The Scientific Temper (2024) Vol. 15 (spl): 182-190
E-ISSN: 2231-6396, ISSN: 0976-8653
Doi: 10.58414/SCIENTIFICTEMPER.2024.15.spl.22
https://scientifictemper.com/

RESEARCH ARTICLE

Enhanced malicious node identification in WSNs with directed acyclic graphs and RC4-based encryption

Rekha R.*, P. Meenakshi Sundaram

Abstract

In wireless sensor networks (WSNs), ensuring secure data transmission while preventing malicious activity is a critical challenge. This paper presents a novel approach for the identification of malicious nodes in WSNs by integrating directed acyclic graphs (DAGs) with the RC4 encryption algorithm. DAGs are employed to establish a hierarchical structure that enables efficient data flow and tracking of communication patterns across the network. By utilizing DAGs, the system can monitor the consistency and integrity of data transmission, making it easier to detect anomalies caused by malicious nodes. The RC4 encryption algorithm further strengthens the approach by securing the communication between nodes, preventing unauthorized access and tampering. In combination, DAGs and RC4 provide a robust framework for both detecting malicious nodes and securing data exchanges. Experimental simulations demonstrate that the proposed approach enhances network security by identifying compromised nodes with high accuracy while maintaining efficient communication and low overhead. This method offers a scalable and secure solution for protecting WSNs from malicious threats.

Keywords: Wireless sensor networks, Encryption technique, RC4, Directed acyclic graphs, Malicious node.

Introduction

Wireless sensor networks (WSNs) have emerged as a transformative technology, playing a crucial role in various applications, including environmental monitoring, healthcare, military operations, and smart cities. These networks consist of numerous sensor nodes that autonomously gather and transmit data over wireless communication channels. Despite their potential benefits, WSNs face significant challenges, particularly concerning security vulnerabilities. The distributed nature of these networks makes them susceptible to various attacks, including the introduction of malicious nodes, which can compromise data integrity, disrupt communication, and undermine the overall functionality of the network, Kandris, D., Nakas, C., Vomvas, D., & Koulouras, G. (2020), Sharma, S., Bansal, R. K., & Bansal, S. (2013, December), Sharma, S., Bansal, R. K., & Bansal, S. (2013, December).

Malicious nodes may act as intruders that perform hostile actions, such as data alteration, eavesdropping, or even launching denial-of-service attacks. The presence of such nodes can severely degrade network performance and reliability, making it essential to develop effective detection mechanisms to identify and isolate them promptly. Traditional security mechanisms often focus on encryption and authentication, but they may not adequately address the dynamic nature of WSNs, where nodes can fail or be compromised without warning. As a result, a holistic approach that combines anomaly detection and secure communication is imperative to safeguard WSNs from malicious threats, Temene, N., Sergiou, C., Georgiou, C., & Vassiliou, V. (2022), Gomathi, S., & Gopala Krishnan, C. (2020).

Directed acyclic graphs (DAGs) offer a promising structure for enhancing the security and efficiency of data communication in WSNs. By organizing nodes in a directed graph format, DAGs facilitate efficient data routing and enable the detection of irregular communication patterns. This hierarchical representation of nodes allows for better tracking of node interactions and data transmission flows, making it easier to identify nodes that exhibit suspicious behavior. Furthermore, the use of DAGs can improve

PG and Research Department of Computer Science, Maruthupandiyar College (Affiliated to Bharathidasan University, Tiruchirappalli), Thanjavur, Tamilnadu, India.

***Corresponding Author:** Rekha R., PG and Research Department of Computer Science, Maruthupandiyar College (Affiliated to Bharathidasan University, Tiruchirappalli), Thanjavur, Tamilnadu, India., E-Mail: npramki@gmail.com

How to cite this article: Rekha, R., Sundaram, P. M. (2024). Enhanced malicious node identification in WSNs with directed acyclic graphs and RC4-based encryption. The Scientific Temper, **15**(spl):182-190.

Doi: 10.58414/SCIENTIFICTEMPER.2024.15.spl.22

Source of support: Nil

Conflict of interest: None.

Received: 08/09/2024 **Accepted:** 23/09/2024 **Published**: 16/10/2024

network scalability, making it suitable for large-scale deployments, Jane Nithya, K., & Shyamala, K. (2022).

In addition to leveraging DAGs, employing robust encryption algorithms is vital for securing data exchanges between sensor nodes. The RC4 encryption algorithm, known for its simplicity and efficiency, provides a fast stream cipher suitable for resource-constrained environments typical of WSNs. By encrypting data transmitted across the network, RC4 not only ensures the confidentiality of the information but also serves as a deterrent against unauthorized access and manipulation by malicious entities. The combination of DAGs for structural integrity and RC4 for data security form a comprehensive approach to addressing the challenges posed by malicious nodes in WSNs.

This paper proposes a novel methodology that integrates directed acyclic graphs and the RC4 encryption algorithm to enhance the identification of malicious nodes within WSNs. By monitoring communication patterns through the DAG structure and securing data transmissions using RC4, the proposed approach aims to achieve high accuracy in malicious node detection while maintaining efficient network operation. The contributions of this work include an innovative framework for secure and reliable communication in WSNs, alongside empirical validation of its effectiveness through simulations.

Background Study of Malicious Node Detection

WSNs are increasingly used in various fields due to their capability to collect and disseminate data in real time. These networks consist of numerous sensor nodes, which communicate wirelessly to monitor environmental conditions or specific parameters in applications like smart cities, agriculture, healthcare, and military surveillance. However, the open nature of WSNs makes them vulnerable to a range of security threats, particularly from malicious nodes that can compromise the integrity and functionality of the network, Anand, C., & Vasuki, N. (2021), Ibrahim, D. S., Mahdi, A. F., & Yas, Q. M. (2021), Ibrahim, D. S., Mahdi, A. F., & Yas, Q. M. (2021), Ramasamy, L. K., KP, F. K., Imoize, A. L., Ogbebor, J. O., Kadry, S., & Rho, S. (2021).

Understanding of the Malicious Nodes

Malicious nodes can be defined as compromised or rogue sensor nodes that perform unauthorized actions within a WSN. They can disrupt normal operations through various attacks, such as Olakanmi, O. O., & Dada, A. (2020), Ramasamy, L. K., KP, F. K., Imoize, A. L., Ogbebor, J. O., Kadry, S., & Rho, S. (2021):

Data Manipulation

Malicious nodes may alter the data being transmitted, leading to incorrect information being relayed to the sink node or central processing unit. This can have dire consequences in applications such as healthcare, where accurate data is critical.

Denial of Service (DoS)

A malicious node may execute a DoS attack by overwhelming the network with false traffic or continuously sending erroneous data, thereby preventing legitimate nodes from communicating effectively.

Eavesdropping

Malicious nodes can intercept sensitive information being transmitted across the network, potentially leading to data breaches or unauthorized access to sensitive information.

Sybil Attacks

In this scenario, a single malicious node may present multiple identities to the network, thereby gaining an unfair advantage in resource consumption and manipulation of routing protocols.

Challenges in Malicious Node Detection

The detection of malicious nodes in WSNs is fraught with challenges due to the following factors:

Resource Constraints

Sensor nodes typically have limited computational power, memory, and battery life, which constrains the complexity of detection algorithms that can be deployed.

Dynamic Network Topology

WSNs are often characterized by frequent changes in topology due to node mobility, energy depletion, or node failures. This dynamism complicates the detection of malicious activities since the patterns of legitimate communication can vary widely over time.

Distributed Nature

The decentralized architecture of WSNs means that no single node has complete knowledge of the network's state, making it difficult to monitor interactions effectively and detect malicious behaviors.

Adversarial Behavior

Malicious nodes can mimic legitimate node behaviors, making it challenging for detection mechanisms to distinguish between genuine and malicious activities without incurring false positives.

Directed Acyclic Graph Approach

Directed acyclic graphs (DAGs) are a type of data structure that consists of nodes and directed edges, with the essential property that there are no cycles. This means that it is impossible to start at a node and follow a series of edges to return to the same node. DAGs have gained significant attention in the context of WSNs due to their ability to represent hierarchical structures and manage complex relationships between nodes efficiently, Prabakaran, R., & Arun, C. A. (2023).

In WSNs, DAGs can facilitate the representation of communication paths and data flow, allowing for effective

monitoring and analysis of interactions among sensor nodes. By utilizing DAGs, network administrators can track data transmission patterns, which is critical for detecting anomalies and identifying malicious nodes, Kably, S., Arioua, M., & Alaoui, N. (2022).

Nodes

Each sensor in the network is represented as a vertex in the graph.

Edges

Directed edges signify communication links, indicating the direction of data transmission.

This structure allows for efficient monitoring of data flow and identification of irregular communication patterns that may indicate the presence of malicious nodes.

Mathematical Representation of the DAGs

A DAG G can be formally represented as a pair $G = (V, E)$, where: V is the set of vertices (nodes), such that $V = \{v_1, v_2, \ldots, v_n\}$. E is the set of directed edges, where each edge (v_i, v_j) indicates a directed link from node v_i to the node v_j.

Properties of DAGs

Acyclic Nature

There are no cycles in the graph, ensuring that there is a clear, one-way flow of information.

Topological Sorting

A topological order of the nodes can be obtained, which is a linear ordering of vertices such that for every directed edge (v_i, v_j), v_i appears before v_j.

The topological sort is important for processing the nodes in a manner that respects their dependencies and communication flows.

Anomaly Detection Using DAGs

Detection of malicious nodes using a DAG-based approach involves monitoring the communication patterns and analyzing them for anomalies. The following steps outline the detection process:

Step 1: Establishing Communication Patterns

For each node in the DAG, we can track the amount of data transmitted over time. Let $D(v_i)$ represent the amount of data sent by node v_i in a given time interval T.

Step 2: Defining Anomaly Metrics

To detect anomalies, we can define metrics such as:

- *Data Rate*

The data rate of a node can be calculated as Where $R(v_i)$ is the data rate for node v_i, $D(v_i)$ is the total data sent by the node, and T is the time period over which the data was collected.

$$R(v_i) = \frac{D(v_i)}{T}$$

- *Expected Behavior*

Establish a threshold τ based on the average data rate of the network. This can be derived from historical data or established norms.

Step 3: Anomaly Detection Algorithm

The anomaly detection algorithm can be defined as follows:

- *Step 3.1: For each node v_i:*
- Calculate the data rate $R(v_i)$.
- If $R(v_i) > \tau$ (where τ is the threshold), flag v_i as a potential malicious node.

- *Step 3.2:*

Use statistical methods (e.g., z-score or standard deviation) to further validate anomalies, where μ is the mean data rate of the neighboring nodes. σ is the standard deviation of the data rates of the neighboring nodes.

$$Z(v_i) = \frac{R(v_i) - \mu}{\sigma}$$

If $|Z(v_i) > z_\alpha|$ where (z_α) is the critical value for a given significance level), classify v_i as malicious.

Step 4: Trust Management in DAGs

Combining the anomaly detection mechanism with a trust management system enhances the accuracy of malicious node identification:

- *Trust Score Calculation*

Each node v_i can maintain a trust score $T(v_i)$ based on interactions with neighboring nodes. The trust score can be updated using the following formula: where $S(v_i)$ is the trust score received from neighbors. α is the weighting factor $(0 < \alpha < 1)$ that balances the old trust score with new evidence.

$$T(v_i) = \alpha.T(v_i) + (1 - \alpha).S(v_i)$$

- *Malicious Node Identification*

A node is considered malicious if its trust score falls below a certain threshold T_{min}

If $T(v_i) < T_{min}, then\ v_i$ is flagged as malicious

Rc4-Based Encryption Approach

RC4 (Rivest Cipher 4) is a symmetric stream cipher known for its simplicity and efficiency. It was designed by Ron Rivest in 1987 and has been widely used in various applications, including SSL/TLS protocols for secure web traffic. RC4 is particularly suitable for WSNs due to its lightweight nature and low computational overhead, which is critical for resource-constrained sensor nodes, Alshawi, I., & Al-badrei, H. (2022).

In the context of malicious node detection in WSNs, RC4 can be utilized to secure communication between nodes, ensuring data confidentiality and integrity. This makes it challenging for malicious nodes to intercept, alter, or inject data into the network, Abdulhameed, H. A., Mosleh, M. F., Mohammed, A. T., & Abdulhameed, A. A. (2023, December).

RC4 Algorithm Structure

The RC4 encryption algorithm consists of two main processes:

Key Scheduling Algorithm (KSA)

This phase generates a permutation of all possible byte values (0-255) based on the secret key. The KSA initializes the state array S and a key array K.

Pseudo-Random Generation Algorithm (PRGA)

This phase generates the pseudo-random keystream based on the permutation created in the KSA. The keystream is then XORed with the plaintext to produce the ciphertext.

Key Scheduling Algorithm (KSA)

The KSA takes a key of length L (in bytes) and initializes the state array S of length 256. The steps are as follows:

Step 1: Initialize the state array:

$$S[i] = i \; for \; i = 0,1,2, \ldots ,255$$

Step 2: Key mixing

The array S is permuted based on the key: Swap $S[i]$ and $S[j]$

$$j = (j + S[i] + K[i \; mod \; L]) mod \; 256$$

This process results in a scrambled state array S.

Pseudo-Random Generation Algorithm (PRGA)

The PRGA generates the keystream from the state array S:

Step 1: Initialize: $i = 0, j = 0$

Step 2: For each byte of plaintext: Swap $S[i]$ and $S[j]$

$$i = (i + 1) \; mod \; 256$$

$$j = (j + S[i]) \; mod \; 256$$

Step 3: Generate the keystream byte: The output byte K is used to encrypt the plaintext.

$$K = S[(S[i] + S[j]) \; mod \; 256]$$

Encryption and Decryption Process

The encryption and decryption processes in RC4 are identical and involve the XOR operation:

Encryption

Given plaintext P: where C is the ciphertext.

$C[i] = P[i] \oplus K[i]$ for each byte i

Decryption

Given ciphertext C:

$P[i] = C[i] \oplus K[i]$ for each byte i

The operation ensures that the same keystream can be used for both encryption and decryption, making RC4 a symmetric cipher.

Proposed Enhanced Malicious Node Detection with Encryption-based Dag Approach

The integration of directed acyclic graphs (DAGs) with RC4-based encryption provides a robust framework for detecting malicious nodes in WSNs. This hybrid approach leverages the structural advantages of DAGs for monitoring data flow and the security benefits of RC4 encryption for safeguarding communication. Below is a detailed working process of this hybridization.

Overview of the Hybrid Approach

The hybridization process involves the following key components:

DAG Structure

Used to represent the communication topology of the WSN, enabling efficient monitoring of data transmission and anomaly detection.

RC4 Encryption

Provides data confidentiality and integrity during communication between nodes, ensuring that malicious nodes cannot easily intercept or tamper with the data.

The hybrid model facilitates the secure exchange of information while allowing for the real-time detection of anomalies that may indicate malicious activity.

System Components

Network Topology

- The WSN is organized in a DAG format, where each node represents a sensor device and directed edges represent communication paths.
- The base station (sink) is connected to multiple nodes, establishing a multi-path communication model.

Key Management

Each node generates and manages its encryption key for RC4. The keys should be periodically updated to enhance security.

Data Collection

Sensor nodes collect data and encrypt it using the RC4 algorithm before transmitting it through the DAG structure.

Procedure of Proposed Enhanced Malicious Node Detection with Encryption-based DAG approach

This procedure outlines the systematic steps for integrating DAGs with RC4 encryption to detect malicious nodes in Wireless Sensor Networks (WSNs).

Step 1: Network Initialization

- *Step 1.1: Define Network Parameters*
- Determine the number of nodes n.
- Define the key length L for RC4 encryption.
- *Step 1.2: Construct the DAG*
- Initialize a directed acyclic graph G where: Nodes V represent sensor devices and Edges E represent communication paths.
- For each node v_i: Generate a unique identifier. Initialize a trust score (e.g., $T(v_i) = 1.0$). Initialize a data rate variable (e.g., data_rate(v_i)=0).

- *Step 1.3: Establish Communication Links:*

For each node, identify its neighbors and add directed edges to the graph.

Step 2: Key Generation and Management

- *Step 2.1: Generate Encryption Keys: For each node v_i:*

Generate a random key K_i of length L for RC4 encryption.

- *Step 2.2: Securely Share Keys*

Distribute keys to direct neighbors while ensuring confidentiality (using a secure channel if necessary).

Step 3: Data Collection and Transmission

- *Step 3.1: Collect Sensor Data*

Each node v_i gathers environmental data (e.g., temperature, humidity).

- *Step 3.2: Encrypt the Data*

Before transmission, encrypt the collected data P using the RC4 algorithm: $C = P \oplus K_i$ where C is the cipher text, P is the plain text and K_i is the keystream generated from the RC4 algorithm.

- *Step 3.3: Transmit Encrypted Data*

Send the encrypted data C to all neighbouring nodes $N(v_i)$ in the DAG.

Step 4: Monitor Data Transmission

- *Step 4.1: Data Rate Monitoring*

Each node monitors the incoming encrypted data packets:

- Count the number of packets received over a specific time interval.
- Update the data rate for each node v_i:

$data\_rate(v_i) = CountIncomingData(v_i)$

Step 5: Anomaly Detection

- *Step 5.1: Calculate Expected Data Rate*

For each node $\mathbf{v_i}$

- Compute the expected data rate based on historical data or average rates from neighboring nodes.

- *Step 5.2: Perform Statistical Analysis*

Calculate the Z-score to identify anomalies: where μ is the mean and α is the standard deviation of the data rates of neighboring nodes.

$$Z(v_i) = \frac{data\_rate(v_i) - \mu}{\alpha}$$

- *Step 5.3: Flag Suspicious Nodes*

If $|Z(v_i) > z_\alpha|$ where z_α is a predefined threshold), mark node v_i as suspicious.

Step 6: Trust Score Update

- *Step 6.1: Update Trust Scores*

For each suspicious node v_i and for neighbouring node v_j : update the trust score of v_j based on its interactions with v_i: Where $S(v_j)$ is the status of node v_j (0 for suspicious, 1 for normal)

$$T(v_j) = \alpha . T(v_j) + (1 - \alpha) . S(v_j)$$

Step 7: Identify Malicious Nodes

- *Step 7.1: Determine Malicious Status*

For each node v_i in the network:

- If $T(v_j) < T_{min}$ (where T_{min} is a trust threshold), mark v_j as malicious.

Result And Discussion

The performance of the proposed hybrid DAG+RC4 approach can be evaluated with the existing symmetric encryption techniques like RC4, advanced encryption standard (AES), and triple data encryption standard (3DES) with varying numbers of nodes. The evaluation metrics like detection rate (in %), packet delivery ratio (in %), packet loss (in %), end-to-end delay (in ms), energy consumption (in Joules), throughput (in mbps) are used in this research work.

Performance Analysis with 10% Malicious Node

Table 1 depicts the detection rate (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious and varying numbers of nodes.

From Table 1, the proposed hybrid DAG+RC4 approach consistently demonstrates a high detection rate, starting from 96.34% at 80 nodes and reaching 97.89% at 150 nodes. AES also shows strong performance, with detection rates ranging from 90.24 to 92.89% as the number of nodes increases. RC4 exhibits moderate performance, ranging from 88.12% to 90.78%, indicating it is less effective than both the hybrid and AES approaches. 3DES has the lowest detection rates across all node sizes, starting at 82.47% and increasing to 84.68%.

Table 2 depicts the packet delivery ratio (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious and varying number of nodes.

Table 1: Detection Rate (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious node

Number of nodes	*Detection rate (in %)*			
	Proposed DAG+RC4	*RC4*	*AES*	*3DES*
80	96.34	88.12	90.24	82.47
90	96.56	88.39	90.58	82.78
100	96.78	89.00	91.02	83.10
110	97.01	89.25	91.45	83.42
120	97.23	89.54	91.78	83.65
130	97.45	90.01	92.12	84.01
140	97.67	90.34	92.45	84.32
150	97.89	90.78	92.89	84.68

Table 2: Packet delivery ratio (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious node

Number of nodes	*Packet delivery ratio (in %)*			
	Proposed DAG+RC4	*RC4*	*AES*	*3DES*
80	95.12	85.32	88.67	82.14
90	95.34	85.64	89.01	82.48
100	95.56	86.00	89.34	82.80
110	95.78	86.32	89.67	83.10
120	96.01	86.67	90.01	83.45
130	96.23	87.00	90.34	83.67
140	96.45	87.34	90.67	84.00
150	96.67	87.67	91.01	84.35

Table 3: Packet loss (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious node

Number of nodes	*Packet loss (in %)*			
	Proposed DAG+RC4	*RC4*	*AES*	*3DES*
80	4.88	14.68	11.33	17.86
90	4.66	14.36	10.99	17.52
100	4.44	14.00	10.66	17.20
110	4.22	13.68	10.33	16.90
120	3.99	13.33	9.99	16.55
130	3.77	13.00	9.66	16.33
140	3.55	12.66	9.33	16.00
150	3.33	12.33	9.00	15.65

From Table 2, the proposed hybrid DAG+RC4 approach achieves a consistently high packet delivery ratio (PDR), starting from 95.12% at 80 nodes and increasing to 96.67% at 150 nodes. AES performs well with PDR values ranging from 88.67 to 91.01%, although it remains below the hybrid approach. RC4 demonstrates moderate performance, with a PDR between 85.32 and 87.67% across the various node sizes. 3DES shows the lowest PDR, starting at 82.14% and increasing to 84.35%, indicating its limitations compared to the other algorithms.

Table 3 depicts the packet loss ratio (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious and varying number of nodes.

From Table 3, the proposed hybrid DAG+RC4 approach exhibits the lowest packet loss percentage, starting at 4.88% at 80 nodes and decreasing to 3.33% at 150 nodes. AES shows moderate performance, with packet loss ranging from 10.99 to 9.00%, indicating it has a significantly higher loss rate compared to the hybrid approach. RC4 displays higher packet loss rates, ranging from 14.68 to 12.33%, indicating its inefficiency in dealing with malicious nodes compared to the proposed method. 3DES has the highest packet loss across all node sizes, ranging from 17.86 to 15.65%, reflecting its limitations in maintaining packet integrity in the presence of malicious nodes.

Table 4: End-to-end delay (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious node

Number of nodes	*End-to-end delay (in ms)*			
	Proposed DAG+RC4	*RC4*	*AES*	*3DES*
80	35.67	55.34	50.23	65.45
90	36.02	56.10	51.00	66.12
100	36.38	56.78	51.78	66.85
110	36.73	57.45	52.56	67.50
120	37.10	58.12	53.33	68.20
130	37.46	58.80	54.01	68.85
140	37.82	59.45	54.78	69.50
150	38.10	60.12	55.34	70.20

Table 5: Energy consumption (in Joules) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious node

Number of nodes	*Energy consumption (in Joules)*			
	Proposed DAG+RC4	*RC4*	*AES*	*3DES*
80	1.25	1.85	1.65	2.10
90	1.28	1.87	1.68	2.15
100	1.30	1.90	1.70	2.20
110	1.32	1.93	1.73	2.25
120	1.35	1.95	1.75	2.30
130	1.37	1.98	1.78	2.35
140	1.40	2.00	1.80	2.40
150	1.42	2.02	1.83	2.45

Table 4 depicts the end-to-end delay (in ms) obtained by the Proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious and varying number of nodes.

From Table 4, the proposed hybrid DAG+RC4 approach consistently achieves the lowest end-to-end delay, starting from 35.67 ms at 80 nodes and increasing to 38.10 ms at 150 nodes. AES shows moderate delay performance, with end-to-end delays ranging from 50.23 to 55.34 ms, which is significantly higher than the hybrid approach. RC4 exhibits even higher delays, ranging from 55.34 to 60.12 ms, indicating its inefficiency in routing data through the network. 3DES has the highest end-to-end delay across all node sizes, ranging from 65.45 to 70.20 ms, reflecting its limitations in handling communication efficiently.

Table 5 depicts the energy consumption (in Joules) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious and varying number of nodes.

From Table 5, The Proposed Hybrid DAG+RC4 approach demonstrates the lowest energy consumption, starting from 1.25 J at 80 nodes and increasing to 1.42 J at 150 nodes.

Table 6: Throughput (in mbps) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious node

Number of nodes	*Throughput (in mbps)*			
	Proposed DAG+RC4	*RC4*	*AES*	*3DES*
80	22.35	12.85	15.47	9.85
90	22.67	13.00	15.78	10.10
100	23.01	13.20	16.05	10.30
110	23.35	13.40	16.45	10.55
120	23.67	13.60	16.78	10.80
130	24.00	13.80	17.01	11.05
140	24.35	14.00	17.35	11.30
150	24.67	14.20	17.68	11.55

AES shows moderate energy consumption, with values ranging from 1.65 to 1.83 J, indicating it consumes more energy compared to the hybrid approach. RC4 exhibits higher energy consumption, ranging from 1.85 to 2.02 J, suggesting its inefficiency in energy utilization during data transmission. 3DES has the highest energy consumption across all node sizes, ranging from 2.10 to 2.45 J, highlighting its significant overhead in energy usage.

Table 6 depicts the throughput (in mbps) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious and varying numbers of nodes.

From Table 6, the proposed hybrid DAG+RC4 approach achieves the highest throughput, starting from 22.35 Mbps at 80 nodes and increasing to 24.67 Mbps at 150 nodes.

AES performs moderately well, with throughput ranging from 15.47 to 17.68 Mbps, but it is still lower than the hybrid approach. RC4 shows lower throughput, ranging from 12.85 to 14.20 Mbps, indicating a significant reduction in data transmission efficiency. 3DES has the lowest throughput across all node sizes, starting at 9.85 Mbps and reaching 11.55 Mbps, highlighting its limitations in achieving high data rates.

Performance Analysis with 20% Malicious Node

Table 7 depicts the detection Rate (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 20% malicious and varying number of nodes.

From Table 7, The Proposed Hybrid DAG+RC4 approach consistently achieves a high Detection Rate, starting from 87.23% at 80 nodes and increasing to 89.10% at 150 nodes.

AES demonstrates moderate detection capabilities, with rates ranging from 64.35 to 68.05%, indicating its lower efficiency in detecting malicious nodes compared to the hybrid approach.

RC4 shows even lower detection rates, ranging from 69.81 to 72.15%, reflecting its limitations in effectively identifying malicious nodes in the network. 3DES has the lowest detection rates across all node sizes, ranging from 58.11 to 55.30%, highlighting its inefficiency in detecting malicious activity.

Table 7: Detection rate (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 20% malicious node

Number of Nodes	*Detection Rate (in %)*			
	Proposed DAG+RC4	*RC4*	*AES*	*3DES*
80	87.23	70.54	65.78	58.11
90	86.75	69.81	64.35	57.45
100	88.10	71.24	66.88	56.60
110	87.54	70.45	65.40	55.82
120	88.35	71.10	67.11	56.99
130	87.80	70.99	66.25	55.50
140	88.95	71.90	67.90	54.75
150	89.10	72.15	68.05	55.30

Table 8: Packet delivery ratio (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 20% malicious node

Number of nodes	*Packet delivery ratio (in %)*			
	Proposed DAG+RC4	*RC4*	*AES*	*3DES*
80	92.47	74.22	70.85	65.30
90	91.78	73.00	68.99	64.15
100	93.12	75.45	71.40	66.55
110	92.85	74.55	69.67	63.50
120	93.47	76.20	72.11	66.90
130	91.23	74.88	70.25	62.75
140	92.05	75.30	71.78	65.00
150	93.67	77.10	73.50	64.40

Table 8 depicts the packet delivery ratio (in %) obtained by the Proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 20% malicious and varying number of nodes.

From Table 8, the proposed hybrid DAG+RC4 approach consistently achieves a high Packet Delivery Ratio, starting from 92.47% at 80 nodes and increasing to 93.67% at 150 nodes. AES demonstrates moderate delivery ratios, ranging from 68.99 to 73.50%, indicating it is less effective in maintaining packet delivery compared to the hybrid approach. RC4 shows lower delivery ratios, ranging from 73.00 to 77.10%, reflecting its limitations in ensuring effective packet delivery in the presence of malicious nodes. 3DES has the lowest delivery ratios across all node sizes, ranging from 64.15 to 66.90%, indicating its inefficiency in delivering packets in a compromised network environment.

Table 9 depicts the packet loss ratio (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious and varying number of nodes.

From Table 9, the proposed hybrid DAG+RC4 approach maintains the lowest packet loss ratio, ranging from 5.99% at 150 nodes to 8.14% at 110 nodes, demonstrating its effectiveness in minimizing packet loss. AES shows higher packet loss ratios, fluctuating between 26.85 and 32.11%,

Table 9: Packet loss (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious node

Number of nodes	*Packet loss (in %)*			
	Proposed DAG+RC4	*RC4*	*AES*	*3DES*
80	7.45	25.67	30.22	34.56
90	6.88	26.55	31.10	36.78
100	7.29	24.78	29.95	35.34
110	8.14	25.82	32.11	33.99
120	6.45	23.94	28.65	37.12
130	7.88	24.34	29.20	34.88
140	6.22	22.79	27.40	35.00
150	5.99	21.15	26.85	38.45

Table 10: End-to-end delay (in %) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious node

Number of nodes	*End-to-end delay (in ms)*			
	Proposed DAG+RC4	*RC4*	*AES*	*3DES*
80	25.33	47.22	52.11	60.45
90	24.88	48.14	53.57	61.20
100	26.15	49.67	54.10	62.55
110	25.70	50.25	51.89	59.88
120	24.44	48.89	53.10	63.15
130	25.82	49.20	52.45	60.90
140	24.99	50.55	55.23	62.70
150	23.88	51.13	54.67	64.11

indicating a less robust performance in handling malicious activities compared to the hybrid approach. RC4 displays even higher packet loss ratios, ranging from 21.15 to 26.55%, highlighting its limitations in maintaining effective communication under attack conditions. 3DES consistently has the highest packet loss ratios across all node sizes, varying from 33.99 to 38.45%, reflecting significant inefficiency in packet delivery in the presence of malicious nodes.

Table 10 depicts the end-to-end delay (in ms) obtained by the Proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious and varying number of nodes.

From Table 10, The Proposed Hybrid DAG+RC4 approach consistently achieves lower end-to-end delay, starting from 23.88 ms at 150 nodes and ranging up to 26.15 ms at 100 nodes, indicating its efficiency in maintaining fast communication. AES experiences moderate delays, with values ranging from 51.89 to 54.67 ms, reflecting a slower response time under the influence of malicious nodes compared to the hybrid method. RC4 exhibits higher end-to-end delays, fluctuating between 47.22 and 51.13 ms, which shows its limitations in quickly processing data in the presence of threats. 3DES has the highest end-to-end delays across all node sizes, ranging from 60.45 to 64.11 ms, indicating its inefficiency in providing timely responses in a compromised network environment.

Table 11: Energy consumption (in Joules) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious node

Number of nodes	*Energy consumption (in Joules)*			
	Proposed DAG+RC4	*RC4*	*AES*	*3DES*
80	0.25	0.68	0.73	0.82
90	0.24	0.69	0.75	0.83
100	0.26	0.70	0.72	0.84
110	0.25	0.71	0.76	0.81
120	0.23	0.67	0.74	0.85
130	0.24	0.68	0.77	0.80
140	0.22	0.66	0.73	0.86
150	0.21	0.65	0.78	0.87

Table 12: Throughput (in mbps) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious node

Number of nodes	*Throughput (in mbps)*			
	Proposed DAG+RC4	*RC4*	*AES*	*3DES*
80	42.57	25.34	21.89	18.76
90	43.12	24.88	20.67	17.45
100	41.78	26.11	22.34	19.08
110	42.23	25.56	21.12	18.12
120	43.45	23.78	20.45	17.34
130	42.67	24.12	21.56	18.98
140	44.12	25.45	19.78	16.78
150	45.34	22.89	20.01	17.56

Table 11 depicts the energy consumption (in Joules) obtained by the proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious and varying numbers of nodes.

From Table 11, the proposed hybrid DAG+RC4 approach demonstrates the lowest energy consumption, ranging from 0.21 J at 150 nodes to 0.26 J at 100 nodes, indicating its efficiency in energy usage. AES shows moderate energy consumption, with values ranging from 0.72 to 0.78 J, indicating higher energy requirements in the presence of malicious nodes compared to the hybrid approach. RC4 exhibits higher energy consumption, fluctuating between 0.65 and 0.71 J, suggesting its limitations in energy efficiency when processing data with threats. 3DES consistently has the highest energy consumption across all node sizes, ranging from 0.82 to 0.87 J, indicating significant inefficiency in energy usage under compromised network conditions.

Table 12 depicts the throughput (in mbps) obtained by the Proposed DAG+RC4 approach and existing RC4, AES, and 3DES with 10% malicious and varying numbers of nodes.

From Table 12, The Proposed Hybrid DAG+RC4 approach consistently achieves the highest throughput, ranging from 41.78 Mbps at 100 nodes to 45.34 Mbps at 150 nodes, indicating its efficiency in data transmission. RC4 shows moderate throughput, with values fluctuating between 22.89 and 26.11 Mbps, demonstrating lower performance in handling data under malicious conditions compared to the hybrid method. AES has lower throughput, ranging from 19.78 to 22.34 Mbps, indicating its inefficiency in maintaining effective data transmission. 3DES consistently exhibits the lowest throughput across all node sizes, with values ranging from 16.78 to 19.08 Mbps, reflecting significant limitations in its data handling capabilities in a compromised network environment.

Conclusion

The proposed hybrid DAG+RC4 approach for detecting malicious nodes in WSNs has demonstrated significant advantages over traditional algorithms such as 3DES, AES, and RC4. Through a comprehensive evaluation of key performance metrics—including detection rate, packet delivery ratio, packet loss ratio, end-to-end delay, energy consumption, and throughput—the proposed method has shown superior effectiveness and efficiency in maintaining network security and performance.

The Hybrid DAG+RC4 approach consistently achieved higher detection rates, effectively identifying malicious nodes even as network size increased. This capability is vital for safeguarding the integrity of the network. The proposed method maintained a high packet delivery ratio, indicating robust communication reliability, whereas existing algorithms experienced higher rates of packet loss. The hybrid approach exhibited lower end-to-end delays, ensuring quicker communication, which is essential for real-time applications. The proposed method also demonstrated superior energy efficiency, consuming less power while delivering optimal performance. In contrast, the existing algorithms consumed significantly more energy, potentially limiting their operational lifespan in sensor networks. The Hybrid DAG+RC4 approach maintained the highest throughput, ensuring efficient data transmission even in the presence of malicious activities, whereas the existing methods struggled to achieve similar levels.

References

Abdulhameed, H. A., Mosleh, M. F., Mohammed, A. T., & Abdulhameed, A. A. (2023, December). A lightweight hybrid cryptographic algorithm for WSN security using the Raspberry Pi as a node. *In AIP Conference Proceedings* (Vol. 2834, No. 1). AIP Publishing.

Alshawi, I., & Al-badrei, H. (2022). Secure Routing Protocol for WSNs Using Bacterial Foraging Optimization and Improved RC4. *Informatica*, 46(8).

Anand, C., & Vasuki, N. (2021). Trust based DoS attack detection in wireless sensor networks for reliable data transmission. *Wireless Personal Communications*, 121(4), 2911-2926.

Fahmy, H. M. A., & Fahmy, H. M. A. (2020). Energy Management Projects for WSNs. *Wireless Sensor Networks: Energy Harvesting and Management for Research and Industry*, 611-637.

Gomathi, S., & Gopala Krishnan, C. (2020). Malicious node detection in wireless sensor networks using an efficient secure data aggregation protocol. *Wireless Personal Communications*, 113(4), 1775-1790.

Ibrahim, D. S., Mahdi, A. F., & Yas, Q. M. (2021). Challenges and issues for wireless sensor networks: A survey. *J. Glob. Sci. Res*, 6(1), 1079-1097.

Jane Nithya, K., & Shyamala, K. (2022). A systematic review on various attack detection methods for wireless sensor networks. *In International Conference on Innovative Computing and Communications: Proceedings of ICICC 2021*, Volume 3 (pp. 183-204). Springer Singapore.

Kably, S., Arioua, M., & Alaoui, N. (2022). Lightweight Direct Acyclic Graph Blockchain for Enhancing Resource-Constrained IoT Environment. *Computers, Materials & Continua*, 71(3).

Kandris, D., Nakas, C., Vomvas, D., & Koulouras, G. (2020). Applications of wireless sensor networks: an up-to-date survey. *Applied system innovation*, 3(1), 14.

Olakanmi, O. O., & Dada, A. (2020). Wireless sensor networks (WSNs): Security and privacy issues and solutions. *Wireless mesh networks-security, architectures and protocols*, 13, 1-16.

Prabakaran, R., & Arun, C. A. (2023). Hybrid Genetic Algorithm with Directed Acyclic Graph for Enhancing Data Transmission and Reducing Energy Hole Problem.

Ramasamy, L. K., KP, F. K., Imoize, A. L., Ogbebor, J. O., Kadry, S., & Rho, S. (2021). Blockchain-based wireless sensor networks for malicious node detection: A survey. *IEEE Access*, 9, 128765-128785.

Sharma, S., Bansal, R. K., & Bansal, S. (2013, December). Issues and challenges in wireless sensor networks. *In 2013 international conference on machine intelligence and research advancement* (pp. 58-62). IEEE.

Temene, N., Sergiou, C., Georgiou, C., & Vassiliou, V. (2022). A survey on mobility in wireless sensor networks. *Ad Hoc Networks*, 125, 102726.