Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 7252791, 11 pages
https://doi.org/10.1155/2022/7252791

Research Article

Node Replication Attack Detection in Distributed Wireless Sensor Networks

L. Sujihelen,[1] **Rajasekhar Boddu**,[2] **S. Murugaveni**,[3] **Ms. Arnika**,[4] **Anandakumar Haldorai**,[5] **Pundru Chandra Shaker Reddy**,[6] **Suili Feng**,[7] **and Jiayin Qin**[7]

[1]*Sathyabama Institute of Science and Technology, Chennai, India*
[2]*Department of Software Engineering, College of Computing and Informatics, Haramaya University, Dire Dawa, Ethiopia*
[3]*Department of ECE, SRM Institute of Science and Technology, India*
[4]*Department of Computer Science and Engineering, Faculty of Engineering and Technology, SRM Institute of Science and Technology NCR Campus, Modinagar, Ghaziabad, Uttar Pradesh, India*
[5]*Department of Computer Science and Engineering, Sri Eshwar College of Engineering, Coimbatore, Tamil Nadu, India*
[6]*Department of Computer Science and Engineering, CMR College of Engineering and Technology, Telangana, Hyderabad, India*
[7]*Computer, Electrical and Mathematical Sciences and Engineering Division, King Abdullah University of Science and Technology (KAUST), Thuwal 23955-6900, Saudi Arabia*

Correspondence should be addressed to Rajasekhar Boddu; rajsekhar.boddu@haramaya.edu.et

Received 25 February 2022; Revised 18 April 2022; Accepted 9 May 2022; Published 31 May 2022

Academic Editor: Junjuan Xia

Wireless sensor network (WSN) is an emerging technology used in emergency scenarios. There are a number of possible threats to WSNs because they use unsupervised IP addresses. Securing networks with unattended sensors is a real challenge nowadays. Sensor nodes lack power and storage, making them incompatible with normal security checks. It will be vital to make advancements in sensor network architecture and protocol design. There will be more vulnerability to attack if there is a lack of security. Especially, one key attack is node replication which induces the sensor node to acts as an original node, collecting data from the network and sending it to the attacker. In dynamic WSN, detecting an assault is difficult to find replica nodes. Therefore, this paper proposes a Strategic Security System (SSS) to discover replica nodes in static and dynamic distributed WSNs. It is mainly focused on enhancing detection accuracy, time delay, and communication overhead. The present system includes Single Stage Memory Random Walk with Network Division (SSRWND) and a Random-walk-based approach to detect clone attacks (RAWL). The proposed system has less memory and better detection accuracy.

1. Introduction

A real challenge in the present day is making the networks secure while dealing with unattended sensors. Sensor nodes are incompatible with routine security inspections due to their inherent power and storage limitations. Emerging innovations in designing the architecture of sensor networks and inventing new protocols will gain significant importance in the future. These advancements also invite new types of risks and attacks to affect the integrity of WSNs. It is important to detect and prevent node replication attacks in WSNs. Researchers' schemes for detecting threats and preventing them are presented in this study. The existing researchers have classified the detection schemes into two categories: static and mobile WSNs. Researchers have proposed various techniques for distributed wireless sensor networks.

Node replication attack is an active attack in WSN [1]. The replication attack is the root cause of many attacks in WSN. The replica node will behave like an original node and sense confidential data from the sensor networks [2, 3]. The overview of the replication attack is shown in Figure 1. The replicated node directs the target to attack.

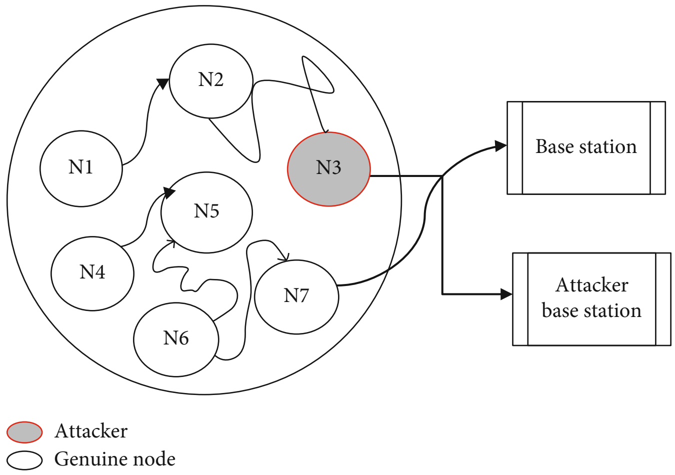


Figure 1: Overview of the replication attacks.

With a centralized approach, it is easy to detect node replication attacks. Monitor nodes send alarm messages when replicated nodes are present. There has been some analysis of detection schemes for centralized approaches.

The first solution for detecting a clone attack is based on a base station. The same node ID with a different location will identify the cloned node. The drawbacks are high communication costs and less detection accuracy. In addition to random key predistribution, the predistribution of keys is subject to certain conditions [4]. The node is assigned as a replicated node if it exceeds the threshold or criteria for key usage. The base station will count how many times it has been used. The drawback is an imbalance of message transfers. The SET approach is another method for analyzing and reporting clone detection [5]. This scheme divides the network into subsets. A separate leader is present for every subset. Base station information is sent to the subset leader. A clone node is found by performing intersection operations on each root. The base station performs the intersection computation.

A social fingerprint is computed by extracting neighborhood characteristics [6]. A fingerprint is attached to every message, and it verifies them accordingly. Using S-disjunct code, fingerprints are generated with less communication and computation overhead. There are highly sensitive sensors in the base station that allows fingerprint verification. The Bloom Method is used to detect node replication attacks [7]. This scheme consists of predistribution, election, and detection phases. All nodes are assigned a node ID in the predistribution phase. Cluster heads are elected during the election phase. Cluster heads construct bloom filters in the detection phase; then another cluster head verifies them. In the CSI approach, nodes sense and transmit information to neighbors [8].

There is a fixed threshold value assigned. A clone node is assigned to a sensor reading greater than its threshold value. In token-based clone detection, the token is transferred with location ID and node ID to another node before transferring the message. If the neighboring node is cloned, it can misbehave with the token message after receiving the token. The node authentication is done before transferring the communication. The limitation of this approach is less detection accuracy. Distributed techniques do not have a central monitoring authority. Yet the witness node is selected at random to receive the information. All neighbor nodes send signed copies of location claims when they broadcast their location. However, the security is enhanced despite the high communication cost. RM introduces LSM to reduce communication costs. Using LSM, the overall number of communications costs is reduced, but the energy consumed increases. RM or LSM cannot detect masked replication attacks.

Another approach to the location-based key protocol is improving detection accuracy based on a bilinear map [9]. The sensor retrieves the location-based key by its current position. The key computation is based on Elliptical Curve Cryptography (ECC). The SDC Scheme maps each node to a single cell with its location. Occasionally, the node's storage cell serves as its witness [10]. The node identities are mapped to the destination cells using a hash function as part of the P-MPC scheme. There are multiple cells with different probabilities associated with each node ID. It should be considered in the detection protocol that witness node selection is random and widely distributed. A uniform witness distribution is achieved by selecting nodes pseudorandomly. The main disadvantage of RED is how witness nodes are selected. This scheme cannot detect a masked replication attack [11]. An order-based protocol is discussed to limit the deployments on nodes. The installed nodes should establish communication with their neighbors [12]. There is a possibility that the replicated node has access to the old, deployed key, therefore failing to establish the keys. An active detection approach randomly selects several nodes and assigns them as witness node. The communication cost is high, and detection accuracy is high [13].

The paper [9] recommends that past nodes be selected using a Random Walk (RAWL) scheme to be the witnesses. The first step of RAWL involves broadcasting a signed location claim from every node. Subsequently, selected neighboring nodes receive the claim. Third, random nodes are chosen, and information is sent to initiate a random walk through the network, and only those nodes that pass the walk will serve as witness nodes. Revoked replicas differ from their originals in the fourth step. TRAWL store's location claims within its table entries are significantly smaller than location claims. All passed nodes will remain witness nodes after they begin their random walk. The MRWS protocol will reduce the detection accuracy and detect replica nodes in large-scale sensor networks [14]. SSRWND protocol is the modification of RAWL and TRAWL. The communication cost and memory overhead are high for storing the node information in the Random Walk Table during a random walk in RAWL and TRAWL [15]. The SSRWND communication cost and detection accuracy are average. During the centralized approach of mobile WSNs, all mobile nodes are controlled by a monitor. The SPRT is used to detect mobile replication in a centralized approach. Low error rates are characteristic of this protocol. Since two nodes are at the same place simultaneously, replica nodes perform better than original nodes. The nodes checked for identical identities in the network are compared if the node pace is high.

An approach is discussed for detecting mobile replication using SPRT. Low error rates characterize this protocol. Because a replica node moves faster, the speed of the replica node is very high compared to the maximum node speed configured by the system. We check whether, in a network, identical identities have the same speed if the speed of the mobile node is high. This duplication of nodes will lead to nullity. By identifying replica nodes from the network, the hypothesis is accepted. In [16], the authors propose using both the time domain (TDD) and the space domain (SDD). All sensor nodes share a random number if they meet at any time. The same node is validated again if it runs into it by verifying the random number. A random number of bits is generated between two nodes in the range of each other. Each node's responsibility is to manage the table storing its ID and a random number. A comparison is made with the random number if it meets the node. If two random numbers are the same, each node will generate a new one. Replicas are also detected in cases where a node failed to meet earlier. The advantage of using XED is that replica nodes can be identified easily, but the drawback is that memory capacity is increased, and false replica detection results. There should be a limit on how many times a node can be visited in each interval if there is no replicating node.

Check the node ID and the time on a network with two replicas. It has two steps, offline and online: an EDD scheme. Before the sensor deployment, the network planner performs offline [17]. In this step, the parameters are calculated, including the duration to distinguish between the authentic node and the replica. Each node will perform the online step for each move. Every time interval, node replicas are encountered. In large-scale WSNs, efficient and distributed detection (EDD) cannot be applied. This scheme detects the UTLSE, which stores only a one-time location claim per replica of each witness node. Using the MTLSD protocol, nodes are assigned time-location claims. As opposed to protocol UTLSE, the MTLSD protocol has a greater detection chance [18].

Node replicator attacks in mobile wireless sensor networks can be detected using a distributed approach that does not require routing information. Node replication attack is a severe attack. The attack can be exceedingly injurious to many important functions of the sensor network, such as routing, resource allocation, and misbehavior detection. They exchange their time-location claims when they meet each other. Nodes will not transmit a witness's time location if they receive a neighbor. Even if a witness is not within the range, it will store the location claim. Instead of transmitting these claims, the witnesses carry them around the network. Data stored on local memory for every mobile node is verified with SDD-LC, exchanging information. In SDD-LWC, the information exchanged between nodes is common between their locally maintained tables. In SEDD [19], each node is monitored. Subnodes are monitored by each node at a set time interval, referred to as a monitor set. In addition to reducing storage overhead, the nodes monitored are monitored simultaneously. The article mentions another approach for exploiting mobile sensor networks to detect clone attacks. It has several advantages, including efficiency, avoiding synchronization, and revocation.

Another approach is to detect the replica node using the HIP/HOP method [20]. This method is the modification of the existing work [21]. In this scheme, the implied methodology divides the time into different rounds. The sensor nodes send their location claim to a neighbor node every round. The neighbor node will compare its history log for duplication. If the duplication occurs, then the location is verified. If there is a location conflict, the identified node will be assigned as a clone node [22]. The HIP verifies all the logs received and identifies the duplications. It has less storage requirement compared to the existing detection technique. In the HIP/HOP approach, the same technique will be challenging for global detection. Another approach is introduced to detect the replica node globally [23]. This approach is for detecting hybrid and global detection methods. The time slot is calculated as rounds. When the round is less, the detection accuracy is less [24, 25]. The existing system detection accuracy is very less, and the memory is very high. The proposed technique is the modification of SSRWND [15] and RAWL [9]. In the existing system, the network is divided into different regions. A node is selected for a random walk [19, 20]. The communication cost of selecting the node in each region is high in the existing system. Each node location claim, node ID, and signature are stored in a random walk. The drawback of RAWL and SSRWND is it occupies more memory to store the information of each node, such as node ID, signature, and location claim. If the duplication occurs in the random walk in the existing system, then the node-location ID is verified. If the location ID is different, the node is assigned as a replicated node. The major drawback in the existing system is that if the node ID and location ID are the same, it is assigned as a genuine node. If the genuine node is replicated, it is very difficult to identify in RAWL and SSRWND protocol. A SSS approach is proposed to overcome the issues in RAWL and SSRWND.

The Strategic Security System approach has three phases: prediction, detection, and isolation. If it predicts any node as a clone node in the prediction phase, it is sent to the next phase. The detection phase verifies the node ID, packet loss, distance, energy level, and key. If any discriminants occur, that node is sent to the isolation phase. In the isolation phase, the node is isolated from the network connection. Compared with the existing methods, this method's detection probability is high, with less communication overhead and less memory capacity. Although there are the possibilities of launching both active and passive attacks in a WSN, it is vital to first cater to the active attacks to reduce the negative consequences and disastrous aftermaths of an attacker. In such an attempt, a shortlist of the area of the active and serious attacks is presented in this section. The proposed system focuses on the detection of node replication attacks. This section explains the different techniques already proposed by different researchers. If the random walk is not visited in any region in RAWL, this protocol fails to detect the replicated node. To overcome the existing techniques' issues, the SSS technique is proposed. The proposed technique is the modification of the RAWL and SSRWND. The detection accuracy, memory overhead, and communication overhead are reduced in the proposed technique. This article is arranged as an Introduction, proposed system, SSS approach, Results and Discussion, simulation results, Conclusions, and References.

2. Materials and Methods

This proposed SS method detects the replica node in the distributed static topology. It influences the modified SSRWND algorithm to detect node replication attacks. In distributed static topology, the nodes are randomly placed in the network, and the base station is placed at one of the edges of the topology area. Each node has to report the various conditions in the region of interest to the base station from time to time. The initiation of the network operation begins with the nodes being configured as the WSNs and the registration of each node in the WSN with the BS. At one time run, the SS is performed during the network initiation, initiated by the BS. After this, any random node can become or probe another node to become an SS-Manager. This node can either be at the SS-Predict or the SS-Detect mode, depending on the security threat level sensed by the neighboring nodes. The proposed system performs three distinct operations for security preservation. The entire proposed SS method architecture is given in Figure 2. Assume the node$\{a, b, c, d, e\}$ in the network. All the nodes should update the node ID andD_{ij}in the table at T's random time. The node$\{a\}$is duplicated, and node$\{e\}$did not update its information in the table. The node $\{a, e\}$ is selected and transferred to SS-Detect. SS-Detect phase can be initiated by a node that has predicted the presence of the malicious node or by the transition from the SS-Predict. This is a critical mode in which the predicted nodes (NodePM) undergo vigorous checking to identify maliciously or just let go as legitimate nodes.

As observed in Figure 2, the SS method is probed to act as a predictor of a replication attack (SS-Predict), a detector of the replication attack (SS-Detect), and isolation of the replication attacker node (SS-Isolate) at one point of time during network operation. A group of communication nodes generally operates in the manner assumed below:

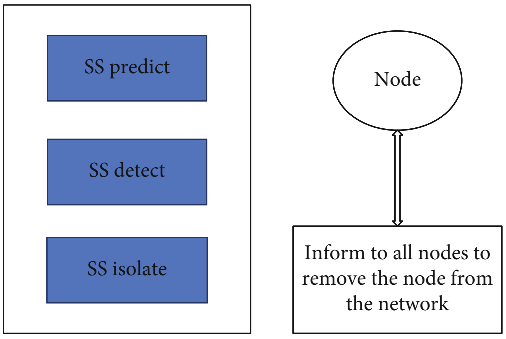


FIGURE 2: Architecture of SSS in a WSN.

A source sends a request to the destination to find the route to the destination.

A corresponding reply is obtained with the route to the destination.

The data are transmitted from the source to the destination along the route obtained.

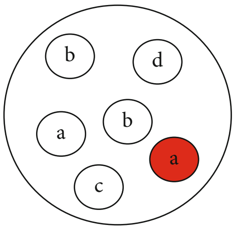
A final acknowledgment of the receipt of the entire data is sent back to the source.

During these communication processes, a replicator can easily act as an internal node (a node in the network since the commencement of the network operation) and misuse/modify/reinforce larger attacks. Generally, a small attack, when successful, can be the source of the possibility of launching a chain of large attacks in a system. There is a constant need to launch random checks in a WSN system. Taking this issue as an important motive, the SS-Predict for a system is proposed. All nodes in a network with higher resources (bandwidth, energy, and processing capability) are shortlisted when a trigger is created. A node is decided to be the SS-Manager for that session and can act either as a predictor and/or a detector and/or an isolator.

2.1. Network Model. The network is divided into different regions. A base station is present on any side of the network. The network is divided into three areas. The SS-Manager was selected based on energy level and not SS-Manager from the three areas. The SS-Manager performs a random walk at a random time. During a random walk, all nodes update their information in a table. The SS-Manager is more secure. The SS-Manager performs three operations. The three operations of the proposed system are discussed below.

2.2. SS-Predict. SS-Predict is one of the operations of the proposed system in which a node in the network picks up the presence of malicious activity. A node that identifies the malicious activity is called SS-Manager. SS-Predicts follows two cases:

2.2.1. Case 1. SS-Manager verifies the table if any duplicate node ID is present. And also, the node does not update the information in the Random Walk Table (RWT), which is also shortlisted and sent to the SS-detect. The duplicate node ID, speed, battery level, and key will be selected in the



(a)

NID	D_{ij}
A	a–>b
B	b–>c
C	c–>d
D	d–>a
A	a–>b

(b)

Figure 3: (a) Network structure. (b) Random Walk Table.

existing system and checked for the replica. In the SS method, if the node does not update the details in RWT during the random walk, that node also checks for the replica. At a random time, t, each node updates its information as node ID (NID) and the distance between the neighbor nodes (Dij).

Assume the nodes$\{a, b, c, d, e\}$ in the network are shown in Figure 3(a). All the nodes should update the node ID andD_{ij}in the table at$T$'s random time. The node$\{a\}$is duplicated, and node$\{e\}$did not update its information in the table shown in Figure 3(b). The node$\{a, e\}$ is selected and transferred to SS-Detect.

2.2.2. Case 2. The neighbor nodes of the current node are assessed for the abnormal variation in the packet delivery ratio (M_1), packet loss ratio (M_2), and the delay metrics (M_3). The metrics M_1, M_2, and M_3 are measured using equations (1), (2), and (3), respectively.

$$M_1 = \frac{\text{PR}}{\text{PR} + \text{PL}}, \tag{1}$$

$$M_2 = \frac{\text{PL}}{\text{PR} + \text{PL}}, \tag{2}$$

$$M_3 = \frac{\text{Delay}}{\text{Current simulation time}}. \tag{3}$$

Each node is checked for variation in its recent metrics. $M_{1\text{avg}}$, $M_{2\text{avg}}$, $M_{2\text{avg}}$, and M_{avg} are the individual and total averages. The conditions in equations (4), (5), and (6) are tested for the prediction of a malicious node (NodePM). If any of the conditions are true, then the SS-Manager automatically transits into the SS-Detect mode, the next mode of operation in SS.

$$M_1 < < M_{1\text{avg}} \text{ and } M_1 > > M_{1\text{avg}}, \tag{4}$$

$$M_2 < < M_{2\text{avg}} \text{ and } M_2 > > M_{2\text{avg}}, \tag{5}$$

$$M_3 < < M_{3\text{avg}} \text{ and } M_3 > > M_{3\text{avg}}. \tag{6}$$

2.3. SS-Detect. SS-Detect phase can be initiated by a node that has predicted the presence of the malicious node or by the transition from the SS-Predict. This is a critical mode in which the predicted nodes (NodePM) undergo vigorous checking to identify maliciously or just let go as legitimate nodes. This checking process will use cryptographic methods to identify and distinguish the malicious node. The detected nodes will be labeled as "NodeDM" at the end of the SS-Detect mode. If the above metrics are not satisfied, the detection phase is used. Each node stores the neighbor node information such as node ID and distance between the node at a different time (dj, tj) and (dk, tk). If $dj > = dk$ at time $tj < tk$, then

$$\frac{dk - dj}{tk - tj} \neq [v \min, v \max], \quad \text{if } tj < tk, dj < \text{d}kdj. \tag{7}$$

The deviation is measured based upon the binary deviation array. If the deviation is greater than or equal to 1, it is assigned as a replicated node.

$$D = \begin{cases} 1, & d_j > dk \text{ or } d_j < dk, \\ 0, & d_j = dk. \end{cases} \tag{8}$$

2.4. SS-Isolate. The SS-Managers can only reach the isolated phase. However, any random node that cannot enter SS-Isolate mode will be genuine. If a node enters the SS-Isolate, the node is assigned as a replica node. The detected node information is sent to the neighbor node of the replica node, depending on the location and distance. Each node stores the neighbor information while deploying the node. If the node receives the neighbor node as a replica node, then the node removes the replica node from the neighbor list. If any node sends data to another node, the node should send data to the nearest node. If the nearest node is in the neighbor list, the data will be passed through that path. The operational flow of how the modes can be achieved in SS is shown in Figure 4.

3. Results and Discussion

Let us assume $\{a, b, c, d, e, f, g, h, i\}$ are the nodes, and two replica nodes $\{a, e\}$ are present in the network. At the random time (t), each node updates its node ID and distance in the RWT table. Suppose during random walk the node ID $\{a, b, c, d, e, f, g, h\}$ has entered the information in the RWT table. The node ID $\{a, e\}$ has been duplicated in the table, and the node ID $\{i\}$ has not been entered in the table.

```
Input:   Input all the nodes in each region
Output:   Filter the node has the same node id, node not updated in RWT table, Condition not satisfied nodes.
Description:
1.        Initialize the nodes
2.        if (Time=t) then
           UT[i]⊠— (NID[i], Dij[i]); //update node id and distance should be updated
3.        For (j=1;j<=I;j++)
           If(NID[k]=NID[j]) or ( !list[j][table])
              Send the node to SS-Detect
4.        For(i=1 to node)
           m1[i]=pr[i]/(pr[i]+pl[i])
           m2[i]=pl[i]/(pr[i]+pl[i]
           m3[i]=delay/current time
           If(m1[i] < m1avg && m1[i]>m1avg)
           If(m2[i]<m2avg && m2[i]>m2avg)
           If(m3[i]<m3avg && m3[i]>m3avg)
                SSS-Detect(node[i])
```

Algorithm 1: SS-Predict.

```
Input:   Filtered Node from SS-Predict
Output:   Detect the exact Attacker Node
Description:
     1.  If(node.dj>node.dk) then
          Assign A as 1
          Else
          Assign A as 0
     2.  If (Vi>vj), then //check the battery level for the nodes
          Assign A as 1
          Else
          Assign A as 0
     3.  If(node.A==1)
          SS-Isolate(node)
```

Algorithm 2: SS-Detect.

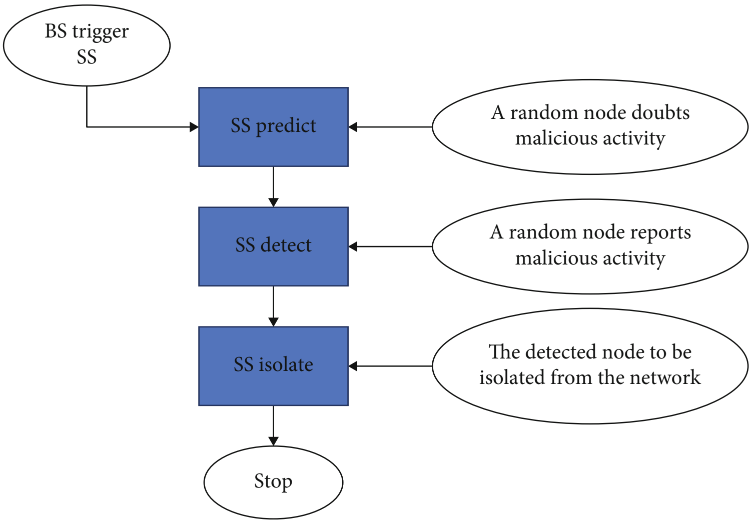

BS trigger SS
SS predict
A random node doubts malicious activity
SS detect
A random node reports malicious activity
SS isolate
The detected node to be isolated from the network
Stop

Figure 4: Working in SS modes.

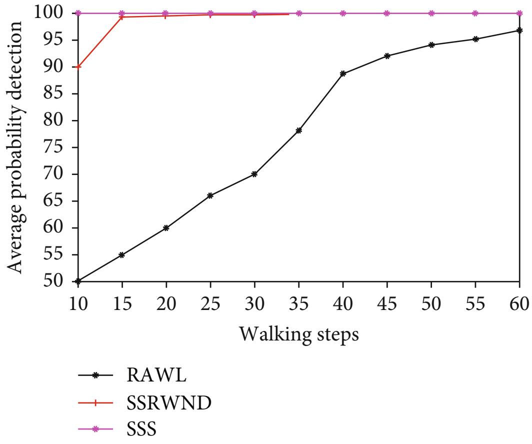


Figure 5: Detection ratio.

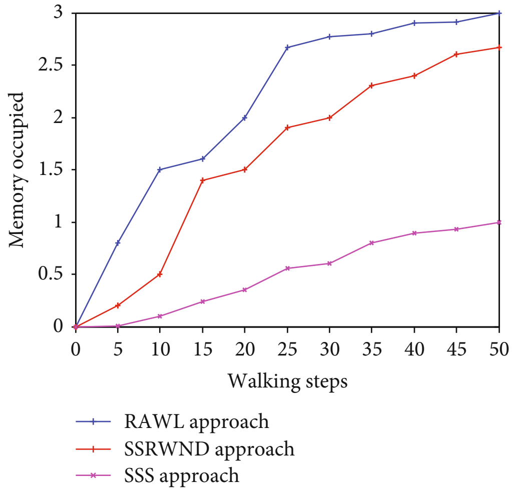


Figure 6: Memory overhead.

If there is any deviation in any metrics, the node is filtered. In predict phase, the node ID$\{a, e, i\}$ is filtered and transformed to the next phase. The next phase checks for the deviation in the distance and battery level. The replicated node is reported to all the nodes and isolated from the network. The probability of storing the distance information independently is $P(C2/(\sqrt{n}\log n))$ [RAWL]. Consider in $C1\sqrt{n}\log n$ step random walk; the distance claim stored is $P(C2/(\sqrt{n}\log n))$. If the distance claim is not updated in the table, then the probability is assigned as

$$P_{\text{no}} = \left(1 - \frac{C2}{\sqrt{n}\log n}\right) C1.\sqrt{n}\log n, \qquad \approx \frac{1}{e^{c1.c2}}. \tag{9}$$

$$P_{no} = 1 - \frac{1}{e^{c1.c2}}$$

Memory Cost per node is $O(C1.C2.\text{claim} + C1\sqrt{n}\log n$ .Entry). The table entry size in the SS technique is 2 bytes, 1 byte for node ID, and 1 byte for distance information. The memory cost of storing the information in random walk occupies 40 bytes in RAWL and SSRWND. It is 20 times more in RAWL and SSRWND.

Table 1: Detection probability.

	Walk step						
Methods	10	15	30	40	45	50	60
RAWL	50	60	70	88.7	91.9	94.0	96.7
SSRWND	90	99	99.7	100	100	100	100
SSS	100	100	100	100	100	100	100

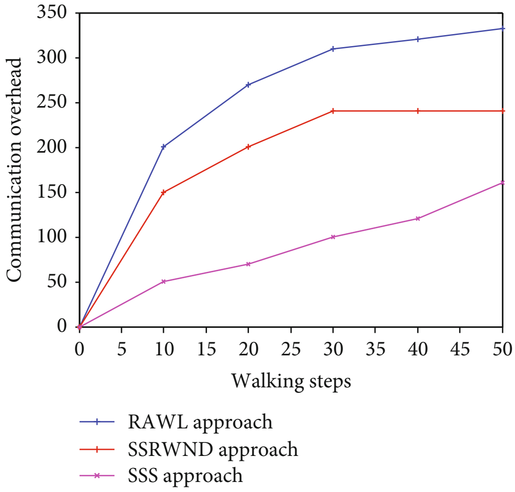


Figure 7: Communication overhead.

Table 2: Simulation parameters [26].

Parameter	Value
Channel type	Wireless channel
Simulation time	100 s
Number of nodes	50
MAC type	802.15.4
Traffic model	CBR
Simulation area	1100 × 700
Transmission range	250 m
Network interface type	WirelessPhy
Initial energy	10 J

3.1. Security Analysis. The detection ratio is represented in equation (10).

$$\text{DR} = \frac{\text{Number}_{\text{Correctly identified attacks}}}{\text{Number}_{\text{Total attacks}}}. \tag{10}$$

Figure 5 shows the detection ratio of the SS over existing methods RAWL and SSRWND. Figure 6 shows that the SSS detection ratio is higher than RAWL and SSRWND. It detects the misbehavior node and checks for the cloned node. In a random walk, each node updates its claim and node ID in the RWT. If any conflicts occur, then the SSS-

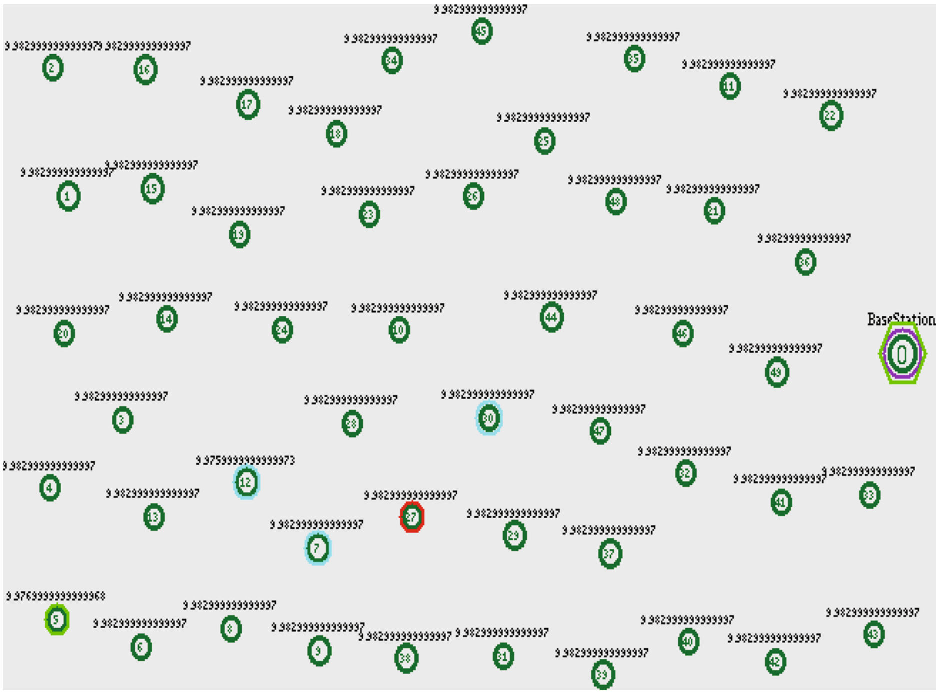


Figure 8: NAM output of the flat topology.

Detect to be invoked. Each node should be verified with distance and packet delivery ratio in SSS-Detect. The filtered nodes from the SSS-Detect are assigned as clone node. In RAWL and SSRWND, if the random walk step increases, the detection rate is high. In SSS-Detect, if the random walk step is very less, it detects the 100% replicated node. If the random walk $r = 5$, the detection probability is shown in Table 1.

3.2. Memory Overhead. In SSS, the memory occupied for storing the information is 2 bytes. The existing system takes 40 bytes to store the information during a random walk. In the SSRWND approach, if the random walk is 5, it has 3 regions, and the walk step is 11. Then, the memory overhead is 1.8 KB to 4.94 KB. In the RAWL approach, if the random walk is 5 and 51 steps, the memory overhead should be 2.92 KB to 11.46 KB. If the random walk is 5 and 3 regions in the proposed approach, then the walk step is 5. The memory overhead is 0.33 KB to 1.04 KB, as shown in Figure 6.

3.3. Communication Overhead. The random walk$r = 3$, and the walk step is 5. The random walk should be in two regions and calculated as 3/2. The walking step is calculated as $2 * 16 = 32(C)$ and random walk as $3 * 5 = 15(A)$ for two regions. The total communication cost is calculated as $C + A(32 + 15 = 47)$. The communication cost is calculated and shown in Figure 7.

3.4. Simulation Analysis. NS-2 is used to perform simulations, and the simulation metrics are shown in Table 2 and Figure 8.

3.5. Packet Received. The total number of packets was received in an SSS plot against the existing protocols in Figures 4 and 5, respectively. The SSS checks the node ID and distance. The SSS performs better than the RAWL and SSRWND, as observed in Figure 9.

3.6. Packet Lost. SSS performs better in both packet delivery and packet lost, as shown in Figure 10, compared to their existing baseline protocols.

3.7. Delay. We varied the network size from 50 to 250 systems to evaluate network scalability and test the effects. It is compared with the current system and the proposed algorithm. The delay of the proposed system compared to the existing one will be very low if the node is 50, which is shown in Table 3, and the graph is shown in Figure 11.

3.8. Energy Consumption. Energy consumption improves the energy required to pass data packets, as shown in Table 4. It wastes energy by sending many unnecessary routing of data packets (including control packet data) without significantly increasing the transmission rate of data packets, as shown in Figure 12.

$$\text{Energy consumption (EC)} = \text{Initial energy} - \text{Current energy (J)}. \tag{11}$$

3.9. Analysis of Packet Delivery Ratio. It is the sum of the number of data packets that the destinations successfully received to the number of data packets that the source produces. Depending on time, the packet delivery ratio is given, where the number of nodes is m and the interval of time is s. Figure 13 indicates the pattern of PDR at various times. It has been shown in Figure 13 that the rate of packet loss decreases over time. The analysis shows that the proposed

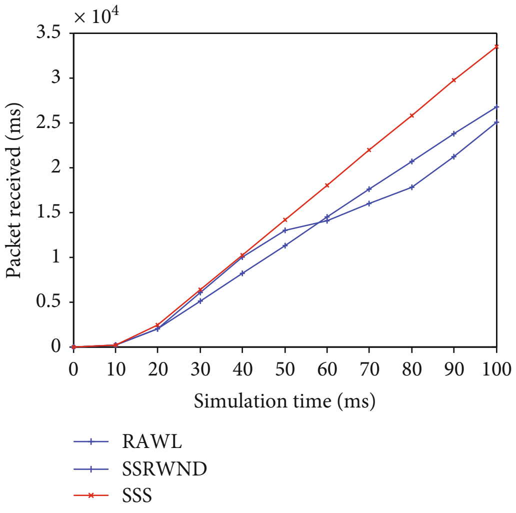


Figure 9: Packets received.

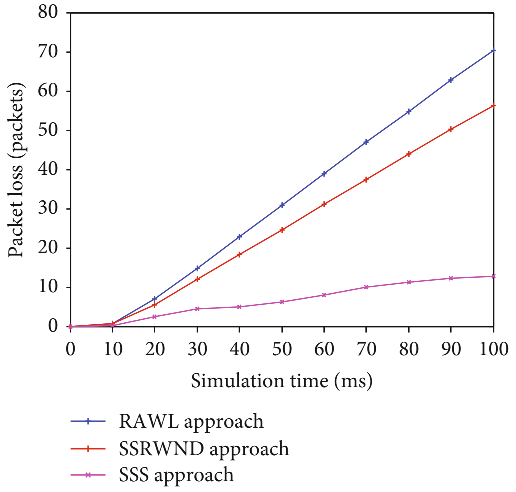


Figure 10: Packets Lost.

Table 3: Delay.

No. of nodes	50	100	150	200	250
Proposed algorithm	1	1.6	2	2.8	3
Existing algorithm	3	3.6	4	4.2	6

algorithm takes only a minimum number of nodes compared to the existing algorithm and less time to calculate it. With less time, PDR is increased, but more and more time is increasing. Figure 13 shows the analysis of the packet delivery ratio. For simulation, the number of nodes taken is 10, 20, 40, 60, 80, and 100. The packet delivery ratio is less compared with the existing system.

3.10. Analysis of Network Throughput. Network throughput is when packet orbits are successfully delivered over a network channel-

: throughput (bits/sec) = sum (number of successful packets) ∗ (average packet size))/total time sent in delivering that amount of data. The network throughput may be zero due to the jamming attack in certain cases. Let r be the transmission range and s be the time in secs. Mathematically, the network throughput can be expressed as the number of messages

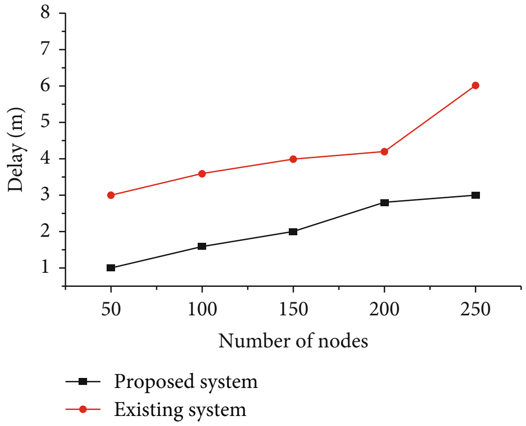


Figure 11: Delay

Table 4: Energy consumption.

No. of nodes	50	100	150	200	250
Proposed algorithm	380	400	600	920	1100
Existing algorithm	240	360	400	520	600

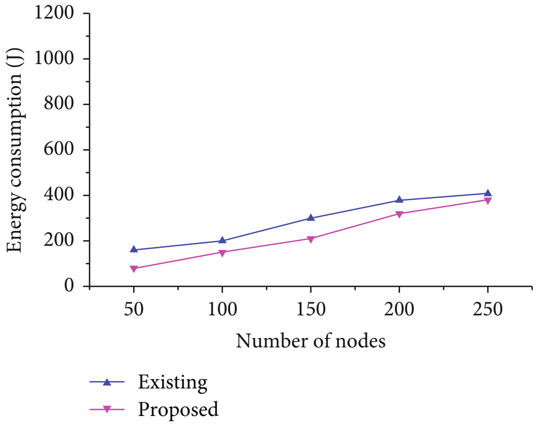


Figure 12: Energy consumption.

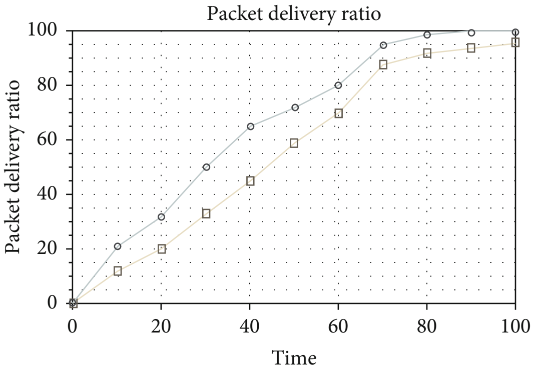


Figure 13: Analysis of packet delivery ratio

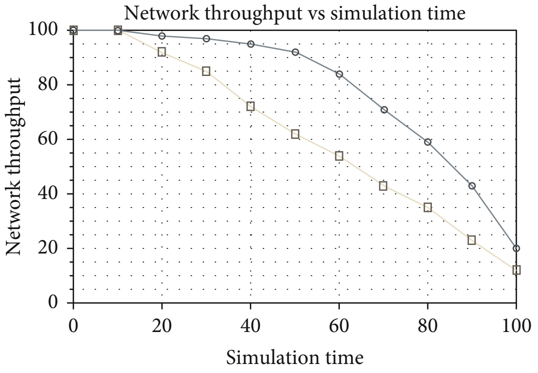


Figure 14: Analysis of network throughput.

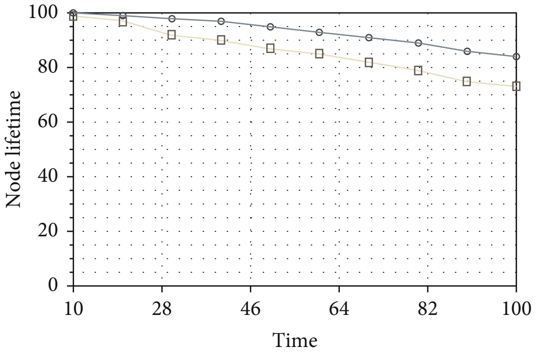


Figure 15: Analysis of node lifetime.

successfully transmitted per unit of time. And Figure 14 demonstrates the analysis of network throughput, which shows that the network throughput is high because the proposed algorithm does not use any extra information for dynamic updates compared to the existing technique. Figure 14 shows the analysis of network throughput. For simulation, the number of nodes taken is 10, 20, 40, 60, 80, and 100. The network throughput is less compared with the existing system.

3.11. Analysis of Node Lifetime. The lifetime of a node *Vx*, at time *t*, L*vxt*, is expressed as the ratio of the residual energy (Et) to the initial energy content of the node (Einit) and is expressed as a percentage value. In Figure 15, it is inferred that the node's lifetime in the proposed technique is longer than the lifetime of the node, which gives high stability among the nodes in the network and reduces overhead. The lifetime of the node should be stable. If the node is drained, it loses its energy; then, the node will be hidden and causes significant interference in the transmission.

Figure 15 shows the analysis of network throughput. For simulation the number of nodes taken is 10, 20, 40, 60, 80, and 100. The network throughput is less compared with the existing system. In the simulation, 100, 200, 500, 1500, and 2000 nodes are deployed in the network and verified. Initially, the number of malicious nodes is detected by verifying the Node-ID of the node that participates in the data transmission. To check the performance, the simulation is applied two times as the existing approach and the other one is with the proposed approach.

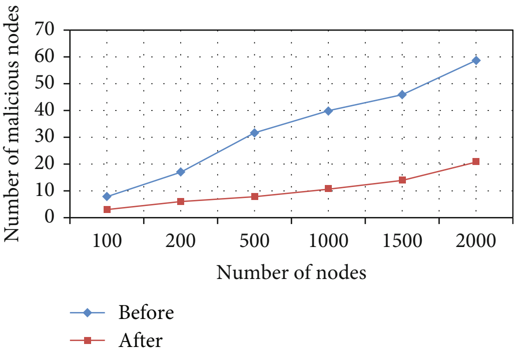


Figure 16: Controlled and uncontrolled malicious before and after applying the proposed system.

The result shown in Figure 16 illustrates the malicious behavior before and after applying the proposed system to the network. The number of malicious activities increases according to the number of nodes deployed in the network. From Figure 16, it is clear that severely malicious activity is found for more nodes, and for fewer nodes, less malicious activity is found. This controlled activity is deployed while verifying the node information itself. They are eliminated and do not reoccur during data transmission.

4. Conclusions

This paper presented a novel Strategic Security System (SSS), which performed three major operations such as prediction, detection, and isolation of replicators. The main advantages of this system are as follows: (i) there is no requirement for special nodes to act as an SS-Manager, and (ii) it consumes very petite additional resources from any security system. It is compatible with hierarchical topologies. From the simulation results, it can be concluded that this method serves to be one of the most efficient methods to eliminate the replicators from a network.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

All authors have equal contribution.

References

[1] W. He, X. Liu, H. Nguyen, K. Nahrstedt, and T. Abdelzaher, "PDA: privacy-preserving data aggregation in wireless sensor networks," in *Proceedings of the IEEE 26th IEEE International Conference on Computer Communications (INFOCOM'07)*, pp. 2045–2053, Anchorage, Alaska, USA, 2007.

[2] B. Parno, A. Perrig, and V. Gligor, “Distributed detection of node replication attacks in sensor networks,” in *2005 IEEE symposium on security and privacy (S&P'05)*, pp. 49–63, Oakland, CA, USA, 2005.

[3] W. T. Zhu, J. Zhou, R. H. Deng, and F. Bao, “Detecting node replication attacks in wireless sensor networks: a survey,” *Journal of Network andComputer Applications*, vol. 35, no. 3, pp. 1022–1034, 2012.

[4] Y. Li, M. T. Thai, and W. Wu, *Wireless Sensor Networks and Applications*, Springer, New York, NY, USA, 2008.

[5] E. C. Ngai, J. Liu, and M. R. Lyu, “On the intruder detection for sinkhole attack in wireless sensor networks,” in *IEEE international conference on communications, 2006. ICC'06*, vol. 8, pp. 3383–3389, Istanbul, Turkey, 2006.

[6] D. Gračanin, K. P. Adams, and M. Eltoweissy, “Data replication in collaborative sensor network systems,” in *2006 IEEE International Performance Computing and Communications Conference*, p. 8, Maryland, USA, 2006.

[7] A. P. R. Da Silva, M. H. Martins, B. P. Rocha, A. A. Loureiro, L. B. Ruiz, and H. C. Wong, “Decentralized intrusion detection in wireless sensor networks,” in *1st ACM international workshop on Quality of service & security in wireless and mobile networks*, pp. 16–23, ACM, 2005.

[8] S. Avancha, J. Undercoffer, A. Joshi, and J. Pinkston, “Security for wireless sensor networks,” in *Wireless Sensor Networks*, pp. 253–275, Springer, US, 2004.

[9] A. S. K. Pathan, Ed., *Security of Self-Organizing Networks: MANET, WSN, WMN, VANET*, CRC press, 2010.

[10] A. S. K. Pathan, H. W. Lee, and C. S. Hong, “Security in wireless sensor networks: issues and challenges,” in *Advanced communication technology, 2006. ICACT 2006*, vol. 2, p. 6, Phoenix Park, 2006.

[11] A. P. Renold, R. Poongothai, and R. Parthasarathy, “Performance analysis of LEACH with gray hole attack in wireless sensor networks,” in *International Conference on Computer Communication and Informatics (ICCCI), 2012*, pp. 1–4, Coimbatore, India, 2012.

[12] J. Wang, G. Yang, Y. Sun, and S. Chen, “Sybil attack detection based on RSSI for wireless sensor network,” in *2007 International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 2684–2687, Shanghai, China, 2007.

[13] T. G. Lupu, “Main types of attacks in wireless sensor networks,” in *WSEAS international conference. proceedings. recent advances in computer engineering*, Stevens Point, Wisconsin, USA, 2009.

[14] I. Krontiris, T. Giannetsos, and T. Dimitriou, “Launching a sinkhole attack in wireless sensor networks; the intruder side,” in *2008 IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*, pp. 526–531, Avignon, France, 2008.

[15] W. Z. Khan, M. Y. Aalsalem, and N. M. Saad, “Distributed clone detection in static wireless sensor networks: random walk with network division,” *PLoS One*, vol. 10, no. 5, article e0123069, 2015.

[16] M. Y. Aalsalem, W. Z. Khan, N. M. Saad, M. S. Hossain, M. Atiquzzaman, and M. K. Khan, “A new random walk for replica detection in WSNs,” *PLoS One*, vol. 11, no. 7, article e0158072, 2016.

[17] Y. Zeng, J. Cao, S. Zhang, S. Guo, and L. Xie, “Random-walk based approach to detect clone attacks in wireless sensor networks,” *IEEE Journal on selected areas in communications*, vol. 28, no. 5, pp. 677–691, 2010.

[18] S. Lalar, S. Bhushan, and N. A. Surender, “Clone detection using fuzzy logic in static wireless sensor network,” *International Journal of Vehicle Information and Communication Systems*, vol. 5, no. 3, pp. 334–353, 2020.

[19] M. Numan, F. Subhan, W. Z. Khan et al., “A systematic review on clone node detection in static wireless sensor networks,” *IEEE Access*, vol. 8, pp. 65450–65461, 2020.

[20] M. Jamshidi, S. S. Poor, N. N. Qader, M. Esnaashari, and M. R. Meybodi, “A lightweight algorithm against replica node attack in mobile wireless sensor networks using learning agents,” *IEIE Transactions on Smart Processing & Computing*, vol. 8, no. 1, pp. 58–70, 2019.

[21] M. Jamshidi, M. Esnaashari, A. M. Darwesh, and M. R. Meybodi, “Using time-location tags and watchdog nodes to defend against node replication attack in mobile wireless sensor networks,” *International Journal of Wireless Information Networks*, vol. 27, no. 1, pp. 102–115, 2020.

[22] S. Anitha, P. Jayanthi, and V. Chandrasekaran, “An intelligent based healthcare security monitoring schemes for detection of node replication attack in wireless sensor networks,” *Measurement*, vol. 167, p. 108272, 2021.

[23] L. Sujihelen and C. Jayakumar, “Inclusive elliptical curve cryptography (IECC) for wireless sensor network efficient operations,” *Wireless Personal Communications*, vol. 99, no. 2, pp. 893–914, 2018.

[24] L. Sujihelen and C. Senthilsingh, “Detect the replica node in mobile wireless sensor networks,” in *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, Madurai, India, 2021.

[25] L. Sujihelen, M. Satyanarayana, and C. Senthilsingh, “Replica node detection in distributed wireless sensor networks,” in *2021 5th International Conference on Trends in Electronics and Informatics (ICOEI)*, Tirunelveli, India, 2016.

[26] L. Sujihelen, C. Jayakumar, and C. Senthilsingh, “SEC approach for detecting node replication attacks in static wireless sensor networks,” *Journal of Electrical Engineering and Technology*, vol. 13, no. 6, pp. 2447–2455, 2018.