Hindawi
Mobile Information Systems
Volume 2023, Article ID 1029875, 12 pages
https://doi.org/10.1155/2023/1029875

Research Article

Detection and Counter Measure of Packet Misrouting by GPS-Free Positioning of Ad-Hoc Nodes

Godfrey Winster Sathianesan,[1] **S. Gnanavel**,[1] **R. Salini**,[2] **V. Raji**,[3] **R. Vijay Anand**,[4] **and Md Salah Uddin**[5]

[1]*Department of Computing Technologies, School of Computing, SRM Institute of Science and Technology, Kattankulathur 603203, Tamil Nadu, India*
[2]*Department of Computer Science and Engineering, Panimalar Engineering College, Chennai, Tamil Nadu, India*
[3]*Department of Computer Science and Engineering, S K P Engineering College, Tiruvannamalai 606611, Tamil Nadu, India*
[4]*School of Information Technology and Engineering, Vellore Institute of Technology, Vellore, Tamil Nadu, India*
[5]*Department of Multimedia and Creative Technology, Daffodil International University, Dhaka 1207, Bangladesh*

Correspondence should be addressed to Md Salah Uddin; salah.mct@diu.edu.bd

Received 21 December 2022; Revised 1 February 2023; Accepted 17 February 2023; Published 19 April 2023

Academic Editor: Robin Singh Bhadoria

The ad-hoc networks are a developing area of research with a large and wide variety of applications and related requirements. The nodes participating in an ad-hoc network use different routing protocols to send their packets from one node to another. But most of the time, the ad-hoc network is not suitable for urgent needs. For this, an ad-hoc network GPS-free positioning system can be used in emergency situations to save people in danger. Nodes participating in an ad-hoc network choose the best route from various nodes near them to send information through the complex system. For this, we have used trust dynamic source routing (TDSR) in our research work to determine and design the best route to transmit information, and we designed this system with the aim of being able to be used in emergencies. We have compared our proposed routing protocols with dynamic source routing (DSR) and found out that TDSR is working excellent.

1. Introduction

Ad-hoc networks are becoming more popular because they can be used in a variety of applications or fields and can be easily used to get the most out of them at a lower cost. Generally, they are used in unmanned areas to monitor information that is happening there, and if any help is needed, they are used in most rescue operations. Now, it is used for important tasks such as traffic monitoring, disaster monitoring, and remote sensing. The ad-hoc network is a set of portable node associations; it is used as a correspondence network, which involves self-communication and coordination of nodes.

Types of ad-hoc networks organise the following:

(i) Mobile ad-hoc network

(ii) Wireless sensor network

(iii) Hybrid network.

Nodes have a lot of memory and can store many different things. Nodes can automatically connect to each other and monitor each other. Specially appointed nodes are in reasonability of the routing, sending information, gathering information, security controlling of a network, perceiving malevolent nodes, and different variables tested in the network security issues [1].

Hence, it is not useful that the malevolent node stays in the network and may choose to change identities and try to enter the network as a newcomer. Malicious node isolation plays an important role in improving network security and performance. Referring to [2, 3], we can easily detect the malicious location by considering the concept of the

Bayesian model and the cooperation of a node that provides a node to its neighbor. After uploading the stored information to the RL technique, we use it to analyze it. Each gateway node chooses a way to deliver packets based on the reward of service in a meaningful way. To begin with, this paper proposes an ad-hoc network architecture, which is divided into two types, namely, the trusted mobile node and the trusted local portable node with RL technique.

A reliable network and a reliable mobile network follow both specified and unspecified responders at the disaster site. Corresponding to the nearest trusted mobile node, the trusted nodes within their boundaries describe the terrain of the disaster area. Many researchers have worked in recent years to overcome these challenges. They use various techniques related to artificial intelligence (AI) so that the network learns based RL and solves the challenge automatically. RL plays a very important part of the role in the development of AI. Apart from that, the RL methodology has played an important role in the research project and research process. It enables the user to carefully analyze the environment and make better predictions for the computer. However, to get a better forecast, the user needs to know the entire process, which is RL time-consuming and not good for large networks. RL development is an important requirement to learn new methods and functions for node design in a wireless ad-hoc network.

It is designed for emergency analysis and has important applications in the military and other fields. Due to the frequent changes in the nodes in the routing protocols, the tasks in them cannot be easily predicted, so the communication decision must be made quickly.

The other technique proposed in this field [4] depends on the Bayesian diversion hypothesis, which tries to conserve energy. To introduce such an attack, a malevolent node can stealthily drop any or all information or routing packets going through it. Because of the absence of physical assurance and a solid medium to get to a system, a packet dropping attack represents a genuine risk to the routing method in MANETs. In this way, it is evident that periods of correspondence, essentially route disclosure and information transmission stage, are ought to be ensured, calling for far-reaching security things.

1.1. GPS-Free Positioning. To recognize a malignant node in progressing information transmission and data passes with a source and progress of different nodes. A malevolent node misroutes the continuous information packets to disturb the correspondence. The neighbourhood trustier portable node finds an alternate way to convey the rest of the information packets in an alternate path. A nearby, trustier portable node limits the wrongly recognisable identification of honest-to-goodness nodes as malicious nodes. The nearby trustier portable node enhances the correct malicious node discovery.

Figure 1 shows that a local-thruster mobile node is identifying the position of neighbouring nodes using GPS-free positioning methods. Every participating node was able to identify the neighbouring node's position without GPS

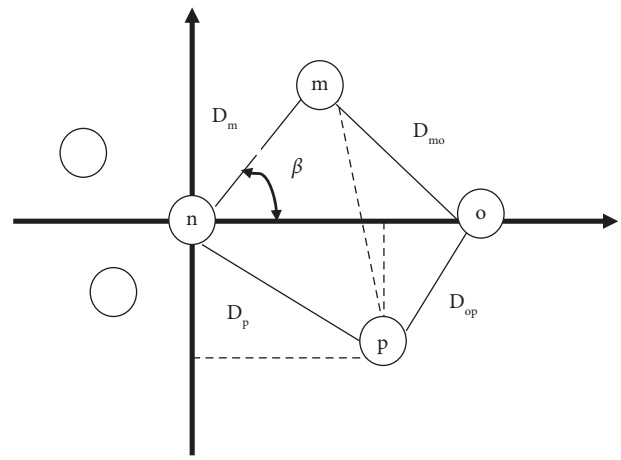

Figure 1: Local-trustier mobile node is identifying the position of neighboring nodes along GPS-free positioning methods.

[5]. The position of every node can be determined by its (x, y) coordinates, and its position values can be represented as (0, 0), where D_m, D_{mo}, D_{op}, D_p are distance between origin node to neighboring nodes. N, m, o, and p are participating nodes, and β is an angle of node n.

1.2. Motivation for Malicious Node Detection from Participating Nodes in the GPS-Free Positioning. To eliminate security problems from the GPS-free positioning network with RL, the information recorded by the participating nodes is determined by each individual's behaviour and analysis by RL, which is a significant problem when the packet is misdirected and the above-average packet loss from a node increases.

Since communicating with the same node every time is not guaranteed, it becomes challenging for each node to decide which node to trust among the nodes participating in the GPS-free positioning network. The neighbour node exchanges inaccurate data because of the different attackers, which affects the overall execution performance of the network. A few attacks, for example, the Sybil attack, the irregularity attack, and collusion protection are the principle situation in a network where a message can be modified via a third party, which can cause huge issues for the customers. A definitive objective of security is to offer protection administrations, for example, interruption discovery, authentication, getting the right of entry to control, and identifying the malicious nodes for powerful routing [6].

The role of trust value computation is to determine the level of trust or confidence in a certain entity or system, based on various factors such as past behaviour, reputation, and available data. This computation is used in various applications such as security, recommendation systems, and online transactions to determine the reliability and credibility of the entity in question.

We have used trust dynamic source routing (TDSR) in our research work to determine and design the best route to transmit information, and we designed this system with the aim of being able to be used in emergencies. We have compared our proposed routing protocols with dynamic

source routing (DSR) and found out that our system works excellent.

The rest of the paper is divided into the following sections: Section 2 is designed to make a thorough study of the work related to the paper and look at the related work. Section 3 provides discussions on it as well as the identification methods for detecting malicious nodes in a well-structured GPS-free positioning network. Section 4 is given in such a way that this information can be analysed and processed through mathematical analysis, and all of that information is compared with the information that precedes it. Section 5 simulates results and Section 6 gives the conclusion of the paper.

2. Related Work

Khalil and Bagchi stealth packet dropping is an attack that prevents an intermediate node from reaching a target through malicious behaviour. This type of attack is difficult to detect, and they have developed a protocol that can detect and isolate these attacks [7].

Taheri emphasised that secure routing is important for nodes to trust each other. Researchers have developed several trust-based routing algorithms to aid in this process. In the referred paper, we study different algorithms to optimise different ad-hoc routing protocols to improve trust between nodes in VANET [8].

Patel and Jhaveri identified one of the advantages of nodes in mobile ad-hoc networks as the fact that they work together to share information with their neighbors. By attacking other network nodes, they make it difficult for other nodes to detect them [9].

Kautoo et al. proposed a new protocol that uses a support vector machine to determine which routes are reliable and can improve the performance of STDSR by quickly finding routes with different mobility and numbers of nodes [10].

Bhorkar et al. try to find the best way for packets to reach their destination, taking into account the latency of network connections and the amount of traffic sent. This can be difficult because we must decide which path is the shortest and which will cause the slightest delay. This paper describes a way to do this using a distributed, opportunistic routing policy that considers congestion on network links. This policy tries to route packets to their destination as quickly as possible but also takes into account the level of congestion on network links [11].

Abderrahmane and Ali. In a wireless multihop network, packets are routed through intermediate nodes along the source-destination path. A dropper push is a type of attack where a network node drops packets to protect its resources [12]. The Merkel tree principle was used to justify this proposed approach. Through simulations, they have shown that approach is effective and evaluated it in both an ad-hoc network and a reactive routing protocol [13].

Kulkarni used RL techniques in ad-hoc nodes, which work smartly in GPS-free positioning. There are three different ways to think about estimating the value of an item. The first is called a reward predictor, which takes a state and predicts what rewards will be given to that state in the future. The second is called successor representations, which decompose the value function into two parts, namely, the reward predictor and the successor graph. The successor function shows how many nodes are currently in a state, and the reward predator maps the state to different rewards. A state's value function is computed as the inner product of the successor graph and the reward weights [14].

Li uses reinforcement learning to learn how to do something by getting feedback from people or things trying to do it. In this referred paper, we explore a different way to perform the compression abstraction task, using distributional semantics to measure the matching quantities [15].

Capkun et al. concluded that there is no guarantee that information transmitted between nodes in an ad-hoc network will always be accurate. Sometimes, it can be caused by malicious nodes. However, since the network is mainly used for emergencies, the information transmitted is accurate and truthful. This is done using various technologies so that the information cannot be read or changed by others. Nodes in an ad-hoc network are always changing, so there is no fixed location where all information is stored [5].

The details described in the referred paper clearly describe how to place the nodes in a high-density state, separate the roots and zones, and transfer the information between the two nodes. This paper aims to provide static information about the positioned node, and how to position and assign a positioning algorithm to a node is clearly described. Particularly, when ad-hoc network applications are developing in the fields regarding the industrial, military environment, rescue operation, and network protection issues are ought to be paid greater attention to study [16].

3. System Architecture and Method to Identify Malicious Nodes

Figure 2 shows that the proposed architecture helps to identify malicious nodes in the established path of the ad-hoc network.

There are differed ways to find the optimal edge of ad-hoc networks and identify the malicious nodes in their network. The first way to participate in the active node is to determine the credibility of the participant's node and to identify the attackers' node in the network that detected the performance of nodes of ad-hoc networks [1].

In a two way network, first way obtained from intensity of trust and it is reviewed, and the head of clusters does not perform any of work. The second step is that when it is sent to the next node, there are different issues of externality, bandwidth, acknowledgment, and approval. A malicious system allows a node participating in the cluster to make its decision on the wrong path. Therefore, a wrong decision is made by the node operating in random ways, and the source node causes the source to send its message in the wrong direction. Package deals are part of the business. The second method is the active node, which sends a signal from the attacker to detect a vulnerable hole and collects behavioural details of active nodes.

The cluster head node acts as a passive node of cluster, and the local-trustier node acts as an active node of the

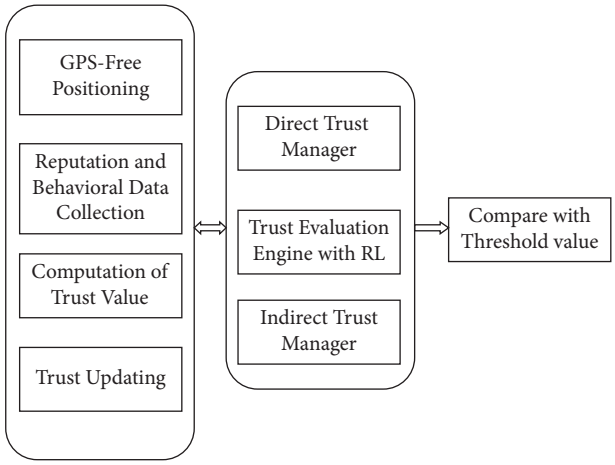

Figure 2: Framework of TLS.

cluster [17–19] Both exchange their information about the routing table through data packets. In addition, the cluster head node directly communicates with the other edge of the cluster, which links the cluster heads with the node of the local trustier.

(i) Local-trustier nodes collect additional information from intermediate nodes along the path to determine the expected time of the next packet from the source.

(ii) Temporary intermediate nodes act as routers that help forward packets on behalf of other nodes. The transit time of each packet is stored in the table of each intermediate node.

(iii) The trusted node calculates the communication time and the time of the packet from the source based on the travel time of previous packets and the GPS-free positioning of the central node.

(iv) Local trusted nodes manage route trust computation from recommendations of neighboring nodes along the route.

Each node has an access policy that includes access, read, modify, and configure permissions. Depending on access to legislation, enforcement can be divided into three levels, namely, low level, medium level, and higher level. Low-level members send applications for admission. A medium-level node can send and read; the higher-level node is authenticated. Currently, researchers are investigating temporal security issues, including security methods, encryption-awareness methods, cluster membership management, key distribution, detection and detection of intrusions, and denial of service (DoS) [13, 20]. The transmission limit of a node's ad-hoc network can be any condition. Information is sometimes received from nodes participating in the network, such as (i) good behaviour of the node can give good thoughts to others; (ii) negative behaviour of the node can give positive thoughts to others; (iii) good behaviour of the node can give negative thoughts to others; and (iv) negative behaviour of the node can give negative thoughts to others [21].

3.1. Node Selection Process in Ad-Hoc Network. The trusted node is selected based on different parameters as follows: (i) the trust value of a node, which is gathered from local neighbour nodes and neighbour networks' nodes; (ii) direct methods; (iii) indirect methods. When a new node joins an existing network, local-trustier calculates the weight score for all nodes, local-trust calculates the weight score for all nodes, then local-trust uses all node trust, sending and receiving Hello/ACK. It is transferred to different migration portals. Trust value is efficiently computed when network size is sparse. It allows only limited nodes in this network. The GPS-free positioning in an ad-hoc network does not utilize GPS. The mobile node positioning detail is given by GPS. Mobility nodes can be able to identify the other nodes' position when GPS cannot be used and a signal movement works on the nodes, the position information of the nodes can be identified by GPS-free positioning algorithms. The algorithm is referred as the "Self-Positioning Algorithm" (SPA). It uses range measurements between the nodes to build a network coordinate system. The time of arrival (TOA) method obtains the range between two mobile nodes; location-assisted routing and geodesic packet forwarding are also discussed in the referred paper. The algorithm checks whether the newly arrived node joins the network or not [5].

Algorithm1 is used when a new node joins the structure of the ad-hoc network, and then the parameters of the newly arrived nodes are checked by trusted local nodes. The line referring from 3–6 implies that a new node either accepts packet massage or deny pocket messages. Line 7 explains how a new ID is provided within the scope of the sparse network. In line 8, new nodes will make the decision on joining, either the GPS or GPS-free positioning. In line 9–19, the new nodes give their credible value to get the process, whether in a reliable score value range. Line 20–25 checks the new or existing node connection.

The following parameters are used to measure the trust evaluation, direct trust computed based on peer-to-peer, channel conditions, and interference. Indirect trust is computed based on the number of packets forwarded, overhead, misroute rate, and packet drop rate. There are two authority trust nodes to assign a new node, they are named as follows: (i) Local-thruster node (active participant). (ii) Cluster head (passive participant). In broadcast networks, when a node does not receive a signal from the mobile network, a GPS-free positioning algorithm is used to find the node's new location. The direct threshold value is computed based on peer-to-peer, channel conditions, and interference, as well as the indirect threshold value, which is computed based on the number of packets forwarded, overhead, misroute rate, and packet drop rate. In total, seven parameters are used to calculate the threshold value of the node based on the network condition and the node's motion.

3.2. Node Position Identification Process. GPS node positioning improves right detection concerning neighboring nodes. The node's performance has been discovered to contain malicious recreation based on the previous GPS positioning. If the previous node's position indicates a close

Procedure: requests_to_join_ new_node (ID_{chn}, $ID_{newnode}$, $TV_{newnode}$, ts)
Input:ts-denotes timestamps of starts process, $ID_{newnode}$-denotes ID of new node, ID_{chn}-denotes ID of cluster head node, $TV_{newnode}$-denotes trust value of new node.
Procedure:
(1) new node: Send JOIN REQUEST message to local-trustier node.
(2) Weight: local-trustier node starts a timer for computing the location of new node; the local-trustier node receives and saves packets during the timer.
(3) if the new node receives an ACCEPTED packet then.
(4) new_ node ← true.
(5) else if the new_node received a DENIED packet then.
(6) id ← 0, max_weight time ← 0;
(7) while id < max [node-id] do
(8) if GPS free positioning (vector value of new_node) then
(9) if new_node.weight > max_weight then
(10) max_weight ← new_node.weight
(11) if (local_truster node collects the routing table of newly join node and determine whether to join the network or not) then
(12) if (trust score > threshold value)
(13) append (new_node id)
(14) append (new_node MAC address)
(15) computing (to able forward packets to neighboring node/network based on energy level).
(16) computing (neighboring node hop count and collect intermediate node of newly join node)
(17) end if
(18) end if
(19) end if
(20) else if (check the new node is either new node or rejoin node) then
(21) new node access policy with RL
(22) new node permission (read, modify, forward, process based on three level (low, middle, high))
(23) new node permission (provides certificate authority based on the three levels with RL)
(24) new node trust score based past details
(25) end if
(26) i++
(27) end while

Algorithm 1: New node requests to join the network.

position to not deviate from the path between the specified coverage, it will now not be forwarded because it might be a malicious node. If the previous node's role denotes external routes beyond the source afterward, it can live away from the path. It improves the right selection respecting malicious nodes.

Algorithm 2 is used to identify new node that has joined, that is, the GPS-free positioning algorithm. Step 2 to 5 is input to the algorithm. Lines 6–8 check the condition of GPS position. Line 9 computes one hop. Likewise, in this algorithm, we find the GPS position for new node connected in the ad-hoc network.

3.3. Optimal Path Identification Processes. When a source sends a packet to a destination, it triggers Algorithm 3 on its network. The middle of the node in the path identifies the packet misrouting. The forwarding node is recognising the packet misrouting. Packet misdirection is detected by the next node in the direction. There are a few extra overheads in a network, including a node that is trusted by other nodes in the network. Both discovery and recuperation times are improved. GPS node positioning improves the correct detection of a malicious node. Node's mobility or malicious activity can be detected based on previous GPS positioning. If the previous node's position denotes a closer position in such a way as not to deviate from the path and the packet was not forwarded, then it might be a malicious node. If the previous node's position indicates that it is far from the source, it might be moved away from the path. It helps make the right decision about the malicious nodes. The performance can be increased by consuming low power consumption at the time of message communications between the nodes in the network. The misbehaving nodes are detected early in order to avoid traffic collisions, and this strategy helps to have low energy consumption. The traffic collision is decreasing, so the delay in message communication is gradually decreasing.

There are three predominant disadvantages to ad-hoc networks. They are bandwidth limitation, energetic and nonprescient topology, and the restricted processing and minimum staging spaces of ad-hoc nodes. The reliability of ad-hoc networks can be affected in various ways; the physical resources of nodes (such as mobile devices) can be changed or attacked by attackers. The attacker can be labelled using various criteria as shown as follows: (i) external attacks objectives in conformity with motive jamming or blockage, spreading fake routing information or disturbing the providing services by nodes; (ii) internal

(1) Function GPS_free_positioning ()
(2) Input:
(3) N: A set of nodes that participated in the ad-hoc network
(4) m, n, o: n is the neighbor's partner m
(5) ts: the time taken by packet was reached from one node to another
(6) if (local coordinate system) then
(7) n: position (0, 0)
(8) if (neighbor of n) then
(9) while (compute one hop neighbor)
(10) $N_{(m[i])}$ Communicate node $N_{(n[j])}$
(11) if ($N_{(m[i],\ n[j])}$ signal)
(12) D distance between ($N_{(m[i],\ n[j])}$)
(13) if (D = near)
(14) D_{dataset} ($N_{(m[i],\ n[j])}$, D, t)
(15) end while
(16) while (compute two hop neighbor)
(17) $N_{(o[k])}$ Communicate node $N_{(n[i],\ m[j])}$
(18) $\Theta = \cos$
(19) if (compute position of consecutive three nodes $N_{(n,\ m,\ o)}$)
(20) $m_{(x,\ y)} = (0, 0)$
(21) $n_{(x,\ y)} = (D_{i\ n}, 0)$
(22) $o_{(x,\ y)} = (D_{i\ n} \cos \gamma, D_{i\ m} \cos \gamma)$
(23) compute triangulation for distance discovers
(24) end if
(25) while end
(26) end if
(27) end if

Algorithm 2: GPS-free positioning algorithm.

attacks act as an ordinary node in the network which takes piece processes between the community nodes. At that place, there are numerous researchers contributing their interest in the area of trusted and secure routing. These are divided into two parts, which are the cryptographic method and the noncryptographic method. The cryptographic methods typically focus on decryption and encryption, symmetric keys, public keys, and data signatures in ad-hoc networks. With these methods, the procedure can guarantee integrity, classification, nondenial, accessibility, and confirmation of routing messages. Noncryptographic methods are used to protect the movement when other methods fail. There are various ways to solve the problems already identified, such as the watchdog mechanism [19], sprite, or credit-based system. To identify the node catch at the right time as expected. In imitation of spreading a small overhead.

4. Proposed Solution

Trust dynamic source routing (TDSR) is a routing protocol for ad-hoc wireless networks that uses trust values to determine the reliability of a node as a next hop in a route. TDSR calculates trust values based on the historical behaviour of nodes and their interactions with other nodes and uses these trust values to dynamically find and update routes. The use of trust values helps TDSR avoid routing through untrusted or malicious nodes, thus increasing the security and reliability of the network [22].

In TDSR, the local-trustier applies reinforcement learning (RL) to calculate the trust values of nodes. The local-trustier maintains a trust value for each node in the network and updates these values based on the nodes past behaviour.

RL is used to determine the trust value of a node by considering the rewards and punishments that the node receives based on its behaviour. The rewards are given for good behaviour (e.g., successful data transmissions), and punishments are given for bad behaviour (e.g., failure to transmit data). Over time, the trust value of a node reflects its history of rewards and punishments, allowing the local trustier to assess the reliability of a node as a next hop in a route.

The use of RL allows TDSR to dynamically adapt to changes in the network, as trust values can be updated in real-time based on the behaviour of nodes. This helps TDSR to avoid using unreliable or malicious nodes as next hops, thus improving the security and reliability of the network.

The data source for the training of reinforcement learning (RL) algorithms in trust dynamic source routing (TDSR) would typically be the data generated from simulations or real-world deployments of the TDSR network. These data could include information on the behaviour of nodes in the network, such as successful and unsuccessful data transmissions, as well as any rewards or punishments assigned to nodes based on their behaviour. These data are then used to train the RL algorithm to calculate trust values for each node in the network, which in turn is used by TDSR to make routing decisions. The quality and quantity of the

(1) Details of aggregate no. of packets to be transmitted will be sent at the time of path request.
(2) Traveling time from source to current node = (way for request packet/received time at current node) − (path for request/packet delivery time at source).
(3) Each node appends its GPS-free positioning (Vector Value) with every packet.
(4) Each node keeps up its previous two nodes GPS-free positioning.
(5) Expected time of next packet T (exp) = Max (Travelling time of path request/pervious packet) + Max (time term between two back to back packets) ± X
(6) X relies on upon GPS-free positioning of previous nodes.
(7) If no packet is gotten from pervious node after the normal time T (EXP), then the current node presumes that its previous node drops or misroutes the packet and sends a ready message to the nodes to have a place with the present path.
(8) Source node will give a substitute path to current node and sends remaining packets through new path up to current node.
(9) GPS-free positioning of nearby two nodes.

Algorithm 3: The source node will discover a path.

data used for training will play an important role in the accuracy and performance of the RL-based trust calculation mechanism in TDSR.

In the proposed work, we have tested several QOS parameters-based blueprints to observe or forestall flooding malicious assaults among ad-hoc.

(i) Throughput: Node performance is measured by counting how many packets (data) were successfully received by the node and then how long it took to send all the packets, which are given in the following equation:

$$\text{Throughput\_of\_a\_node} = \frac{(\text{Total\_ data\_bits\_Received})}{(\text{Total\_ data\_bits\_fowarded})}. \tag{1}$$

(ii) Packet loss rate: When sending packets, the greater the distance (hop count) between the sender and the destination, the more likely the packets will be lost. Packet loss occurs when some packets sent by a source do not reach their destination, which is given in the following equation:

$$\text{Packet\_Loss\_Rate} = \frac{\text{No\_of\_packets\_lost}}{\text{No\_of\_packets\_sent}}. \tag{2}$$

(iii) Packet delay: Depending on the network, the time it takes for a packet to reach its destination may be longer, which is given in the following equation:

$$\text{Packet\_Delay} = [\text{Receive\_Time\_at\_Destination}] - [\text{Transmit\_Time\_at\_Source}]. \tag{3}$$

(iv) End-to-end delay: The end-to-end delay measures how long it takes packets to travel from their source nodes to their destination nodes. Network traffic is the sum of the time packets take to travel from one node to another and the time it takes for each node to process them, which is given in the following equation:

$$\text{End} - \text{to} - \text{enddelay} = N[\text{Transdelay} + \text{Propogation\_ delay} + \text{Propogation\_ delay}]. \tag{4}$$

(v) Packet delivery ratio: A source's success rate is the number of packets it successfully received divided by the number of packets it sent, which is given in the following equation:

$$\text{PDR} = \frac{\sum \text{Number\_of\_sent\_Received\_packet}}{\sum \text{Number\_of\_sent\_packet}}. \tag{5}$$

(vi) Successful delivery rate (SDR): It describes the successful packets delivered to the packets transmitted, which are given in the following equation:

$$\text{SDR} = \frac{\text{No of packet delivered successfully}}{\text{No of packets transmitted.}} \tag{6}$$

4.1. Behavioural Data Collection. Behavioural data collection helps to record the behaviours of module nodes and the ways in which those behaviours datasets are connected. In the preferred paper, Algorithm 4 is proposed for behavioural data collection at the forwarding node. The trust value estimation of a node relies on how successful the information

was aimed in its lifetime. The total number of packets a node can send is limited by the total number of transmissions it has completed and not completed. When a node drops packets, it adds the packets it trusts to its "consider" list. This means the node will try to resend these packets.

4.2. Mathematical Analysis. In most of the proposed works discussed in Section 3, trust in the network is built by accepting and rejecting information from nodes. This work proposes an authentication and authorization model based on trusted nodes, which expands the role of gateway nodes in communication systems.

4.2.1. Calculation for a New Node Joins. Another new node trying to join the network sends its ID to the local trust node to be recognised as the principle of its network. A local trust node (LTN) estimates the nominal trust required by its node and its verification (TVth) (threshold value) in the neighbouring network. Acknowledgment provides a reliable guarantee, which signals to its neighbours an effective test. The transaction between n and m is part of a formal trust. Nodes participating in the intermediate path can be malicious nodes or normal nodes. Confirmation of the request and node should be available after a short time.

$$\text{LTN} = \text{Procedure: new\_node\_requests\_to\_join}\left(1D_{\text{chn,}}ID_{\text{newnode}}, \text{TV}_{\text{newnode}}, ts\right)\ldots \tag{7}$$

Equation (7) is used if a node that joins the network sees properties associated with nodes in the original network. new_node_requests_to_join—procedure for new node requests to join the network IDchn denotes ID of cluster head node, IDnewnode denotes ID of new node, TVnewnode denotes trust value of new node, and ts denotes timestamps of starts process.

4.2.2. Trust Model. Assessing the trust value of a node depends on how well information is protected during its lifetime. The total number of packets controls the number of completed and uncompleted broadcasts dropped by each node in the network. Add packets dropped by gateways that they think are trusted. A reliability control system is used to verify certain parts of the node. The given methods are determined by trust nodes like the direct trust model, indirect trust model, and local-trust node (LT) recommendation method. Equation numbers are known in the following order:

$$TVn, m = DTV\ n, m + IDV\ n, m + LTV\ n, m\ \ldots\ldots\ldots\ldots\ldots\ldots\ldots, \tag{8}$$

where TVn, m indicates believes node n on node m. DTV n, m is a direct reliability estimate of node n and its neighbors n, m. IDV n, m records the absolute trust index of node n and node m when it receives trust level m from a one-hop neighbor node. In this process, the trust value is used to destroy the public key, issue the public key, request the target node's public key, and then request the public key.

4.2.3. Access Permission. Direct and indirect trust values are produced beforehand and traded through current communication between the cluster head node (CHN) and the local-trustier node (LTN). Every node must demand a trust value (TV) from a neighboring, trusted node before entering the ad-hoc networks. Every node gets precisely one trust value, after safely confirming their identity with the local-trustier node. The node n gets authorization signal from local-trustier node as takes after which is given in the following equation:

$$\text{LTN} \longrightarrow n\text{: } TVn = [\text{IDA}, KPu, ts, et, P]\text{KEY}_{\text{Pr}}\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \tag{9}$$

We denote the local-trustier region by the symbol n with the ID of different nodes, and we assign a common code to the following messages to identify the message that the node sends: the timestamp reported by the symbol TS analysis, the time limit, and permission symbol P. These are generated by computing the values of trusted local-trusted nodes. All nodes must know the local-trusted node information. Purpose for carrying the information; nodes use this authentication information to identify themselves to different nodes.

4.2.4. Neighbor Node Authentication Process. Neighbour verification is a process of verifying whether the sender's message has reached the intended recipient. Node n sends data to the receiver, which receives the data from the neighbour node, which is given in the following equation:

Input: $n = 0$, N, A,CHN, LTN
N: Number of nodes participated in cluster.
A: Attacker node
CHN: Cluster head node
LTN: Local truster node
Output: resistant to attacker
(1) Regressive {
(2) if (the test attacker information is not available in routing table of local-truster node and cluster head node) then
(3) Release from this practice
(4) else
(5) Discover the attitude of attackers' node = {attackers$_n$}$\in$ N
(6) Approve action on attackers' node
(7) Discover action of attacker
(8) if (all attacker nodes follow the approve action) then
(9) The number of attacker' node increases or decreases.
(10) Update information into cluster head and local-truster node.
(11) end if
(12) n++}

Algorithm 4: Behavioural data Collection.

$$n \longrightarrow \text{Send: } [\text{PAC}K_{\text{id}}, \text{IDneigh}, \text{Pern}, \text{INn}, \text{ct}, P]\text{KEYPr} \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots \quad (10)$$

The PACK incorporates a packet identifier, the ID of the neighboring (ID_{neigh}), n's permission (Per_n), an immediate neighboring IN_n, the current time ct, and get to authorization (P), all marked with n's private key. To take into consideration effortlessness of immediate neighboring reusing, the prompt neighbor and timestamp are utilized as a part of simultaneousness with each other. It is made sufficiently huge with the node goal of abstaining from reusing in the conceivable clock skew between recipients. On the flip side, it is near the neighbors who have seen their timestamp with a specific node. On the off chance that the neighbor timestamp of the late period comes back to the correct parcel, the neighbors are stifled and thought to be acknowledged later.

Local-trustee detects tuples that do not send messages. Otherwise, the node continues to refer the message's content, adds its own specified state, and forwards the message to its next hop. Alteration of information or attack on truth is a countersignature. Let m be the neighbour that receives the PACKid of n sent as given in the following equation:

$$m \longrightarrow \text{Send: } [\text{PAC}K_{\text{id}}, \text{IDn}, \text{Pern}, \text{INn}, \text{ct}, \text{P}]\text{KEYPr}]\text{KEYpr}, P\,m \ldots.. \quad (11)$$

After receiving the PACK_{id}, m*'s* neighbor O shows the token and the given authentication token, then generates my own identity token, registers m as his ancestor, and shows the message he sent first. Add your own flag to the message. It returns the PACKid.

$$o \longrightarrow \text{Send: } [\text{PAC}K_{\text{id}}, \text{IDn}, \text{Pern}, \text{INn}, \text{ct}, \text{P}]\text{KEYpr}], \text{Po.} \ldots\ldots\ldots\ldots... \quad (12)$$

Each node repeats these steps to verify the previous node's signature, provide proof of the previous node's signature, and record the previous node's identity, where KEYPr-private key of node n; KEYPu-public key of node n; ts stands for timestamp; and show the time of the damage; IDn-terminal identification number; Per_n-erasing method with margin n which is given in equation (12).

4.2.5. Two-Nodes Authentication Process. Consider two nodes n and m. Each node will have time stamps PST (Packet Sending Time); PRT (Packet Receiving Time).

Condition 1:

If n is in LS, the accompanying two tests are conducted

Test 1: (For infringement of confidentiality)

Table 1: Simulation parameters.

Simulation parameters	Values
Routing protocol	TDSR
Simulation time	400 sec
Number of nodes	10, 50, 75, 100, 125
MAC protocol	IEEE 802.11
Transmission range	150 m
Number of malicious node	Random waypoint
Traffic model	UDP/CBR
Simulation area	1000 m × 1000 m
Packet size	512 byte

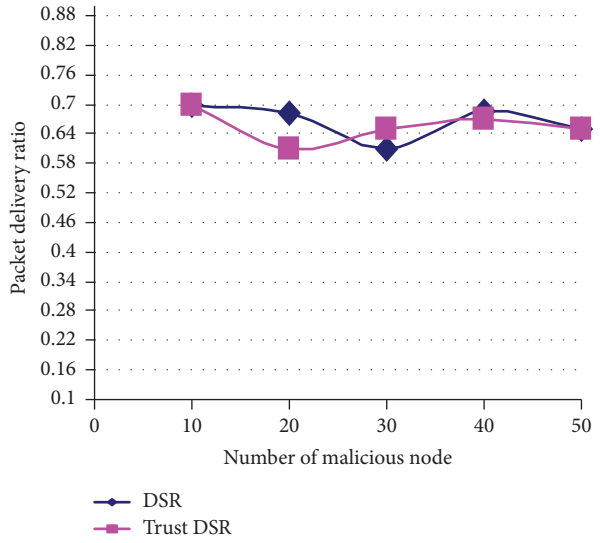

Figure 3: Packet delivery ratio vs. number of malicious node.

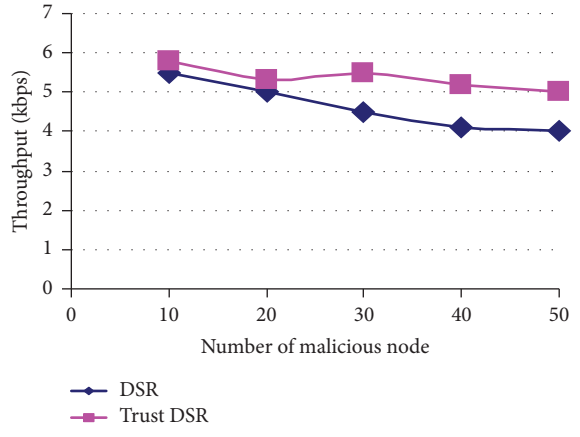

Figure 4: Throughput vs. number of malicious nodes.

If (PRT-PST) > T th (where *Tth* denotes threshold value)

Then TV = TV – 1

Test 2: (For infringement of trustworthinessIf (Permission is not coordinating)

Then TV = TV – 1

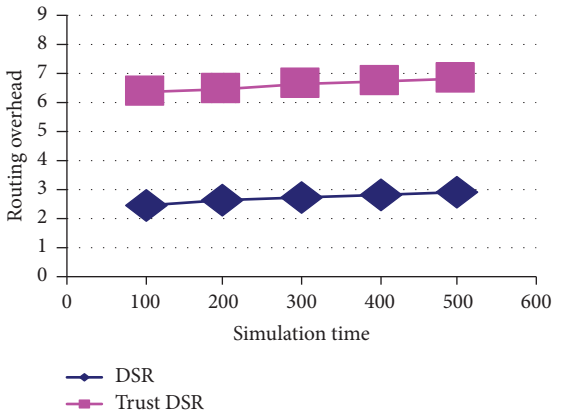

Figure 5: Routing overhead vs. simulation time.

Condition 2:

If *n* is in LTN, then

Code is executed *(Test-1).*

TVi is the confidence *coefficient of node Ni that each node evaluates in TEIk.*

All local nodes send Tci to their cluster head node CHN.

If the CHN detects **that Tci is less than TCth, it places it in the nearest CRL (Certificate Revocation List).**

Node ni sends its installation request to its CHN cluster leader.

CHN checks if it is in the CRL.

If he is found, then.

His claim will be rejected.

Otherwise, it sends *a signal reset response to ni and its signal.*

5. Simulation Environment and Performance of TDSR

The researchers' idea for this project is to implement trust dynamic source routing (TDSR) deployment strategy. The simulation was done in NS 2.34 (Network Simulator). The presented results are analysed using the Trace Graph 2.02 analyzer. Conventional simulations were attempted for small networks of 10, 50, 75, 100, and 125 nodes. The time period used for this analysis is shown in the following figures. Correlation and reporting are done using traditional DSR and TDSR methods.

Each node in the network is assumed to transmit according to the parameters shown in Table 1. Some results reveal that implementation of TDSR is a proposed compromise that favours implementation of traditional DSR. The simulation parameters performed for this research are shown in Table 1. The reliability factor of the node is calculated based on the number of packets lost by the node.

There are several useful metrics for selecting TDSR protocol performance. In this work, the parameters used to evaluate the performance of the protocol are network throughput, packet loss rate, packet delay, end-to-end delay,

packet delivery rate, and successful delivery rate. DSR and TDSR routing protocols are reviewed against the implementation of these provisions. It is clear from the figures that network performance improves when using node reliability. This method helps improve network performance when the amount of packet loss in the network is high. Change the implementation so that it does not affect the performance of the storage package building. A local trust node system supports agreement on lost packets by nodes, which reduces the number of packets. By using this strategy, the number of lost packets in the system is further reduced. This is strongly supported by the results presented in the following Figures.

Figure 3 obviously demonstrates the change in the network execution identified with packet delivery ratio and number of malicious node. TDSR packet delivery ratio improved compared with DSR.

This system will ensure that every packet on the network is trusted, therefore helping to create a secure network that ensures compliance, information security, and, all things considered, system performance. Figure 4 shows the number of malicious nodes and throughput. TDSR throughput high compare with DSR.

Routing overhead increases when communication-based trust policies and behavioural information collection are used to secure the network which is shown in Figure 5.

6. Conclusion

In this work, a monitoring system of connected probes in ad-hoc networks is studied to achieve secure transmission. A request for a new node to join the network algorithm uses the trust value, behavioural data, authority, and access to help determine node trust. A GPS-free positioning algorithm helps determine the location of nodes in a cluster. An optimal path-finding algorithm finds the best path between a source and a destination. A behavioural data collection algorithm is useful for trusted nodes in a set of nodes that collect historical information from nodes. The proposed algorithm TDSR helps collect data from nodes. Our research suggests improvements such as using a trust score when choosing a partner. This calculation improves the performance of the system and the delivery rate of the packets in the system.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] M. Z. Oo, M. Othman, and T. O'Farrell, "A proxy acknowledgement mechanism for TCP variants in mobile ad hoc networks," *Journal of Communications and Networks*, vol. 18, no. 2, pp. 238–245, April 2016.

[2] J. Vellaichamy, S. Basheer, P. S. M. Bai et al., "Wireless sensor networks based on multi-criteria clustering and optimal bio-inspired algorithm for energy-efficient routing," *Applied Sciences*, vol. 13, no. 5, p. 2801, 2023.

[3] K. Liu, J. Deng, P. K. Varshney, and K. Balakrishnan, "An acknowledgment-based approach for the detection of routing misbehaviour in MANETs," *IEEE Transactions on Mobile Computing*, vol. 6, no. 5, pp. 536–550, May 2007.

[4] R. K. Bar, J. K. Mandal, and M. M. Singh, "QoS of MANet through trust based AODV routing protocol by exclusion of black hole attack," *Procedia Technology*, vol. 10, pp. 530–537, 2013.

[5] S. Capkun, M. Hamdi, and J. -P. Hubaux, "GPS-free positioning in mobile ad-hoc networks," in *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, p. 10, Maui, HI, USA, January 2001.

[6] D. Muruganandam and J. Manickam, "Retracted article: an efficient technique for mitigating stealthy attacks using MNDA in MANET," *Neural Computing & Applications*, vol. 31, no. 1, pp. 15–22, 2019.

[7] I. Khalil and S. Bagchi, "Stealthy attacks in wireless ad hoc networks: detection and countermeasure," *IEEE Transactions on Mobile Computing*, vol. 10, no. 8, pp. 1096–1112, 2011.

[8] Y. Taheri, "Hossein GharaeeGarakani and naser mohammadzadeh "AGame theory approach for malicious node detection in," *MANETs" Journal of Information Science and Engineering*, vol. 32, pp. 559–573, 2016.

[9] N. J. Patel and R. H. Jhaveri, "Trust based approaches for secure routing in VANET: a survey," *Procedia Computer Science*, vol. 45, pp. 592–601, 2015.

[10] P. Kautoo, P. K. Shukla, and S. Silakari, "Trust formulization in dynamic source routing protocol using SVM," *International Journal of Information Technology and Computer Science*, vol. 6, pp. 43–50, 2014.

[11] A. Bhorkar, M. Naghshvar, and T. Javidi, "Opportunistic routing with congestion diversity in wireless ad hoc networks," *IEEE/ACM Transactions on Networking*, vol. 24, no. 2, pp. 1167–1180, 2016.

[12] B. Abderrahmane Baadache and B. Ali, "Fighting against packet dropping misbehaviour in multi-hop wireless ad hoc networks," *Elsevier Journal of Network and Computer Applications*, Elsevier, vol. 35, 2012

[13] W. Li and A. Joshi, "SMART: an SVM-based misbehaviour detection and trust management Framework for mobile ad hoc networks," in *Proceedings of the MIL· ITARY COMMUNICATIONS CONFERENCE*, pp. 1102–1107, IEEE, Baltimore Maryland, November 2011.

[14] T. D. Kulkarni, "Deep successor reinforcement learning," 2016, https://arxiv.org/abs/1606.02396.

[15] S. Li, "Deep reinforcement learning with distributional semantic rewards for abstractive summarization," 2019, https://arxiv.org/abs/1909.00141.

[16] L. A. Ali and N. Fisal, "GPS-free indoor location tracking in mobile ad hoc network (MANET) using RSSI," in *Proceedings of the 2004 RF and Microwave Conference*, pp. 251–255, Selangor, Malaysia, October 2004.

[17] T. Tsuda, Y. Komai, T. Hara, and S. Nishio, "Top-k query processing and malicious node identification based on node grouping in MANETs," *IEEE Access*, vol. 4, pp. 993–1007, 2016.

[18] F. B. Beshr, B. Ahmed, S. Aljabri, and R. Tarek, "Sheltami "A guard node (GN) based technique against misbehaving nodes in," *MANET" Journal of Ubiquitous Systems & Pervasive Networks*, vol. 7, no. 1, pp. 13–17, 2016.

[19] E. Hernández-Orallo, M. D. S. Olmos, J.-C. Cano, C. T. Calafate, and P. Manzoni, "CoCoWa: a collaborative contact-based watchdog for detecting selfish nodes," *IEEE Transactions on Mobile Computing*, vol. 14, no. 6, pp. 1162–1175, 1 June 2015.

[20] S. Gopalakrishnan and P. Mohan Kumar, "Performance analysis of malicious node detection and elimination using clustering approach on MANET," *Scientific Research Publishing*, vol. 7, pp. 748–758, 2016.

[21] M. Patel and S. Sharma, "Detection of malicious attack in MANET a behavioural approach," in *Proceedings of the 2013 3rd IEEE International Advance Computing Conference*, pp. 388–393, IACC), Ghaziabad, India, February 2013.

[22] M. Chen, H. Zhao, C. Shi, X. Chen, and D. Niu, "Multi-scene LoRa positioning algorithm based on Kalman filter and its implementation on NS3," *Ad Hoc Networks*, vol. 141, Article ID 103097, 2023.