PCAD: Power Control Attack Detection in Wireless Sensor Networks

Prathap U, Deepa Shenoy P and Venugopal K R
Department of Computer Science and Engineering
University Visvesvaraya College of Engineering
Bangalore University, India
prathap.u@gmail.com

***Abstract*—Security in wireless sensor networks is critical due to its way of open communication. In this paper we have provided a solution to detect malicious nodes which perform radio transmission power control attack and sinkhole attack in wireless sensor networks. In the proposed approach, data transmission is divided into multiple rounds of equal time duration. Each node chooses the parent node in the beginning of the round for forwarding the packet towards sink. Each node adds its identity in the packet as a routing path marker and encrypts before forwarding to parent. Child node observes the parent, handles acknowledgement from 2-hop distance node and decides the trust on parent based on successful and unsuccessful transactions. Each node sends a trust value report via multiple paths to Sink at the end of the round. Sink identifies the malicious node by comparing trust value report received from each node with number of data packets received. Simulated the algorithm in NS-3 and performance analysis compared with other recently proposed approach. Simulation results show that proposed method detect the malicious nodes efficiently and early.**

***Index Terms*—WSN, trust based, malicious node, power control attack, bad mouthing attack, sinkhole attack.**

I. Introduction

A wireless sensor network (WSN) consists of spatially distributed autonomous devices having sensing, computing and communication capabilities. Sensor nodes cooperatively monitor physical or environmental conditions, such as temperature, pressure, sound, vibration, motion or pollutants. Wireless sensor networks are used in environmental conditions where information is difficult to access. Sensor node, also known as a 'mote', is a node in a wireless sensor network that is capable of performing some processing, gathering sensory information and communicating with other connected nodes in the network. Sensor network transmits the data from one node to another node in an adhoc way and finally to a base station where the data is stored, processed and displayed.

Sensor nodes are vulnerable to a wide range of attacks [1]. Attacker can listen to radio transmissions, modify the packet before forwarding, misroute the packet to unintended next hop node, inject false data in the channel, replay previously heard packets to drain the energy of other nodes as battery power is crucial in nodes. Attacker may deploy few malicious nodes with similar or better hardware capabilities or by 'turning' few legitimate nodes by capturing them and physically overwriting their memory. Sybil attack - attacker deployed nodes may also use the identities of the other genuine nodes to frame other genuine nodes as malicious. Packet dropping, modification, misrouting are basic problems which have large impact on the information gathered by sensor nodes as network loses lot of important sensed data. Cryptography techniques alone are not sufficient to protect the data. Attacks such as colluding collision[2], misrouting, power control, sinkhole, wormhole, rushing attacks can be launched without the help of cryptography keys [3].

If a node has the ability to control its power to transmit the data, then it can vary the radio range of data transmission[4]. In Figure 1, if node Y has ability to control the power to vary its data transmission distance, then node Y can use less power such that only Z and other neighbor nodes hear the packet forwarding from Y but X does not hear the packet forwarding from Y as X is farther than any other node. Sender and other neighbor nodes feel that Y has actually transmitted the packet to X but intended recipient X missed to receive the packet. If node Y is successful in achieving the power control attack for all packets to be forwarded then the node Y resembles the sinkhole attack without being detected from others. In sinkhole attack[5] malicious node attracts the routing data by publishing the shortest path to Sink and drops all the packets without forwarding further to Sink. Power control attack is smart way of achieving the sinkhole attack.

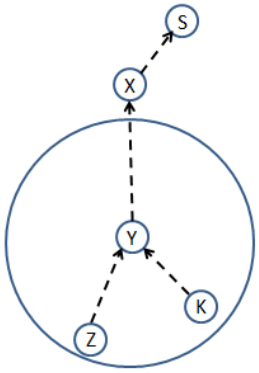

Fig. 1. Power Control Attack Description

In this paper, we propose a scheme 'Power Control Attack Detection (PCAD)' to identify malicious nodes which performs power control attack and sinkhole attack. In figure

1, node Z observes the next hop node Y to detect any sinkhole attack, and expects an acknowledgement from 2-hop distance node X to detect power control attack by 1-hop distance node Y. X does not reply acknowledgement unless it receives packet from Y. Node Z maintain the count of successful and unsuccessful packet transmission from Y based on 2-hop acknowledgement from X. Node Z sends the report to sink S at the end of each round of operation using multiple paths through all selected parents. Sink detects the malicious nodes based on the received data packets and also received reports from all nodes.

In order to detect the sinkhole attack 'Detection of Sinkhole Attack(DSHA)[6]' has been proposed recently in the literature. DSHA first identifies the area of network where malicious node exists and then tries to find the malicious node in the identified area. we provide a simulated analysis comparing the DSHA approach and our proposed approach. The rest of the paper is organized as follows, section II discusses about the related work, section III describes the network model and problem statement, section IV presents the solution and algorithm, section V provides the performance analysis and results, and section VI concludes the work and discusses the future challenges.

II. Related Work

To mitigate the security attacks and improve the reliability, multi-path routing [7], [8], [9], [10] approach has been proposed, where multiple copies of the packet are forwarded to Sink node along the different paths available. Neighbor node observation or monitoring is another approach [11], [12], [13], [14] used to find the malicious activity of the current forwarding node. In monitoring approach, observer nodes monitor the current sender and current receiver for the packet being transmitted. Observers observe for various malicious activities such as packet dropping, modification, power control, sinkhole attacks. Monitoring methods require observer nodes to buffer the packets which are forwarded to next hop node and compare the packet forwarded by next hop node with its buffered packet to find out packet modifications. In [15] both observation based and trust based techniques are used to detect the malicious nodes performing various attacks, but the approach becomes inefficient with the introduction of power control attack. In [2], [4] power control attack has been considered but the approach needs to have observer nodes in the common radio range of the current sender and receiver.

Energy consumption in both multipath routing and neighborhood monitoring is not affordable for sensor networks. In multipath routing, energy is consumed from nodes along multiple path to Sink to transmit same copy of data. In monitoring approach, many nodes observe each hop while a packet being forwarded and energy of all the observer nodes consumed. In [3], energy efficient sleep-wake approach along with local monitoring method is used to detect malicious nodes but cannot control the bad mouthing attack from observers and also need enough number of observes to make this approach feasible. In [5], an approach to detect the sinkhole attack has been proposed based on the CPU usage, but the false positive increases by detecting less utilized node as malicious. Paper[6], proposes a sinkhole detection method based on network flow information and routing pattern in the network but has high false isolation of the genuine nodes.

III. Model and Problem Statement

A. Network Model

We considered wireless sensor network with one Sink node with all the sensor nodes are randomly distributed [16]. After deployment, network initialization and routing path building starts with Sink node. Sink node transmits the path distance information to one hop neighbors. One hop neighbors increment the distance information and share with two hop neighbors and continues till the last hop node. Each node selects a list of parent nodes which have equal and shortest distance to Sink node. Each node selects a parent node among the identified parent nodes and sends parent selection information to Sink. Sink establishes a routing tree rooted at Sink node based on the information received from each node. Data transmission is divided into rounds of equal time duration. Each node chooses a different parent node in the beginning of a round or phase among the selected parents based on the trust they have on the parent.

Intermediate node prepares marker data containing node identity, encrypts the marker data and adds to the packet before forwarding the packet to parent node. Marker data added by each node helps Sink to trace the nodes in the routing path [16]. All the nodes transmit the sensed data towards Sink for processing.

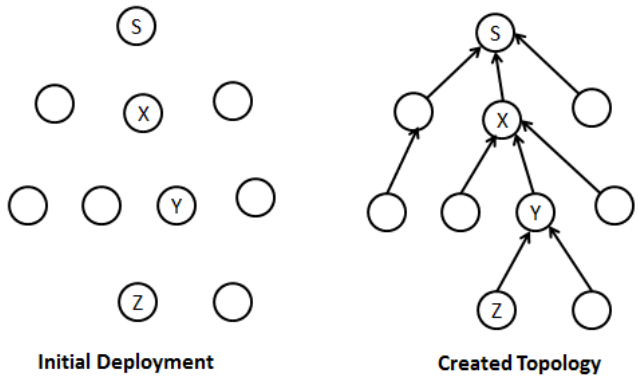

Fig. 2. Deployment and Topology

System Assumptions: PCAD assumes the network is static and the links are bidirectional. PCAD assumes that pair wise keys are shared between Sink and each network node before deployment. Assumed no malicious activity during topology creation. In PCAD each node knows the current *(X,Y)* location and also the location of the neighbor nodes. Source nodes are assumed to be genuine. Assumed that during network initialization a node establishes pair wise keys with two hop distance nodes.

B. Problem Definition

The goal of the PCAD is to detect the malicious nodes which perform power control and sinkhole attacks. In figure 2, node *Z* transmits packet to *Y* to forward towards Sink and node *Y* transmits packet to node *X* to forward towards Sink. If node *Z* is a source node then following are the problems to be detected. i) If node *Y* performs sinkhole attack, then node *Z* does not hear any packet forwarding from node *Y*. ii) If node *Y* performs power control attack, then node *Z* hears the packet forwarding from node *Y* but node *X* does not receive the packet to be forwarded towards Sink. iii) Node *X* does not send the 2-hop acknowledgement even though received the packet successfully from node *Y*, just to frame the node *Y* as malicious. iv) In above three scenarios, either node *Y* is malicious or node *Z* is malicious as both can perform attacks and restrain from sending acknowledgement. Problem is to detect malicious nodes among such pair of nodes <*X,Y*>.

IV. PCAD

PCAD has two modules to detect the malicious nodes perform power control attack and sinkhole attack. Module installed in individual sensor nodes observe the parent, receive two hop acknowledgment and build trust value on parent. Module installed in Sink node maintain the count of packets received on each path and compare with report sent by each sensor node.

A. Sensor Node Module

Figure 3, shows the success case of packet transmission. As a path marker, current sender *Z* adds the encrypted id to the received packet *P* from previous hop and forwards the packet *A* to next one hop node *Y* on the routing path. One hop node *Y* prepares packet *B* by adding the encrypted id and transmits the packet to *X*. Node *Z* also hears the packet transmitted from *Y* and compares with its buffer and waits for the acknowledgement from *X*. Node *Z* clears the buffer and confirms the successful transmission on receiving the acknowledgement. Acknowledgment *C* is encrypted with the shared key between *X* and *Z* to avoid the fabrication and modification of acknowledgement by *Y*.

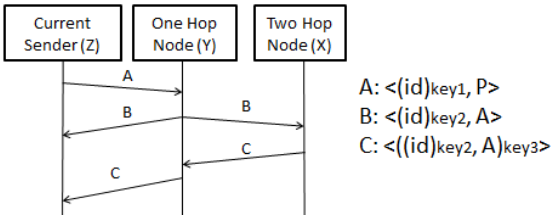

Fig. 3. Successful transmission of Packet

Figure 4, shows the power control attack from one hop forwarding node, where one hop node *Y* uses power such that only current sender *Z* hears the packet but two hop node *X* does not receive the packet. Node *Z* does not receive the acknowledgment from two hop node *X* and after the timeout node *Z* determines the power control attack from one hop node *Y* and reduces the trust value on *Y*.

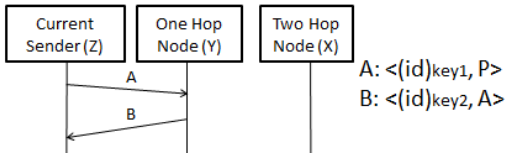

Fig. 4. Power control attack from one hop node

Figure 5, shows the sinkhole attack from one hop node *Y*. Node *Z* does not hear the packet transmission from *Y* even after the timeout period. Node *Z* determines the sinkhole attack from one hop node *Y* and reduces the trust value on *Y*.

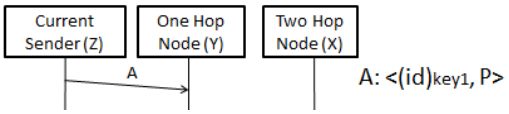

Fig. 5. Sinkhole attack from one hop node

Figure 6, shows that one hop node *Y* does not receive the acknowledgement from two hop node *X*, then *Y* reduces the trust value on *X*. And *Z* does not receive the acknowledgment from *Y* and reduces the trust value on *Y*.

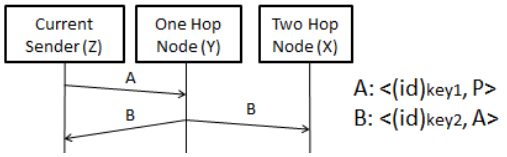

Fig. 6. No acknowledgment from two hop node

Each sensor node builds the trust value based on successful or unsuccessful transactions with parent node and sends the trust value report on parent to Sink node using all the selected parents in forwarding path to Sink.

B. Sink Node Module

Sink starts processing the packet on receive to update the count of the packet a node has participated either in forwarding or generating. The received packet at Sink consists of sequence of encrypted node ids which are path markers added by each forwarding node and also sensed data from source node. Sink starts the decryption process with below steps.

i) First marker information of message m is decrypted with key of first level child node say *X* of Sink to generate $m^{'}$. If $m^{'}$ starts with $< X >$ then *X* is the forwarded node. Else Sink decrypts with key of next immediate first level child node and tries to match the marker information.

ii) If marker information does not match with any of the first level children, then Sink decrypts the complete message with key of first level child say *X* to generate $m^{'}$. If $m^{'}$ starts with $< X, D >$ then *X* is the source node. Else Sink decrypts

with key of next first level child node and tries to check for source node.

iii) If marker matches a node say *X* in step *i*, then $m^{'}$ is updated $m^{'} = m^{'} - < X >$ by removing the marker added by *X* and packet count of node *X* is incremented by one. Now the step *i* and step *ii* are performed for all children of *X* to match for forwarding node or source node.

Notations:

m: received packet at Sink
U, V, S: node id
$pcount_v$*: packet count maintained by Sink for node V*
V_{key}*: shared key between Sink and node V*

Algorithm 1: packet count update for each node

1: Input: Packet <m>
2: U = S, m = m; success = false;
3: ***for*** *each child node V of node U* ***do***
*4: P = decMarker(V_{key}, m); /*decrypts only marker which is two units*/*
5: ***if*** *P starts with <V>* ***then***
6: $pcount_v$*++;*
7: trim <V> from P and get m = P-<V>;
8: U = V; endfor;
9: ***endfor****;*
10: ***for*** *each child node V of node U* ***do***
*11: P = decSourceMsg(V_{key}, m); /*decrypts source message which is two units*/*
12: ***if*** *P starts with <V, D>* ***then*** */*V is the source node*/*
13: $pcount_v$*++;*
14: ***endfor****;*

the packet count recorded by Sink while processing the packet will help Sink to determine the malicious node when Sink receives the report from each node.

C. Malicious Node Detection from Sink

Each node prepares report containing a tuple $<V, V_c, P_v, T>$ where *V* is node id, V_c is count of the packets node *V* has forwarded and generated, P_v is the parent in the current round of operation, *T* is the trust value on parent. Copies of the report is sent to Sink node through all the parent nodes selected during initialization of network. Multiple copies are sent to make sure atleast one copy of the report reaches Sink node in the presence of malicious nodes which perform the sinkhole attack. Sink considers one copy even though it receives more than one copy of the same report.

Notations:

R: vector of reports collected from each node
R_v*: report sent by node V*
R_v*[T]: trust value in the record sent by node V*
U, V: node id
$U_{packetcount}$*: sum of packet counts from all children of U*
U_t*: sum of the trust value from all children U*
U_{avgt}*: average trust value of U*
$U_{children}$*: total number of children of U*
dropthreshold: dropping threshold due to environmental errors
U_{pcount}*: packet count maintained by Sink for node U*
V_c*: count of packets node V has sent in record*
$T_{threshold}$*: trust threshold*

Algorithm 2: malicious node detection at Sink

1: ***for*** *each node U* ***do***
2: $U_{packetcount}$ *= 0;*
3: U_{avgt} *= 0;*
4: ***for*** *each child V of U* ***do***
5: R_v *= R[V];*
6: $U_{packetcount} = U_{packetcount} + R_v[V_c]$*;*
7: $U_t = U_t + R_v[T]$*;*
8: ***endfor****;*
9: $U_{avgt} = U_t / U_{children}$*;*
10: ***if*** *($U_{packetcount} - U_{pcount} >$ dropthreshold)* ***then***
11: ***if*** *($U_{avgt} < T_{threshold}$)* ***then***
12: mark node U as malicious;
*13:****endfor****;*

Sink receives the report from all child nodes of a parent node. Sink calculates the total number of packets parent must have forwarded by adding the packet count from each child report. Sink compares the packet count found from reports with the packet count Sink maintained during packet processing by sink node module. If the difference in the packet count is greater than the threshold, Sink compares the average trust of the parent node. If the average trust is less than the predefined threshold then parent node is marked as malicious node.

V. Performance Analysis

The efficiency and effectiveness of PCAD are evaluated in NS-3 simulator. We have compared proposed approach with DSHA [6]. Simulation is done by deploying 100 nodes randomly in a square area. Each node is installed with 802.15.4 MAC protocol and with channel delay of 2 milli seconds. Simulation ran with generating 50 packets per node. Non leaf nodes are randomly selected as malicious nodes. All nodes act as a source node and generate the data to forward towards sink. Obtained simulation results from the algorithm for various number of malicious nodes.

A. Percentage of Detection

Simulated and found the detection rate when the number of malicious nodes are 10, 20, 30, and 40.

*% detection = (No. of malicious nodes detected / No. of malicious nodes in network)*100*

For each quantity of malicious nodes, traffic is generated in 5 trails and averaged the detected malicious nodes in 5 trails. As shown in figure 7, percentage of detection is improved in PCAD when compare to DSHA approach. In DSHA, the percentage of detection deteriorates as the number of malicious nodes increases. PCAD detects malicious node by Sink considering the total packets transmitted and also reports from each node instead of focusing on certain area of the network as done in DSHA.

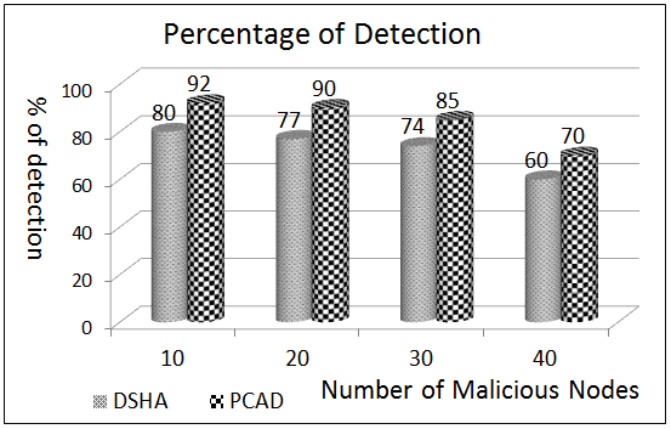

Fig. 7. Percentage of malicious node detection

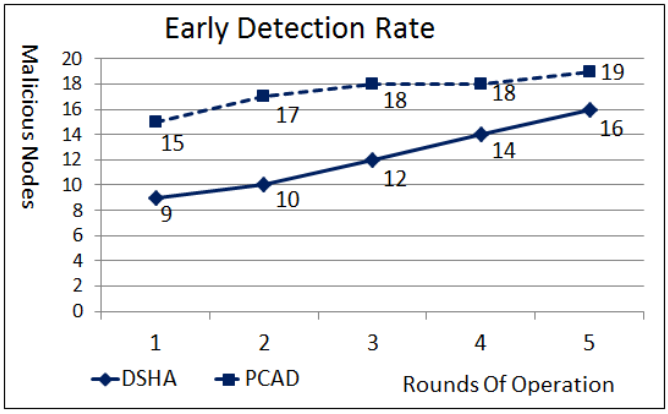

Fig. 9. Early Detection Rate

B. Percentage of False Isolation

Simulated and analyzed the false detection when the number of malicious nodes are 10, 20, 30, and 40.

*% false detection = (No. of genuine nodes isolated / No. of genuine nodes in network)*100*

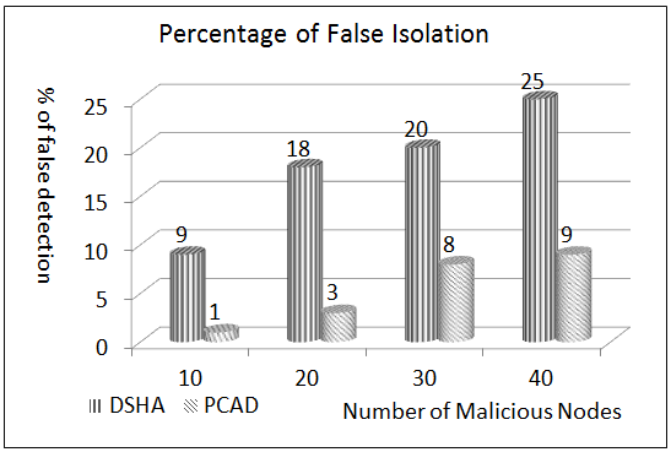

Fig. 8. Percentage of false isolation

As shown in figure 8, percentage of false detection is high in DSHA approach. In PCAD approach, considered trust from all children node to avoid bad mouth attack from a particular child which tries to frame the parent as malicious by sending low trust value to Sink. Sink detects the malicious nodes having the complete state of the network data transmitted.

C. Early Detection Rate

Simulated and analyzed the early detection when the number of malicious nodes are 20 in the network. In both PCAD and DSHA, traffic is generated in multiple rounds of equal duration and tried to find the malicious nodes after each round. DSHA needs long operation of the network to detect the malicious nodes.

As shown in figure 9, PCAD detects the malicious nodes early compare to DSHA, so that network cannot afford to loose lot of meaningful information before all malicious nodes are detected. Both Sink and child follows the same protocol for parent selection. After each round of operation, Sink determines the next possible parent node of the child among the parents list of the child. And even the child follows the same protocol[16] as Sink for parent selection for next round of operation. PCAD detects early as it operates in rounds, detects malicious after each round and child node can change parent node after each round.

D. Bad Mouthing Attack Analysis

PCAD detects the malicious nodes with the support of observation on parent and providing report of the packet transfer count to Sink node. In bad mouthing attack, a node gives false report on the neighbor node just to frame the neighbor node as malicious. In PCAD, child node can try to frame the parent node as malicious by adding the false packet count in report sent to Sink. Sink declares the malicious node only if the packet count difference is greater than threshold and also average trust value is less than threshold. Average trust value of a parent is calculated by averaging the trust values sent by each child node of a parent node. Average trust value is considered to avoid the bad mouthing attack from any particular child node.

VI. Conclusions and Future Directions

Power control and sinkhole are critical security attacks to disrupt the data and operation in wireless sensor networks. Proposed method is proven to be efficient to detect power control and sinkhole attacks compare to DSHA approach. PCAD starts with selection of parent for forwarding the data towards Sink. PCAD observes the parent and expects acknowledgement from 2-hop node to detect sinkhole and power control attacks. Early detection is possible as PCAD operation includes detection of malicious nodes after a round of operation. It also provides flexibility to change the parent node based on child node experience with parent node. PCAD approach does not lose lot of meaningful information as the node changes the parent after each round of operation. Performance results show that PCAD detect the malicious nodes early with high detection rate and

low false detection. Our future work includes providing a integrated solution which detects packet dropping, modifying, misrouting, using wrong identity along with power control and sinkhole attacks.

REFERENCES

[1] H. Chan and A. Perrig, "Security and Privacy in Sensor Networks," In *Computer*, volume 36, pp. 103-105, Oct 2003.

[2] Issa M. Khalil, Saurabh Bagchi, "Stealthy Attacks in Wireless Ad Hoc Networks: Detection and Countermeasure," In *IEEE Transactions On Mobile Computing*, volume 10, pp. 1096-1112, August 2011.

[3] Issa M. Khalil, "ELMO: Energy Aware Local Monitoring in Sensor Networks," In *IEEE Transactions on Dependable and Secure Computing*, volume 8, pp. 523-536, August 2011.

[4] Issa M. Khalil, "MPC: Mitigating Stealthy Power Control Attacks in Wireless Ad Hoc Networks," In *IEEE Global Telecommunications Conference*, pp. 1-6, August 2009.

[5] Changlong Chen, Min Song, and George Hsieh, "Intrusion Detection of Sinkhole Attacks In Large-scale Wireless Sensor Networks," In *IEEE International Conference on Wireless Communications, Networking and Information Security (WCNIS)*, pp. 711-716, June 2010.

[6] Ahmad S S., M.A. Razzaque, Parisa N,and A Farrokhtala, "Detection of Sinkhole Attack in Wireless Sensor Networks," In *Proc. IEEE International Conference on Space Science and Communication*, pp. 361-365, July 2013.

[7] C. Karlof and D. Wagner, "Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures," In *Proc. IEEE First Intl Workshop Sensor Network Protocols and Applications*, pp. 113-127, 2003.

[8] V. Bhuse, A. Gupta, and L. Lilien, "DPDSN: Detection of Packet-Dropping Attacks for Wireless Sensor Networks," In *Proc. Fourth Trusted Internet Workshop*, 2005.

[9] M. Kefayati, H.R. Rabiee, S.G. Miremadi, and A. Khonsari, "Misbehavior Resilient Multi-Path Data Transmission in Mobile Ad-Hoc Networks," In *Proc. Fourth ACM Workshop Security of Ad Hoc and Sensor Networks (SASN 06)*, 2006.

[10] R. Mavropodi, P. Kotzanikolaou, and C. Douligeris, "SECMRa Secure Multipath Routing Protocol for Ad Hoc Networks," In *Ad Hoc Networks*, volume 5, pp. 87-99, 2007.

[11] F. Ye, H. Luo, S. Lu, and L. Zhang, "Statistical En-Route Filtering of Injected False Data in Sensor Networks," In *Proc. IEEE INFOCOM,*, pp. 839-850, 2004.

[12] S. Zhu, S. Setia, S. Jajodia, and P. Ning, "An Interleaved Hop-by-Hop Authentication Scheme for Filtering False Data in Sensor Networks," In *Proc. IEEE Symp. Security and Privacy*, pp. 259-271, 2004.

[13] H. Yang, F. Ye, Y. Yuan, S. Lu, and W. Arbaugh, "Toward Resilient Security in Wireless Sensor Networks," In *Proc. Sixth ACM Intl Symp. Mobile Ad Hoc Networking and Computing (MobiHoc 05)*, 2005.

[14] Prathap U, Nisha K B, P Deepa Shenoy, and Venugopal K R, "SDLM: Source detection based local monitoring in wireless sensor networks," In *Proc. IEEE TENCON* , pp. 1-5, Nov 2015.

[15] Prathap U, P Deepa Shenoy, and Venugopal K R, "CMNTS: Catching malicious nodes with trust support in wireless sensor networks," In *Proc. IEEE TENSYMP* , pp. 77-82, May 2016.

[16] Prathap U, P Deepa Shenoy, and Venugopal K R, "CPMTS:Catching Packet Modifiers with Trust Support in Wireless Sensor Networks," In *Proc. IEEE WIECON-ECE* , pp. 255-258, Dec 2015.