International Journal of Information System Modeling and Design
Volume 13 • Issue 6

Detecting Sinkhole Attacks in IoT-Based Wireless Sensor Networks Using Distance From Base Station

Koushik Mondal, National Institute of Technology Meghalaya, India
Satyendra Singh Yadav, National Institute of Technology Meghalaya, India*
https://orcid.org/0000-0002-7891-6997
Vipin Pal, National Institute of Technology Meghalaya, India
Akhilendra Pratap Singh, National Institute of Technology Meghalaya, India
Yogita Yogita, National Institute of Technology Meghalaya, India
Mangal Singh, Symbiosis Institute of Technology, Symbiosis International University (Deemed), Pune, India

ABSTRACT

Wireless sensor networks (WSNs) are infrastructure-less in nature and contain a number of autonomous sensor nodes. These sensor nodes are dedicated to monitoring the physical conditions of the environment and are organizing the collected data at a central location. Application areas of WSNs, like heathcare and military surveillance, are sensitive. That's why security of WSNs needs to be very effective. Providing security to WSNs plays a major role as it consists of limited resources. The security system should lie within the boundary of the resource potential and be competent enough to handle attacks. Intrusion detection system (IDS) is one type of defense system that can fulfill the measure of limitation of resources. In this paper, a detection technique is proposed against sinkhole attack using the Euclidean distance of each node from the base station. The main advantage of the proposed technique is that it doesn't require any extra hardware setup and doesn't require extra communication cost.

KEYWORDS

Distance From Base Station, Internet of Things, Intrusion Detection, Sinkhole, Wireless Sensor Network

INTRODUCTION

Wireless sensor networks (WSNs) have gained popularity within research community because they provide a promising environment for numerous control and monitoring applications (Singh, Yadav, Kanungo, Pal, & others, 2021; Pinar, Zuhair, Hamad, Resit, Shiva, & Omar, 2016). These low-cost networks allow monitoring processes to be conducted remotely, in real-time and with minimal human intervention. The main feature of WSNs is infrastructure less nature (Pal, Singh, & Yadav, 2015). While addressing the network security of WSN, its infrastructure less nature makes it more vulnerable, and the limitation of resources makes it difficult to get a proper security mechanism. The security mechanism for WSN should be lightweight as well as robust enough to handle the attacks

DOI: 10.4018/IJISMD.297628 *Corresponding Author

faced by these networks. Intrusion detection system (Butun, Morgera, & Sankar, 2013) is a lightweight mechanism as well as can handle the attacks. The primary objective of an intrusion detection system is to convey the information when an attack on network intrusion might be taking place.

There are two types of attacks in WSN- active attacks and passive attacks. An adversary essentially affects the operations in the attacked network in active attacks. Jamming, hole attacks (blackhole, wormhole, sinkhole, etc.), Denial-of-Service (DoS), flooding are examples of active attacks (Wood & Stankovic, 2002). Attackers are typically secret (unseen) and moreover tap the message link to accumulate data or tear down the performance elements of the network in terms of passive attacks. Eavesdropping, node tampering, traffic analysis are examples of passive attacks (Grover, Laxmi, & Gaur, 2013) (Babaeer & Al-Ahmadi, 2020). IDS have been used to detect both types of attacks (Medeira, Grover, & Khorjiya, 2019) (Pundir, Wazid, Singh, Das, JPC Rodrigues, & Park, 2020). Sinkhole attack is an insider attack where an intruder compromises a node inside the network and launches an attack. Then the compromise node tries to attract all the traffic from neighbor nodes based on the routing metric that used in routing protocol. When the malicious node managed to achieve that, it will be able control those data packets passing via it.

The work of the paper presents a sinkhole attack detection mechanism which is primarily based on a parameter distance from base station (DBS). Proposed mechanism works in two phases. In the phase 1, it creates *Neighbor_Database* for each node. *Neighbor_Database* for each node contains its neighbor *node_IDs* and their corresponding DBS values. Whereas phase-2 detects the sinkhole node. At first it calculates a measure called *Difference (%)* value after that it compares the difference value with the threshold value. If the difference value is greater than the threshold value, it will be detected as a malicious node else it's a legitimate node. Here the constant threshold plays a crucial role at the detection phase. For increasing accuracy if threshold is taken very low then misdetection will occur. Moreover, *Detection_Rate* will be very low for very high threshold. That's why threshold is taken at an optimum value so that misdetection is almost zero as well as accuracy high. The mechanism proposed by Ibrahim et al. (Ibrahim, Rahman, & Roy, 2015) is unable to detect 1 hop distance sinkhole attack but the proposed method detects 60% - 80% near to base station (BS) attacks. The proposed method detects 100% sinkhole attacks when the DBS is greater than 35m. The proposed method has been simulated and analyzed for 3 different scenarios of network to endorse the scalability. For scenario-1 50 nodes and scenario-2 100 nodes and scenario-3 400 nodes are deployed.

The remainder of this paper is organized as follows: section II presents the related work. Section III describes the problem statement and the proposed detection method for sinkhole attack. The performance of the proposed algorithm has been evaluated and analyzed in section IV through simulations. Finally, section V concludes this work.

RELATED WORK

This section first discusses the types of attacks in WSN then represents the important contributions in the field of study.

Types of Attacks

Due to limited resources, WSN is susceptible in nature. Although there are lots of attacks against WSN, here some of the important attacks are discussed. These attacks are of two types - *Active attacks and Passive attacks* (Riaz, Buriro, & Mahboob, 2018).

Active Attacks

An active attack tries to change the system's resources or activities. Active attacks entail tampering with the data stream or fabricating misleading statements. These attacks are further discussed as TCP/IP layer wise as follows:

Physical Layer Attacks

At the physical layer, the attacks are hardware-based, and while they are simple to carry out, they require access to hardware to be successful. These assaults are fairly straightforward and do not necessitate a thorough understanding of the technology (Padmavathi, Shanmugapriya, & others, 2009).

Physical Layer Jamming Attack

Attackers can initiate a jamming attack both outside and inwardly. They can launch a jamming attack by utilising a high-power transmitter, creating a signal(s) powerful enough to interfere with lawful communication with wireless medium. As a result, a true source may be prevented from broadcasting packets, or legal packets may be denied as received (Jawandhiya, Ghonge, Ali, Deshpande, & others, 2010). The most prevalent kind of signal jamming is random noise and pulse. Because of the dynamic network structure and frequent location changes of sensor nodes, the adversary finds it difficult to mount a jamming assault.

Passive Information Gathering

An intruder may be able to acquire important information from a wireless sensor network if the information is not encrypted using sophisticated algorithms. With a well-designed antenna and a strong receiver, an attacker may simply get into the data stream. This allows them to intercept messages, including the actual position of sensor nodes, allowing invaders to locate and destroy those nodes. Aside from the sensor node's position, the opponent may look at the application-specific message contents, which include message IDs, time stamps, and other fields of interest (Pathan, Lee, & Hong, 2006).

Device Tempering Attack

Unlike nodes in a wired network, WSN nodes are often soft and compact in nature. A receiver node that is in the route discovery process should validate and sign control messages originated by a node. As a result, route discovery prevents anti-authenticating attacks, such as routing loops, because no node may construct and label a data packet sent in the name of a fictitious or falsified node (Walters, Liang, Shi, & Chaudhary, 2007).

Message Corruption

Whenever an adversary do some changes in the message, it may do some malicious changes which leads to message corruption attack. In this attack mainly data integrity is hampered (Healy, Newe, & Lewis, 2009).

Medium Access Control Layer Attacks

Link Layer Jamming Attacks

The goal of these attacks is to interrupt regular operations among sensor nodes around the jammer. This attack takes use of the vulnerabilities in a few link layer protocols (Law, Hartel, den Hartog, & Havinga, 2005).

Jelly Fish Attacks

In the event of a Jelly Fish attack (Aad, Hubaux, & Knightly, 2004), the malicious node follows the protocol but quietly disrupts, delays, or loses packets on a regular basis. Such an attack is difficult to detect since the node works normally for a long period, and therefore no monitoring tool can negate the trust level of such nodes.

Intelligent Cheater Attacks

These violent attacks are similar to jelly fish attacks, in which the nodes behave normally most of the time but occasionally act maliciously. Because such clever and smart nodes retain their trust rating within a specific threshold level, the damaging potential of such attacks is difficult to identify (Nagel, Shokranian, Bordim, & Nakano, 2008).

Collisions

In explicit packets like as acknowledgement (ACK) messages, an enemy or adversary node may deliberately cause collisions. In a few medium access control methods, this leads in pricey exponential back-off.

Network Layer Attacks

The goal behind a network layer attack is to inject control from the node itself into the sender and receiver paths, causing network data to be diverted. By assaulting any routing protocol, the invader can achieve these goals and objectives. In the next paragraphs, several network layer attacks will be addressed.

IP Spoofing Attack

In the event of conflict detection allocation, a new node chooses a random address (call X) and sends a packet to the sensor network (conflict-detection). Any denial by a node will prevent it from using that address. IP Spoofing occurs when the victim node constantly imitates an associate who has obtained the same IP address and responds with "rejections" (Walters, Liang, Shi, & Chaudhary, 2007).

Link Spoofing Attack

A hostile node propagates false links with other nodes excluding surrounding nodes to disrupt routing processes in this kind of attack.

Flooding Attack

The goal of this attack is to drain network resources such as processing and battery power, bandwidth, and therefore use the node's resources, or to disrupt a routing function to cause stringent network starvation. As a result, bandwidth of network and battery power of nodes will be used, potentially resulting in Denial of Service (DoS) attack (Nayak, Suseela, & Trivedi, 2017).

Blackhole Attack

There are two main aspects to this attack. First, to announce, the node utilises routing protocols such as Ad-hoc On-Demand Distance Vector (AODV), even if the path is altered, it has a route to the destination with the purpose of capturing packets. The intruder then exploits the intercepted packets for its own purposes rather than transmitting them (Nagel, Shokranian, Bordim, & Nakano, 2008).

Grayhole Attack

The Grayhole attack is nothing but a variation on the blackhole attack. Rather than dropping all the packets like blackhole attack, this attack just drops some selective packets.

Wormhole Attack

This attack is regarded as one of the most dangerous and complex in the WSN. An invader maintains track of packets at one point in the network and creates a tunnel to another sensor node in the network, through which the data packets transit. Wormholes (Hu, Perrig, & Johnson, 2006) are the tunnelling between these two collaborating assault pathways.

Resource Consumption Attack

Sleep deprivation attack is another name for a resource consumption attack. By forwarding packets or asking excessive routing information of the target node, the attacker might try to make use of the node's battery.

Sinkhole Attack

The main goal of the attacker in this attack is to draw attention to all traffic from a specified location by exploiting a compromised node. Sinkhole attack is the term for this traffic-gathering method (Karlof & Wagner, 2003) (Sejaphala & Velempini, 2020).

Sybil Attack

In this type of attack, a hostile sensing node displays different IDs to other network nodes. This threat is aimed against fault-tolerant solutions like multi-path routing and distributed storage.

Hello Flood Attack

Several routing systems employ the "HELLO" message to locate nearby nodes and therefore build the network topology. An attacker's easiest approach is to transmit a flood of such messages, flooding the network and preventing other messages from being transmitted. This attack is called Hello Flood Attack (Singh, Jain, & Singhai, 2010).

Selective Forwarding Attack

In a multi hop network, it is critical that all sensing nodes reliably and authentically send the received message(s) to the next node. Meanwhile, a few infected nodes may refuse to forward packets; alternatively, neighbours may begin utilising other nodes.

Routing Table Poisoning Attack

Routing Table Poisoning Attack occurs when a corrupted node in the network modifies the actual route or sends fictional routing updates to other unaffected nodes. This poisoning of the routing tables causes high traffic in particular sections or all over the entire network, as well as sub-optimal routing.

Transport Layer Attacks

SYN Flooding Attack

An invader establishes a large number of half-opened TCP connections with a victimised node, but is unable to finish the handshake required to completely establish that connection. It leads to Denial of Service attack.

Session Hijacking Attack

In this form of attack, the attacker imitates the target node's IP address, discovers an exact sequence number which is expected by the target node, and then launches a Denial of Service attack on the victimised node.

Application Layer Attacks

Malicious Code Attack

Worms, viruses, Trojan Horses, and spyware are examples of malicious programmes that may target both user applications and the operating system. These malicious applications typically cause computer systems and networks to malfunction or slow down (Walfish, Vutukuru, Balakrishnan, Karger, & Shenker, 2010).

Repudiation Attack

A rejection of partial or entire participation in communication is referred to as Repudiation Attack. For example, a selfish individual may refuse to execute credit card purchase operations, which is considered a repudiation assault in the commercial system.

False Data Filtering Attack

In-network data aggregation is commonly used in energy-constrained WSN. Because of the needs of data aggregation, end-to-end encryption becomes impractical. An assault on a point of aggregation allowing an intruder to change or harm the total quantity of data flowing from the down-stream nodes, as well as the total data aggregation result encountered at the base station, putting sensing applications at risk.

Clock Synchronization Attack

In wireless sensor networks, time synchronisation is a critical component. Sleeping patterns can be disrupted by time desynchronization. During the time exchange, an invading node might send a false synchronisation message to its surrounding nodes, causing other nodes to calculate an incorrect skew and phase offset.

Passive Attacks

Malicious nodes use passive attacks to get insight into the nature of behaviors and obtain data from the network without disrupting the operation (Riaz, Buriro, & Mahboob, 2018). The passive attacks are discussed below:

Eavesdropping

By spying on data, an intruder can quickly discover communication. For example, if traffic conveys control the flow of information for a wireless sensor network, an eavesdropper can possibly obtain more information than a server can provide.

Traffic Analysis

Analysis of the communication patterns of WSN is still feasible even when encrypted data are being sent. Sensor communication operations might inadvertently provide enough information to allow an adversary to destroy the sensor network.

Camouflaged Adversaries

In WSN, an intruder can capture or conceal a targeted number of nodes. Following that, this type of nodes can impersonate a normal node to pull packets, then misroute them, and finally do a privacy analysis.

Packet-tracing

In a packet tracing attack, an equipped adversary can inform the location of the immediate sender of an observed packet. The attacker is capable of doing hop-to-hop tracing towards the direction of the source of original data, which exposes the source's privacy.

Ngai et al. (Ngai, Liu, & Lyu, 2006) proposed a detection technique to detect sinkhole attack. In this mechanism the BS first floods with a request message with the IDs of the affected nodes. In reply the affected nodes send their IDs, ID of the next hop and the associated cost. Based on this information the base station to constructs a network flow graph for identifying the sinkhole. For this entire process the communication cost is highly increased as well as the security of this communication is another important aspect. Although authors have proposed encryption and path redundancy to avoid the security issue of communication. Based on Link Quality Indicator (LQI), Choi et al. (Choi,

Cho, Kim, Hong, & Kim, 2009) proposed a detection technique for sinkhole attack for WSN. It can detect sinkhole attack for the networks with LQI based routing and the nodes are divided into detector nodes and general nodes. The detector nodes detect sinkhole with minimum path cost measure. On Salehi et al. (Salehi, Razzaque, Naraei, & Farrokhtala, 2013), first some suspicious nodes are grouped based on their consistency of data. After that it checks the network flow information to detect the exact sinkhole node.

Another sinkhole detection mechanism is proposed by Ibrahim et al. (Ibrahim, Rahman, & Roy, 2015) with hop count parameter. The proposed mechanism calculated a difference value with neighbor nodes' hop count and compared it with a threshold value to check sinkhole node. Some other mechanisms also proposed to detect sinkhole-based routing protocols like the AODV (Dallas, Leckie, & Ramamohanarao, 2007) and the DSR (Pirzada & McDonald, 2005) Protocol. R. Maheswari et al. (Maheshwari, Gao, & Das, 2007) proposed a wormhole attack detection approach based on connectivity information. It compared the connectivity graph of WSN with forbidden structures, based on that it made decision about the presence of wormhole attack. This approach is applicable for both known and unknown communication models of WSN. A hybrid technique of Watchdog and Delphi to detect wormhole attack is proposed by R. Singh et al. (Singh, Singh, & Singh, 2016). By the Watchdog technique it keeps tracking the packet drop information and Delphi takes care about the round-trip time (RTT) of a route. If the RTT and the packet drop count both are low, then the route is suspicious. Actually, it is using the advantages of both Watchdog and Delphi to eliminate their disadvantages. Table 1 summarizes the above discussed technique for finding the malicious nodes.

Table 1. Summary of Related Work

Author	Aim	Work Model
Ngai et al. (Ngai, Liu, & Lyu, 2006)	Sinkhole Attack Detection	By flooding technique, base station gets information about affected nodes. Next, it detects malicious node by creating network flow graph based on the previous information.
Choi et al. (Choi, Cho, Kim, Hong, & Kim, 2009)	Sinkhole Attack Detection	Divides sensor nodes into detector nodes and general nodes. Detector nodes detects attack based on Link Quality Indicator (LQI) technique.
Salehi et al. (Salehi, Razzaque, Naraei, & Farrokhtala, 2013)	Sinkhole Attack Detection	After grouping suspicious nodes, detects malicious node by checking the network flow information.
Ibrahim et al. (Ibrahim, Rahman, & Roy, 2015)	Sinkhole Attack Detection	Based on neighbor nodes' hop count information, it creates difference parameter to decide a node is malicious or not.
R. Maheswari et al. (Maheshwari, Gao, & Das, 2007)	Wormhole Attack Detection	Compares the connectivity graph of WSN with for-bidden structures to detect the attack.
R. Singh et al. (Singh, Singh, & Singh, 2016)	Wormhole Attack Detection	A hybrid technique of Watchdog and Delphi. Watchdog keeps packet drop information and Delphi keeps RTT information. Combining this two information, detects malicious node

Detection of sinkhole attack on a node near to base station is still wide open as the hope techniques are not able to detect the attack at 1-hope. Work of this paper addresses the issue of sinkhole attack in WSN.

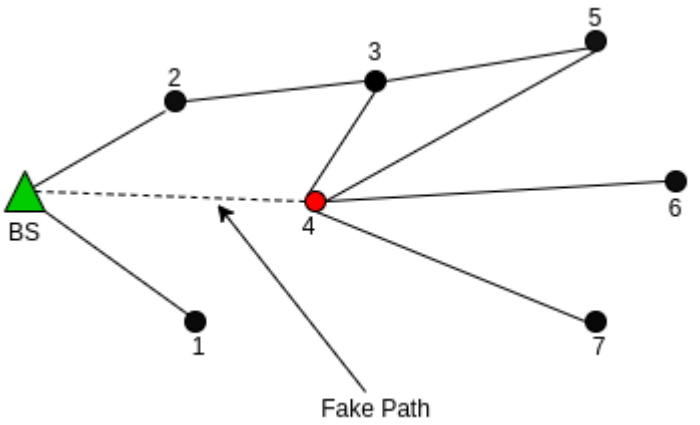
OBJECTIVES AND METHODOLOGY OF THE PROPOSED WORK

Objectives

The sinkhole detection technique of Ibrahim et al. (Ibrahim, Rahman, & Roy, 2015) failed to detect the sinkhole attack near to BS. As they are using hop count as their main detection parameter that's why it's impossible to detect sinkhole attack on 1 hop distance node. A network of 7 nodes and a BS has been depicted in Fig. 1. Sensor nodes are deployed over an open environment and two nodes are neighbor if their Euclidean distance is less than or equal to radio distance (R), where '*R*' is a constant. An attacker can capture any legitimate node and can decrease the DBS value of it to make a fake path to BS. Now the neighbor nodes may choose the path to BS via this malicious node. As shown in Fig. 1 node 4 has a high-quality path (fake path) towards BS with very less distance. As a result, node 3, 5, 6, 7 will chose the path via node 4 to send their data packets to BS. Now the attacker can change the data inside the packets, or it can drop the whole packet.

The scheme presented in this paper solves the 1 hop sinkhole detection and increases the accuracy to detect sinkhole attack throughout the network.

Figure 1. Sinkhole attack demonstration for WSN.



Proposed Detection Technique

The proposed detection technique consists of two phases.

1. The first phase is the construction of *Neighbor_Database* for every sensor node. *Neighbor_Database* contains two fields, node IDs of neighbour nodes and the distance from BS via corresponding neighbour node in terms of DBS value.
2. The second phase is the detection of the sinkhole node by its neighbour nodes. Each node calculates *Difference* parameter based on the DBS values of its *Neighbor_Database* and compares with a constant value called threshold to take decision of sinkhole attack.

Phase 1: Neighbor Database Construction

Since the proposed approach uses DBS as the main parameter for sinkhole node detection, each sensor node creates a database of their neighbor nodes called *Neighbor_Database*. *Neighbor_Database* contains neighbor node IDs and their corresponding DBS values. At first, each node considers another

node as its neighbor if it is within its radio range *'R'*. After that each neighbor node (Ni) of BS starts the DBS (Euclidean distance of Ni from BS) calculation, further neighbor node (Nj) of these nodes takes this DBS value and adds their intermediate distance (Euclidean distance of Nj from Ni). Next this node (Nj) puts the neighbor *node_id* (Ni) and the updates DBS value inside the *Neighbor_Database*. In this way, *Neighbor_Database* has been created for each node.

For example, Table 2 shows the initial *Neighbor_Database* of node 6 of scenario 1 (depicted in Fig. 2, section IV) before sorting. First column of the table contains all the neighbor *node_ids* and the second column contains distance from node 6 to BS via corresponding nodes in terms of DBS value. That means if node 6 sends its data via node 31 to BS it will take 74.97m distance cost.

Table 2 Initial Neighbor_Database of node 6

Node 6	
Node_ID	**DBS**
31	74.97
41	83.27
38	66.17
16	83.27
19	79.93
22	51.65
35	55.75

Table 3. Algorithm 1: Neighbor database construction algorithm

Algorithm 1 Neighbor Database Construction Algorithm	
1:	**for** node n: i to N **do**
2:	*Open_Database* DB_{ni}
3:	**for** node j to N (j^1 i) **do**
4:	**if** Euclidean_Distance(node i, node j) £ R **then**
5:	$DBS_i = DBS_j + Euclidean\_Distance(nodei, nodej)$
6:	*Push* ID_j *and* DBS_i *to* DB_{ni}
7:	**end if**
8:	**end for**
9:	**end for**

Algorithm 1 demonstrates the working of phase 1: *Neighbor_Database* construction

1. After the random deployment each node considers another node as its neighbour if their distance is less than or equal to 'R'.
2. Neighbour nodes (Ni) of BS calculate the Euclidean distance and show it as their DBS value.
3. Next each node (Nj) takes their neighbour node's (Ni) DBS value and adds it with their intermediate Euclidean distance.
4. Each node (Nj) adds the updated DBS value and the corresponding *node_ids* (Ni) in their database.
5. The process continue until each node is having all the neighbours in its database.

Phase 2: Sinkhole Node Detection

To detect the sinkhole attack, a node is selected randomly to make a sinkhole node. After the selection, the DBS value is decreased so that it can show a fake path to BS with very short distance as shown in Fig. 1. Since each node has limited resources and cannot store global information, a node can only use local information to detect sinkhole attacks. To detect sinkhole attack, at first, each sensor node's *Neighbor_Database* needs to be sorted. Next, separate the lowest DBS and corresponding *node_id*. If there are more than one lowest DBS, it separates them all. After this average DBS is calculated excluding the separated lowest DBS value. Now to check whether the corresponding node of the lowest DBS is suspicious or not it calculates the *Difference (%)* value with the help of (1). If the difference is greater than threshold value, the corresponding node is suspicious.

$$Difference\left(\%\right) = \frac{Average\_DBS - Minimum\_DBS}{Average\_DBS} \times 100\% \quad (1)$$

Table 4. Neighbor_Database of node 6 after sorting

Node 6	
Node_ID	**DBS**
22	51.65
35	55.75
38	66.17
31	74.97
19	79.93
16	83.27
41	83.27

After sorting *Neighbor_Database* of node 6, it looks like as Table 3. In this table, the minimum DBS value is 51.65 via node 22. So, node 22 will be excluded and average DBS will be calculated for rest of the nodes' DBS values.

Algorithm 2 demonstrates the working of phase 2: Sinkhole Node Detection

Table 5. Algorithm 2: Sinkhole node detection algorithm

	Algorithm 2 Sinkhole Node Detection Algorithm
1:	*Initialize threshold* T_h
2:	**for** node n: i to N **do**
3:	*sort* DB_{ni} *by DBS value*
4:	*lowest DBS* DBS_{li}=*minimum_DBS*(DB_{ni})
5:	*lowest DBS nodeID* $_i$=*ID*(DBS_{li})
6:	*AverageDBS* DBS_{avg}=*Average*(DB_{ni}−DBS_{li})
7:	*Difference*= [(DBS_{avg}−DBS_{li})/DBS_{avg}]×100
8:	**if** *Difference*^{3}T_h **then**
9:	**return** *"Sinkhole Node Detected" to Base Station*
10:	**end if**
11:	**end for**

1. Short the DBS values of each node's *Neighbor_Database*.
2. Separate the lowest DBS value and its corresponding *node_id*.
3. Calculate average DBS excluding the lowest DBS.
4. Calculate the *Difference (%)* value.
5. If it is greater than threshold value, node activity is suspicious and show sinkhole detected.

As the proposed mechanism of (Ibrahim, Rahman, & Roy, 2015) needs to send a Hello packet to each sensor node from BS for calculating the hop count value. In comparison with that our proposed scheme doesn't require any extra communication cost as no communication is required here.

Simulated Experimental Analysis and Results

Simulation Parameters

The proposed approach needs to know the DBS value of each node and to calculate the DBS value the node positions and their intermediate Euclidean distances is required. As shown in Table 4, the simulation is performed over a 100 x100 m^2 area. The simulation is done for 50, 100 and 400 nodes. It resulted in three scenarios (to affirm the scalability of the proposed scheme) - 50 nodes for scenario 1, 100 nodes for scenario 2 and 400 nodes are taken for scenario 3. For each scenario single Base Station (BS) is taken. Those sensor nodes are deployed randomly over a specific region. After deployment the node position is fixed for the entire process. One random node is selected, and its DBS value is decreased manually to make it sinkhole node. The threshold value is taken between 10% to 90% for best result. The radio range (R) is 20m. The simulation has been carried out in Java language.

Table 6. Simulation Parameters

Network Parameters	Values taken
Simulation Area	100x100 m^2
Number of Nodes	50, 100, 400
Threshold Value	10% - 90%
Radio Range	20m

Performance Metric

For this work, the detection rate is calculated as

$$Detection\_Rate(\%) = \frac{Number\_of\_Detected\_Sinkhole\_Node}{Total\_Number\_of\_Sinkhole\_Node} \times 100 \quad (2)$$

After calculating the *Detection_Rate* a graph is plotted with the DBS value, so the accuracy can be checked with the change of DBS values.

Experimental Result Analysis and Discussion

Fig. 2 shows the network for scenario 1 before placing any attack. Here, each node is having their own minimum distance from the base station.

Figure 2. Network without any attack scenario 1

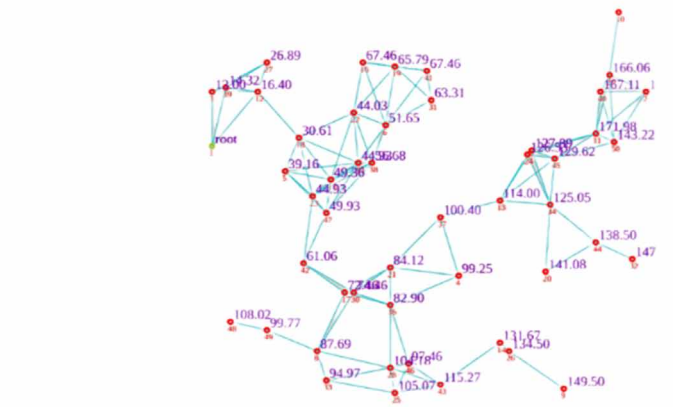
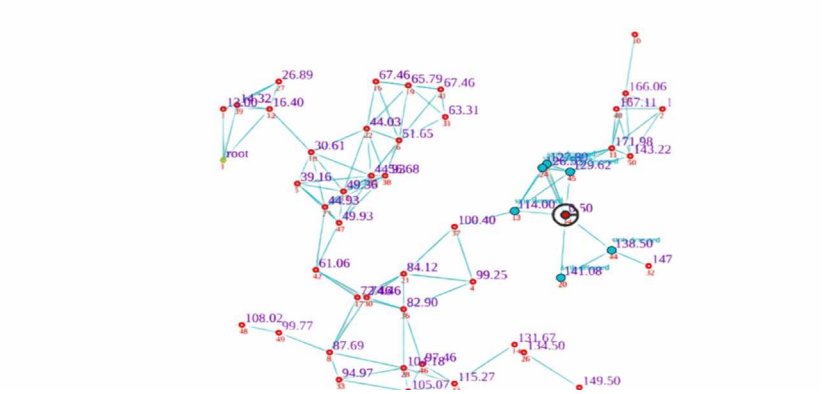


Fig. 3 shows the network after sinkhole attack detection. Here node 34 is a malicious node and all its neighbor nodes are showing that sinkhole is detected.

Figure 3. Network after an attack detection



At first the proposed approach requires an optimum threshold value, after 5 to 10 iterations it ranges from 50% to 60%. After that the comparison of each node's average DBS and the lowest DBS based on a particular node's database has been compiled. Fig. 4, Fig. 5, and Fig. 6 represent the relation between node position with respect to BS and the detection rate for scenarios - 1, 2, and 3 respectively. Near to the BS detection rate is almost 75% for both scenarios 1 and 2 and almost 60% for scenario 3. As the distance is increasing, detection rate is also increasing, and it has 100% detection rate for distance ^{3}35 m.

Figure 4. Detection rate based on distance for scenario 1.

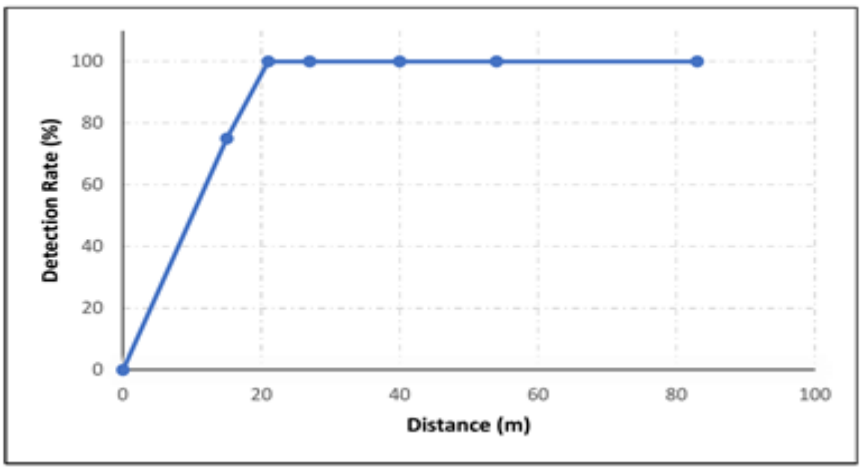


Figure 5. Detection rate based on distance for scenario 2.

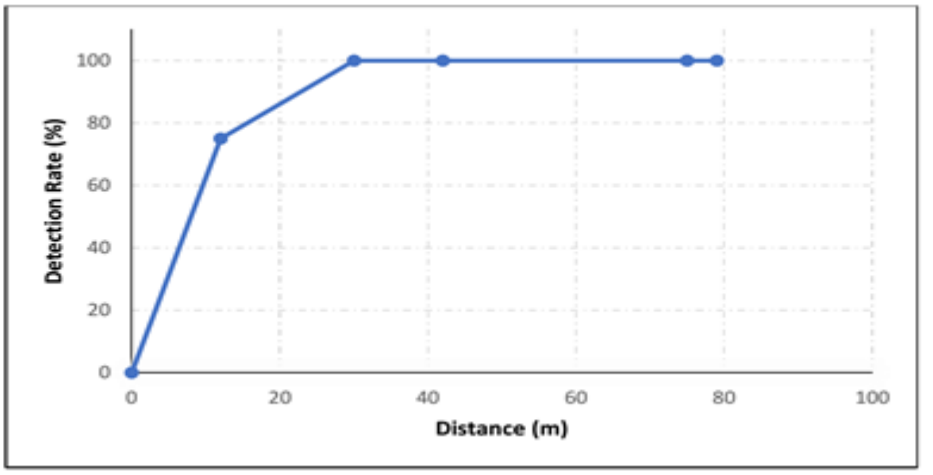


Figure 6. Detection rate based on distance for scenario 3.

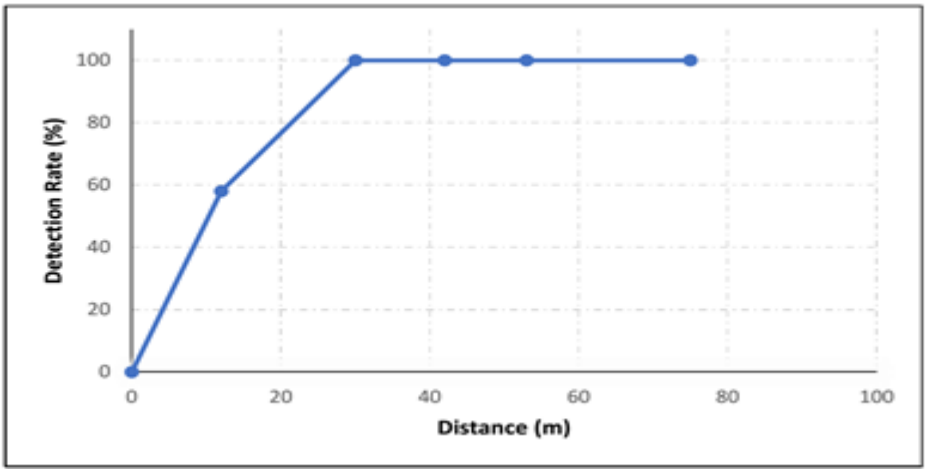


Table 7. Result Comparison

Area of Comparison	Ibrahim et al. (Ibrahim, Rahman, & Roy, 2015)	Proposed Mechanism
Min node distance from BS for detection	25m	>0
Detection Rate near BS	Can't detect	[3] 60%
Min node distance from BS for100% detection rate	[3]70m	[3] 35m

Result comparison of the proposed mechanism with Ibrahim et al. (Ibrahim, Rahman, & Roy, 2015) is shown in Table 5.

CONCLUSION

This paper presented a cost-effective algorithm for detecting sinkhole attack in WSN. This detection technique only required the node location to calculate the DBS value. Based on changes of DBS values, it detected sinkhole attack. As the distance increases detection accuracy of proposed method also increased. The proposed detection mechanism of Ibrahim et al. completely fails to detect the sinkhole attack for the neighbor nodes of BS. For the neighbor node case, the proposed DBS mechanism achieved 55% to 75% detection rate for scenarios under consideration. The proposed method attained the 100% sinkhole attack detection accuracy with distance greater than 25m from BS for scenario 1, and distance greater than 35m from BS for scenarios 2 and 3.

ACKNOWLEDGMENT

The publisher has waived the Open Access Processing fee for this article.

REFERENCES

Aad, I., Hubaux, J.-P., & Knightly, E. W. (2004). Denial of service resilience in ad hoc networks. In *Proceedings of the 10th annual international conference on Mobile computing and networking* (pp. 202–215). doi:10.1145/1023720.1023741

Babaeer, H. A., & Al-Ahmadi, S. A. (2020). Efficient and secure data transmission and sinkhole detection in a multi-clustering wireless sensor network based on homomorphic encryption and watermarking. *IEEE Access : Practical Innovations, Open Solutions*, *8*, 92098–92109. doi:10.1109/ACCESS.2020.2994587

Butun, I., Morgera, S. D., & Sankar, R. (2013). A survey of intrusion detection systems in wireless sensor networks. *IEEE Communications Surveys and Tutorials*, *16*(1), 266–282. doi:10.1109/SURV.2013.050113.00191

Choi, B. G., Cho, E. J., Kim, J. H., Hong, C. S., & Kim, J. H. (2009). A sinkhole attack detection mechanism for LQI based mesh routing in WSN. *2009 International Conference on Information Networking*, 1–5.

Dallas, D., Leckie, C., & Ramamohanarao, K. (2007). Hop-count monitoring: Detecting sinkhole attacks in wireless sensor networks. *2007 15th IEEE International Conference on Networks*, 176–181.

Grover, J., Laxmi, V., & Gaur, M. S. (2013). Attack models and infrastructure supported detection mechanisms for position forging attacks in vehicular ad hoc networks. *CSI Transactions on ICT, 1*, 261–279.

Healy, M., Newe, T., & Lewis, E. (2009). Security for wireless sensor networks: A review. *2009 IEEE Sensors Applications Symposium*, 80–85. doi:10.1109/SAS.2009.4801782

Hu, Y.-C., Perrig, A., & Johnson, D. B. (2006). Wormhole attacks in wireless networks. *IEEE Journal on Selected Areas in Communications*, *24*(2), 370–380. doi:10.1109/JSAC.2005.861394

Ibrahim, A., Rahman, M. M., & Roy, M. C. (2015). Detecting sinkhole attacks in wireless sensor network using hop count. *International Journal of Computer Network and Information Security*, *7*(3), 50–56. doi:10.5815/ijcnis.2015.03.07

Jawandhiya, P. M., Ghonge, M. M., Ali, M. S., & Deshpande, J. S. et al. (2010). A survey of mobile ad hoc network attacks. *International Journal of Engineering Science and Technology*, *2*, 4063–4071.

Karlof, C., & Wagner, D. (2003). Secure routing in wireless sensor networks: Attacks and countermeasures. *Ad Hoc Networks*, *1*(2-3), 293–315. doi:10.1016/S1570-8705(03)00008-8

Law, Y. W., Hartel, P., den Hartog, J., & Havinga, P. (2005). Link-layer jamming attacks on S-MAC. *Proceedings of the Second European Workshop on Wireless Sensor Networks,* 217–225.

Maheshwari, R., Gao, J., & Das, S. R. (2007). Detecting wormhole attacks in wireless networks using connectivity information. *IEEE INFOCOM 2007-26th IEEE International Conference on Computer Communications*, 107–115.

Medeira, P., Grover, J., & Khorjiya, M. (2019). A survey on detecting application layer ddos using big data technologies. *Journal of Emerging Technologies and Innovative Research*.

Nagel, N. R., Shokranian, R., Bordim, J. L., & Nakano, K. (2008). MAC layer misbehavior on ad hoc networks. *2008 IEEE/IFIP International Conference on Embedded and Ubiquitous Computing,* 2, 538–542. doi:10.1109/EUC.2008.152

Nayak, P., Suseela, R. S., & Trivedi, V. (2017). A review on DoS attack for WSN: Defense and detection mechanisms. *2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (ICECDS)*, 453–461. doi:10.1109/ICECDS.2017.8390208

Ngai, E. C., Liu, J., & Lyu, M. R. (2006). On the intruder detection for sinkhole attack in wireless sensor networks. *2006 IEEE International Conference on Communications,* 8, 3383–3389. doi:10.1109/ICC.2006.255595

<prpt>Padmavathi, D. G., & Shanmugapriya, M. (2009). A survey of attacks, security mechanisms and challenges in wireless sensor networks. arXiv preprint arXiv:0909.0576.</prpt>

Pal, V., Singh, G., & Yadav, R. P. (2015). Balanced cluster size solution to extend lifetime of wireless sensor networks. *IEEE Internet of Things Journal*, *2*(5), 399–401. doi:10.1109/JIOT.2015.2408115

Pathan, A.-S. K., Lee, H.-W., & Hong, C. S. (2006). Security in wireless sensor networks: Issues and challenges. *2006 8th International Conference Advanced Tongxin Jishu*, *2*, 6.

Pinar, Y., Zuhair, A., Hamad, A., Resit, A., Shiva, K., & Omar, A. (2016). Wireless sensor networks (WSNs). *2016 IEEE Long Island Systems, Applications and Technology Conference (LISAT)*, 1–8.

Pirzada, A. A., & McDonald, C. (2005). Circumventing sinkholes and wormholes in wireless sensor networks. *IWWAN'05: Proceedings of International Workshop on Wireless Ad-hoc Networks,* 71.

Pundir, S., Wazid, M., Singh, D. P., Das, A. K., & Rodrigues, , J., & Park, Y. (2020). Designing efficient sinkhole attack detection mechanism in edge-based IoT deployment. *Sensors (Basel)*, *20*, 1300. doi:10.3390/s20051300 PMID:32121017

Riaz, M. N., Buriro, A., & Mahboob, A. (2018). Classification of attacks on wireless sensor networks: A survey. *International Journal of Wireless and Microwave Technologies*, *8*(6), 15–39. doi:10.5815/ijwmt.2018.06.02

Salehi, S. A., Razzaque, M. A., Naraei, P., & Farrokhtala, A. (2013). *Detection of sinkhole attack in wireless sensor networks. In 2013 IEEE international conference on space science and communication*. IconSpace.

Sejaphala, L. C., & Velempini, M. (2020). The Design of a Defense Mechanism to Mitigate Sinkhole Attack in Software Defined Wireless Sensor Cognitive Radio Networks. *Wireless Personal Communications*, *113*(2), 977–993. doi:10.1007/s11277-020-07263-9

Singh, J., Yadav, S. S., Kanungo, V., & Pal, V. et al. (2021). A Node Overhaul Scheme for Energy Efficient Clustering in Wireless Sensor Networks. *IEEE Sensors Letters*, *5*(4), 1–4. doi:10.1109/LSENS.2021.3068184

Singh, R., Singh, J., & Singh, R. (2016). *WRHT: A hybrid technique for detection of wormhole attack in wireless sensor networks*. Mobile Information Systems.

Singh, V. P., Jain, S., & Singhai, J. (2010). Hello flood attack and its countermeasures in wireless sensor networks. *International Journal of Computer Science Issues*, *7*, 23.

Walfish, M., Vutukuru, M., Balakrishnan, H., Karger, D., & Shenker, S. (2010). DDoS defense by offense. *ACM Transactions on Computer Systems*, *28*(1), 1–54. doi:10.1145/1731060.1731063

Walters, J. P., Liang, Z., Shi, W., & Chaudhary, V. (2007). Wireless sensor network security: A survey. In *Security in distributed, grid, mobile, and pervasive computing* (pp. 367–409). Auerbach Publications. doi:10.1201/9780849379253-20

Wood, A. D., & Stankovic, J. A. (2002). Denial of service in sensor networks. *Computer, 35*, 54–62.

Satyendra Singh Yadav has received his Bachelor of Engineering in Electronics and Communication from Rajiv Gandhi Proudyogiki Vishwavidyalaya (RGPV) Bhopal a State University of Madhya Pradesh (India), in 2012. In 2018 he received his PhD from National Institute of Technology, Rourkela (India). He was with Instituto de Engenharia de Sistemas e Computadores - Investigação e Desenvolvimento (INESC-ID), Instituto Superior Técnico (IST) Lisbon, Portugal under India-EU NAMASTE mobility project during 2015 to 2016. Dr. Yadav is currently working as an Assistant Professor in the department of ECE at National Institute of Technology Meghalaya (India). He is also serving as Faculty Incharge Computer Center NIT Meghalaya Since July-2020. Prior to joining NIT Meghalaya in Oct. 2019, he has worked as a full-time faculty member at IIITDM Kurnool and IIIT Vadodara. He is actively serving as a reviewer for many IEEE, Elsevier. Wiley, Springer, journals, and conferences. He is also serving as TPC members for various IEEE conferences. Dr. Yadav is currently supervising 2 PhD scholars. His research interests include wireless communication, Resource allocation, Parallel computing, Machine learning, as well as GPU acceleration for 5G and beyond wireless systems. Since 2014, he has been a member of IEEE.

Vipin Pal has received the Ph.D. degree from the Malaviya National Institute of Technology Jaipur, India in 2016. He is currently working as Assistant Professor in the Department of Computer Science and Engineering, National Institute of Technology Meghalaya, India. His research interests are IoT, Wireless Sensor Networks, Soft Computing, Data Mining.

Akhilendra Pratap Singh (Member, IEEE) received the B.Tech degree in Computer Science & Engineering from Uttar Pradesh Technical University Lucknow, Uttar Pradesh, India, in 2006, the M. Tech. degree in Information Security from Motilal Nehru National Institute of Technology Allahabad, Uttar Pradesh, India, in 2011, and the Ph.D. degree in information technology from Indian Institute of Information Technology Allahabad, Uttar Pradesh, India, in 2017. He is currently an Assistant Professor with the Department of Computer Science and Engineering, National Institute of Technology Meghalaya, India. He has authored or co-authored a large number of various research papers in IEEE Transaction, journals and International conferences of high repute. His research interests lie in Blockchain Technology, Service-oriented computing, Wireless sensor network, Network security, Network forensics and machine learning. Dr. Singh is associated with more than 10 international journals of repute as reviewer.

Yogita is working as Assistant Professor in the Department of Computer Science and Engineering at National Institute of Technology Meghalaya, India. Her research interest includes data mining, text mining, and their interdisciplinary applications.

Mangal Singh is working as Associate Professor in the Department of Electronics & Telecommunication Engineering at Symbiosis institute of Technology, Pune. He has an experience of more than 20 years in the field of Teaching, Research and Administration. He obtained his graduation in Electronics and Telecommunication Engineering from National Institute of Technology (formally known as GEC), Raipur, Chhattisgarh and M Tech in Communication Engineering Jadavpur University, Kolkata, West Bengal in 2000 and 2006, respectively. Dr Singh obtained his Ph D in Communication Engineering from National Institute of Technology, Rourkela, Odisha, in 2017. He has served as Associate Professor, Electronics & Communication Engineering, Institute of Technology, Nirma University, Ahmedabad from August 2018 to September 2021 and Associate Professor, Electronics & Communication Engineering, Chhatrapati Shivaji Institute of Technology, Durg, Chhattisgarh from September 2001 to July 2018. He has published more than 10 research papers in the area signal processing for communications, particularly multi-carrier modulation (OFDM) for wireless communication systems in International refereed/peer reviewed Journals and presented/published more than 20 papers in National/ International Conferences/Proceedings. He has 3 Indian patents published in his credit.He has guided more than 05 PG dissertations. He is a Senior Member of IEEE and life member of the IETE and ISTE, India.