_i^{pub} + \delta_i + d_i & 0 \le i \le 15 \\ g_{1,i}^{512} + \hbar_{i-7} + g_{0,i}^{512} + \hbar_{i-16} & 16 \le i \le 79 \end{cases}$$

  For i=1 to 80
    Perform compression with $\wedge, \oplus$ and >>>>
    Update the eight variables for i=i+1
  End for
    Compute the i^{th} intermediate hash value
    Generate the final 512-bit message digest
  End for
  Return Hashcode
End

For data and user authorization, this hashed output is sent to the receiver.

3.8 Data authentication

Then, by using PPSECC, the receiver decrypts the data.

Next, the parity bits added to the data are removed from the decrypted data. Moreover, data reconstruction is performed in which the split shares are combined. The reconstructed data along with the SK and sender and receiver's public keys are converted into hashcode and the resulting hash is verified with the hash, which is received from the sender. If both the hashs match, then the transmission is successful. If they do not match, then the data reconstruction is performed again.

Hence, the proposed key agreement scheme-based framework securely transmits the data. The proposed methodology's efficacy is evaluated in the following section.

4. RESULTS AND DISCUSSION

Here, regarding real-time network processing, the proposed Security aware data transmission scheme's performance is assessed with the existing frameworks. In the working platform of PYTHON, the proposed methodology is implemented.

4.1 Performance analysis of secure transmission

The proposed PPSECC is validated with prevailing ECC, Rivest-Shamir-Adleman (RSA), Data Encryption Standard (DES), and ElGamal to state the effectiveness of the model.

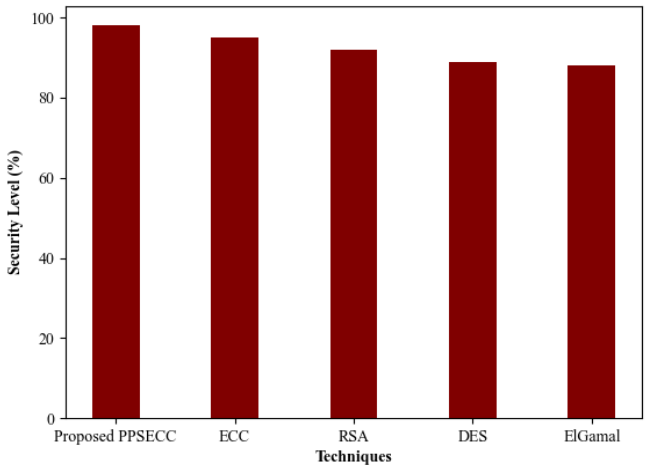

Figure 3. Performance validation based on the security level

The security level of the proposed PPSECC is 98% in Figure 3. However, the existing ElGamal, DES, RSA, and ECC show 88%, 89%, 92%, and 95%, respectively. The proposed model obtained a higher security level than the existing techniques. Thus, the PPSECC securely transfers the data from the sender to the receiver end than the existing models. The inclusion of the SK, which is created for each transmission using the OSMVC-DH algorithm, has enhanced the security of the transmission process to a greater extent.

Table 1. Comparative analysis of proposed PPSECC and Existing methods

Technique/Metrics	Overhead (bits)	Encryption Time (ms)	Decryption Time (ms)
Proposed PPSECC	126.35475	1137	1254
ECC	130.64577	2024	2478
RSA	132.565454	3012	3125
DES	132.24324	4025	4154
ElGamal	141.4444	5301	5468

The encryption overhead, encryption time, together with decryption time consumed by the proposed PPSECC and the prevailing models are elucidated in Table 1. The encryption overhead shows the difference in the length of the encrypted data and the original data. The PPSECC exhibited an encryption overhead of 126.35475 bits, whereas the prevailing models, such as ECC, RSA, DES, and ElGamal exhibited greater overhead of 130.64577 bits, 132.565454 bits, 132.24324 bits, and 141.4444 bits than the proposed model. This proves the PPSECC's efficacy in encrypting the data. Likewise, when compared to the time consumed by the prevailing techniques, the PPSECC consumed a lower encryption and decryption time of 1137 ms and 1254 ms. But the existing methods encrypted and decrypted the data with an average encryption and decryption time of 3590.5 ms and 3806.25 ms, which are high when compared to the proposed technique. Thus, the OSMVC-DH-based PPSECC framework has securely encrypted and transmitted the data compared to the existing models.

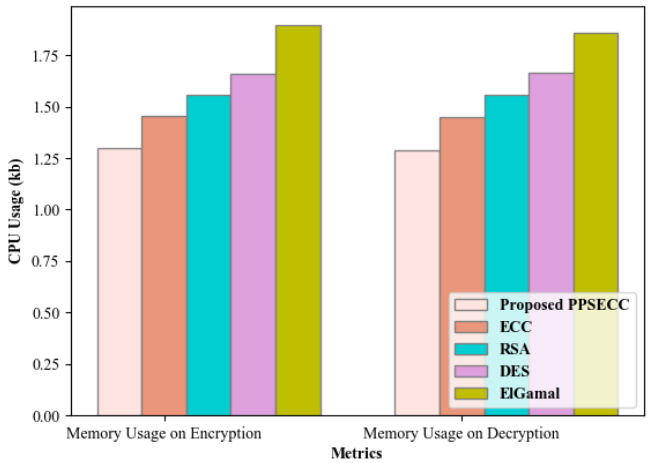

Figure 4. Performance analysis based on memory usage

In Figure 4, the memory used for encryption and decryption is illustrated. The memory used by the proposed technique for encrypting and decrypting is 129874545kb and 128755778kb, respectively. Whereas, the existing techniques like ECC and RSA use about 145454645kb and 155784541 kb of memory for encryption and 144878789kb and 155847752kb for decryption, respectively. Likewise, the memory occupied by other existing methods is also higher than the proposed technique. This lower memory usage of the PPSECC model during encryption and decryption shows the supremacy of the proposed model over the prevailing approaches.

4.2 Performance analysis of optimal path selection

Here, for proving the proposed work's worthiness, the proposed TW-COA's performance is analogized to the prevailing COA, Butterfly Optimization Algorithm (BOA), Sunflower Optimization Algorithm (SFOA), together with the Particle Swarm Optimization Algorithm (PSOA).

In Table 2, the response time, latency, and delay obtained by the proposed and the prevailing models are shown. For a varying number of nodes (100 to 500), the time-based performance comparison is evaluated. For proving the system's efficacy, the response time, latency, and delay must be low. Thus, for 500 nodes, the TW-COA consumed lower response time, latency, and delay of 4570 ms, 10847 ms, and 9325 ms, respectively. Meanwhile, the existing method

obtained an average of 5505 ms response time, 12111 ms latency, and 11351 ms delay for 500 nodes, which are higher than the proposed technique. This reveals the optimal path selection supremacy by the proposed TW-COA over the existing optimizer.

Table 2. Performance comparison of proposed TW-COA with existing methods

Metrics	No. of Nodes	PSOA	SFOA	BOA	COA	Proposed TW-COA
Response Time (ms)	100	2169	1869	1164	864	754
	200	3174	2874	2133	1769	1658
	300	4163	3963	3258	2763	2654
	400	4989	4789	4369	3648	3348
	500	6046	5846	5268	4863	4570
Latency (ms)	100	3875	3365	2987	2378	1875
	200	5745	5478	4875	4484	3847
	300	7765	7365	6841	6124	5784
	400	9765	9364	8894	8326	7854
	500	12875	12457	11874	11239	10847
Delay (ms)	100	3087	2896	2415	1865	1652
	200	4974	4756	4365	3847	3547
	300	7644	7436	6847	5748	5214
	400	9547	9284	8632	7763	7354
	500	13454	11634	10472	9847	9325

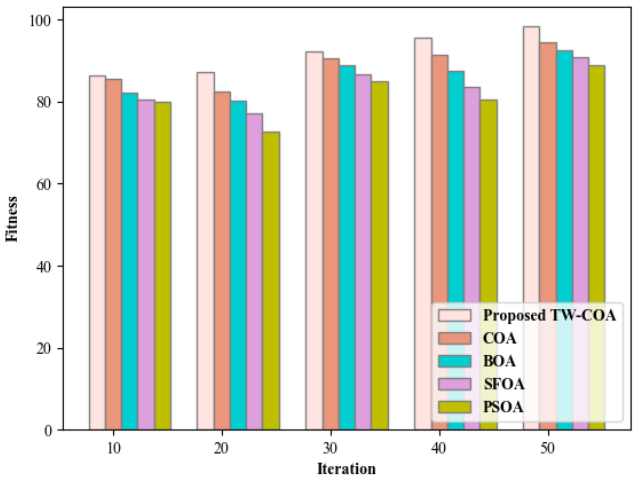

Figure 5. Fitness vs. iteration

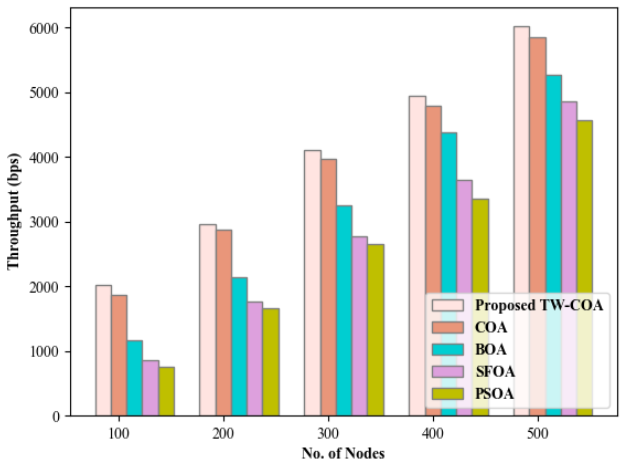

Figure 6. Throughput analysis

The performance measurement of the proposed TWS-COA and existing algorithms, such as COA, BOA, SFOA, and PSOA are exhibited in Figure 5. The fitness vs. iteration explains the ability of the scheme to detect the best-fit solution within a minimal iteration. Thus, the TWS-COA selected the optimal path at iteration 50 with a fitness value of 98.345. But, the existing techniques like COA, BOA, and PSOA select the optimal path with low fitness values of 94.456, 92.463, 90.751, and 88.849 for 50 iterations. Hence, the TWS-COA selected the best path than the existing methods.

The proposed framework's performance measure centered on its throughput is illustrated in Figure 6. A measure of units of information that a node can process in a given amount of time is termed the Throughput. For throughput, the proposed method exhibits a higher range of 2014bps to 6014bps for 100 to 500 numbers of nodes; while the prevailing systems exhibited lower throughput for varying numbers of nodes (100 to 500) in the range of 1162bps to 5136bps. Hence, the higher value of throughput by TWS-COA's performance shows the superiority over existing techniques in best path selection.

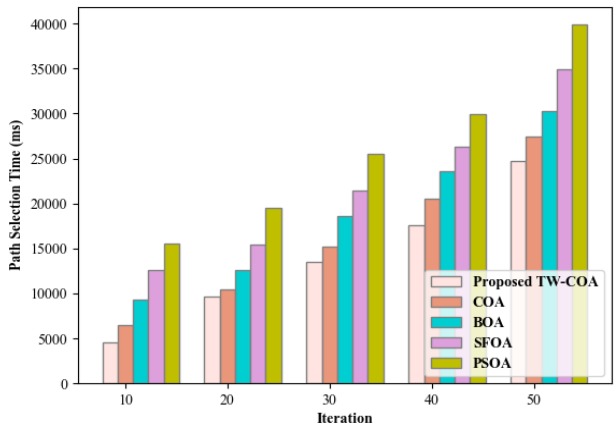

Figure 7. Performance comparison of TW-COA

The time required for selecting the optimal paths by the proposed TW-COA and existing models is elucidated in Figure 7. For 50 iterations, the proposed model has consumed 24754 ms for path selection time, while the prevailing optimization techniques consumed higher path selection time with an average of 33103 ms. The modification of the triangular walk strategy in conventional COA increased the exploration ability, which resulted in better performance of the proposed model when compared to the existing techniques.

4.3 Performance analysis of hashcode generation

In this, grounded on hashcode generation time, the performance of the user and data authorization phase is compared with conventional hashing techniques like SHA-512, Swift, Message Digest-5 (MD5), and tiger hash.

Table 3. Performance validation of proposed SC-SHA-512

Techniques	Hashcode Generation Time (ms)
Proposed SC-SHA-512	313
SHA-512	412
Swift	532
Tiger hash	624
MD5	784

Grounded on the time consumed for hashcode generation, the proposed system's performance is assessed with the existing techniques in Table 3. The Hashcode Generation

Time of the proposed SC-SHA-512 method is 313ms. Conversely, for the existing methods like SHA-512, SWIFFT, Tiger hash, and MD5, the hashcode generation time increases with a difference of 99ms, 219ms, 311ms, and 471ms, respectively. The collision avoidance using the Separate chaining strategy in the proposed hashing technique has exhibited superior performance in generating hashcodes than the prevailing hashing models.

In Figure 8, regarding verification time and SK generation time, the proposed method's performance validation is done. The SK, which was created using OSMVC-DH, was generated within 1600ms. As the SK plays a major role in securely transmitting the data, the lower time consumed for its generation results in a time reduction of the overall process. Moreover, to verify the hashcode for the purpose of user and data authorization, the proposed model just consumed 354ms.

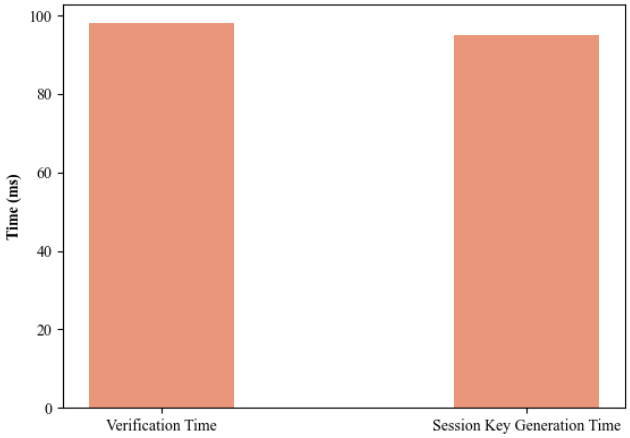

Figure 8. Performance regarding verification time and SK generation time

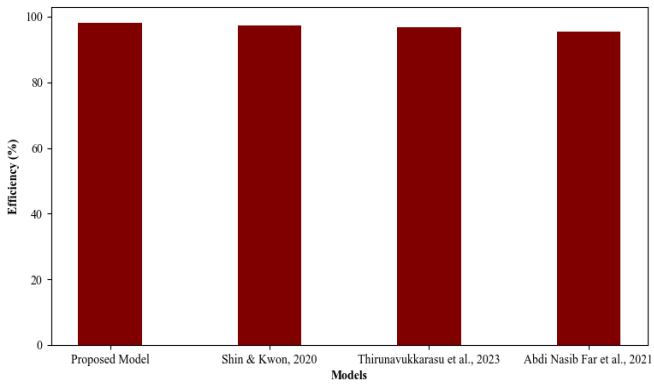

Figure 9. Efficiency analysis

4.4 Comparative analysis based on literature survey

Here, the proposed OSMVC-DH-based PPSECC model's performance is compared with the privacy techniques suggested by Shin et al. [11-13] based on their efficiency.

The efficiency of the proposed model and the existing models discussed in Section 2 is illustrated in Figure 9. A key agreement scheme is introduced in the proposed methodology for securely transmitting the data. Moreover, for authorization purposes, the data, session, and public keys are converted into hashcode which in turn improved the efficiency (98%) of the proposed work. But SK and node authentication are not mentioned even though privacy preserving is performed in the existing models that affect the data transmission's efficiency of about 97.3256% [11], 96.8798% [12] and 95.3265% [13]. Hence, the overall performance of the proposed secure data transmission methodology is better than the prevailing methods.

4.5 Statistical analysis

In this section, the statistical analysis of the proposed framework is done regarding the optimal path selection as depicted in Figure 10.

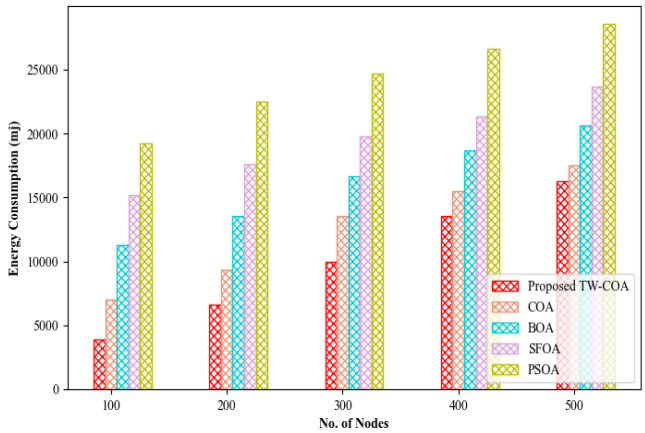

Figure 10. Statistical analysis regarding energy consumption

The proposed TW-COA model selected the optimal path and this path was used for transferring the data in the WSN. Due to the passing of data through the selected path, the energy consumption of the framework was a minimum of about 16248 mJ for transferring 500 nodes. But, the existing COA, BOA, SFOA, and PSOA consumed higher energy of 17548 mJ, 20658 mJ, 23658 mJ, and 28541 mJ, respectively than the proposed model for 500 nodes. Hence, the proposed model outperformed the existing models.

5. CONCLUSION

This paper proposed an effective architecture for secured authentication, authorization, and key agreement schemes for WSNs in 5G-integrated IoT. First, the node was registered and then authenticated using the LS2 Checksum method. The routing was done by using the GEAR model. Then, after routing, the optimal path was selected in 24754 ms with a 98.345% fitness value using TW-COA. Next, the data was encrypted in 1137 ms using the PPSECC technique. During encryption, the SK was created using the OSMVC-DH technique in 1600ms. Later, for a successful transaction, the Hashcode was generated using the SC-SHA-512 model in 310ms. The hashcode-matched data was finally decrypted in 1254ms. Thus, it is concluded that the proposed model effectively secured the data and transmitted the data successfully.

6. FUTURE RECOMMENDATION

Even though the proposed framework secured the data and transmitted the data through WSN efficiently, in real-world scenarios, there exist security and storage issues due to their centralized model. Thus, in the future, a hyperledger

blockchain can be employed for enhancing the performance of the proposed work.

REFERENCES

[1] Darbandeh, F.G., Safkhani, M. (2020). A new lightweight user authentication and key agreement scheme for WSN. Wireless Personal Communications, 114(4): 3247-3269. https://doi.org/10.1007/s11277-020-07527-4

[2] Jegadeesan, S., Azees, M., Ramesh Babu, N., Subramaniam, U., Almakhles, J.D. (2020). EPAW: Efficient privacy preserving anonymous mutual authentication scheme for wireless body area networks (WBANs). IEEE Access, 8: 48576-48586. https://doi.org/10.1109/ACCESS.2020.2977968

[3] Rajawat, A.S., Bedi, P., Goyal, S.B., Shukla, P.K., Jamal, S.S., Alharbi, A.R., Aljaedi, A. (2021). Securing 5G-IoT device connectivity and coverage using Boltzmann machine keys generation. Mathematical Problems in Engineering, 2021: 1-10. https://doi.org/10.1155/2021/2330049

[4] Masud, M., Gaba, G.S., Kumar, P., Gurtov, A. (2022). A user-centric privacy-preserving authentication protocol for IoT-AmI environments. Computer Communications, 196: 45-54. https://doi.org/10.1016/j.comcom.2022.09.021

[5] Polytechnic, F., State, O. (2021). Review techniques in energy conservation and sink node privacy preservation in wireless sensor networks. International Journal of Latest Technology in Engineering, Management & Applied Science, 10(11): 1-6.

[6] Rangwani, D., Sadhukhan, D., Ray, S., Khan, M. K., Dasgupta, M. (2021). An improved privacy preserving remote user authentication scheme for agricultural wireless sensor network. Transactions on Emerging Telecommunications Technologies, 32(3): 1-31. https://doi.org/10.1002/ett.4218

[7] Xia, X., Ji, S., Vijayakumar, P., Shen, J., Rodrigues, J.J. P.C. (2021). An efficient anonymous authentication and key agreement scheme with privacy preserving for smart cities. International Journal of Distributed Sensor Networks, 17(6): 1-13. https://doi.org/10.1177/15501477211026804

[8] Xie, Q., Ding, Z., Hu, B. (2021). A secure and privacy-preserving three-factor anonymous authentication scheme for wireless sensor networks in Internet of Things. Security and Communication Networks, 2021: 1-12. https://doi.org/10.1155/2021/4799223

[9] Wei, J., Phuong, T.V.X., Yang, G. (2021). An efficient privacy preserving message authentication scheme for Internet of Things. IEEE Transactions on Industrial Informatics, 17(1): 617-626. https://doi.org/10.1109/TII.2020.2972623

[10] Zhang, J., Zhang, Q., Li, Z., Lu, X., Gan, Y. (2021). A lightweight and secure anonymous user authentication protocol for wireless body area networks. Security and Communication Networks, 2021: 1-11. https://doi.org/10.1155/2021/4939589

[11] Shin, S., Kwon, T. (2020). A privacy-preserving authentication, authorization, and key agreement scheme for wireless sensor networks in 5G-integrated internet of things. IEEE Access, 8: 67555-67571. https://doi.org/10.1109/ACCESS.2020.2985719

[12] Thirunavukkarasu, V., Kumar, A.S., Prakasam, P., Suresh, G. (2023). Elliptic curve cryptography based key management and flexible authentication scheme for 5G wireless networks. Multimedia Tools and Applications, 1-15. https://doi.org/10.1007/s11042-023-14539-2

[13] Abdi Nasib Far, H., Bayat, M., Kumar Das, A., Fotouhi, M., Pournaghi, S.M., Doostari, M.A. (2021). LAPTAS: lightweight anonymous privacy preserving three factor authentication scheme for WSN-based IIoT. Wireless Networks, 27(2): 1389-1412. https://doi.org/10.1007/s11276-020-02523-9

[14] Meshram, C., Imoize, A.L., Elhassouny, A., Aljaedi, A., Alharbi, A.R., Jamal, S.S. (2021). IBOOST: A lightweight provably secure identity-based online/offline signature technique based on FCM for massive devices in 5G wireless sensor networks. IEEE Access, 9: 131336-131347. https://doi.org/10.1109/ACCESS.2021.3114287

[15] Wei, F., Vijayakumar, P., Kumar, N., Zhang, R., Cheng, Q. (2021). Privacy preserving implicit authentication protocol using cosine similarity for Internet of Things. IEEE Internet of Things Journal, 8(7): 5599-5606. https://doi.org/10.1109/JIOT.2020.3031486

[16] Hajian, R., Erfani, S.H. (2021). CHESDA: Continuous hybrid and energy efficient secure data aggregation for WSN. Journal of Supercomputing, 77(5): 1-31. https://doi.org/10.1007/s11227-020-03455-z

[17] Pothumarti, R., Jain, K., Krishnan, P. (2021). A lightweight authentication scheme for 5G mobile communications: a dynamic key approach. Journal of Ambient Intelligence and Humanized Computing, 1-19. https://doi.org/10.1007/s12652-020-02857-4

[18] Irin Loretta, G., Kavitha, V. (2021). Privacy preserving using multi-hop dynamic clustering routing protocol and elliptic curve cryptosystem for WSN in IoT environment. Peer-to-Peer Networking and Applications, 14(2): 821-836. https://doi.org/10.1007/s12083-020-01038-6

[19] Yu, S.J., Park, Y.H. (2020). SLUA-WSN: Secure and lightweight three-factor-based user authentication protocol for wireless sensor networks. Sensors, 20(15): 1-26. https://doi.org/10.3390/s20154143

[20] Sadhukhan, D., Ray, S., Obaidat, M.S., Dasgupta, M. (2021). A secure and privacy preserving lightweight authentication scheme for smart-grid communication using elliptic curve cryptography. Journal of Systems Architecture, 114: 101938. https://doi.org/10.1016/j.sysarc.2020.101938