International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Detection and Mitigation of Greyhole Attack in Wireless Sensors Network Using Trust Mechanism

Bansi S. Kantariya[1], Dr. Narendra M. Shekokar[2]

[1] Student, Computer Engineering Department, D. J. Sanghvi College of Engineering, Mumbai, India

[1] Professor, Computer Engineering Dept, D. J. Sanghvi College of Engineering, Mumbai, India

Abstract: ***Wireless sensors networks are generally deployed in mission critical environment. The nodes in the network are generally unattended and hostile environment in which they are deployed make the network vulnerable to many attack. One such attack is greyhole attack where the attacker selectively drops the packets in its path to sink or base station. Greyhole attack may reduce the network throughput or even lead to failure in the mission for which the network is deployed. So it becomes necessary to detect such attacks and mitigate it. Detection of greyhole attack is difficult as compare to other form of attack because the node implementing such attacks is the legitimate node having all the necessary cryptographic information. In this paper we have proposed the mechanism to detect and mitigate greyhole attack. We have used trust mechanism to detect the attack. Trust mechanism will calculate the trust value of the node in network which is similar to the concept of trust in human society and then this trust value is use to detect the malicious activity in our case packet dropping. We have used task completion and energy consumption as the parameters for calculating the trust value. There can be a situation where attacker can manipulated its attacking strategies to avoid itself from being detected. Our detection mechanism has also taken care of such situation.***

Keywords: Wireless sensors network, Greyhole attack, Trust mechanism.

1. Introduction

Wireless sensors network is the collection of autonomous, low power wireless sensor nodes. Wireless sensors network is mostly deployed in the mission critical environment. It is used for various monitoring purpose, for example it is use in industry to monitor the chance in parameters like temperature, pressure, sound etc. The sensors node sense the change in environment and forward the data to base station either by broadcasting the data directly or by forwarding it to some intermediate node. Mostly sensor node are unattended by human and the hostile environment in which they are deployed pose the serious security challenge for WSN.

Malicious attacker can capture the node and reprogram the node to implement the various types of attacks. An attacker can launch two type of attack, inside attack and outside attack. The attack that is launched using completely new node which is not authenticated by base station such attack is called outside attack. Inside attack is where attackers captures the authenticated nodes and reprogram these nodes to launch the attack. Insider attack includes attack like packet dropping attack, packet modification attack, misrouting attack, eavesdropping, etc. Outside attack are quite easy to detect as compare to inside attack because the node implementing the inside attack is authenticated and trusted. Even packet modification attack and misrouting attack is easy to detect. But the packet drop is difficult to detect therefore in this paper we are going to deal with one such type of packet drop attack.

In packet drop attack intermediate malicious node drop the packet instead of forwarding toward the base station. Packet drop attack can be of three types; they are blackhole attack, greyhole attack and on-off attack. The description about the attack is mention in table 1. Black hole attack is easy to detect as compare to greyhole and on-of attack.

Table 1: Type Of Packet Drop Attack [1]

Attack	Description
Blackhole attack	Drops all the packets entering the node.
Greyhole attack	Drops some packets entering the node.
On-off attack	Periodically drop the packets based on some pattern.

In this paper we are going to discuss the mechanism to detect and mitigate the greyhole attack in wireless sensor network. We will be using the trust mechanism to detect greyhole attack. A node will use trust mechanism to calculate the trust values of other node. This calculated trust values is equivalent to the notion of trust in human society. The trust value will then be used to detect the greyhole attack. Trust mechanism has two steps; first step is to monitor the behavior of the node and then next step is to calculate the trust value of the node. Once the node has calculated the trust value, it will check the trust value and decide whether there is an attack or not. If the node discovers an attack then it will report this attack to base station and then base station will take the required action.

The paper is organized as follow. In section 2 we have discussed what is greyhole attack is and what are the challenges faced while detecting greyhole attack. In section 3 we have discussed the literature survey based on the detection of greyhole attack. Our proposed detection mechanism is discussed in section 4. The paper is concluded in section 5.

2. Greyhole Attack

Greyhole attack is also called as selective forwarding attack. In greyhole attack malicious node selectively drops the messages that are to be forwarded to the sink node or base station. The decision to drop the packet is based on following two criteria [2].

- The source of the message or
- The type of the message.

The following figures illustrate the selective forwarding attack. In figure 1 the malicious node is dropping the packet based on the source of the packet. It is shown that the packet belonging to node B and node D are dropped. In figure 2 the malicious node is dropping the packet based on the type of the packet. As shown the malicious node is dropping the control packets.

As mention wireless sensors network is use for mission critical application like military surveillance. Presence of greyhole attack can corrupt such network by selectively dropping the important message. This can reduce the throughput of the network and can also lead to failure of a mission for which the network was deployed.

The major challenges face while detecting the greyhole attack are network congestion and power failure. The network congestion can also cause the packet drop, so the presence of congestion presents the difficulty in to detect the attack. The sensor node has limited battery life and if the there is power failure in intermediate node then this may also lead the packets to drop at failed node. So it is necessary to design the protocol which can detect the attack in presence under this condition. Therefore there is a need to distinguish whether the packed are being dropped due to above mention reason or due to malicious activity while designing the detection mechanism.

3. Literature Survey

The greyhole attack was first mention by Karlof [3]. Author has mentioned various type of attack which includes greyhole attack. He has also discussed how this attack can be implemented in various routing algorithm and suggested the countermeasure to avoid the attack. After this there have being many literature based on the detection of greyhole attack.

Xiao [4] has proposed the Checkpoint-Based Multi-Hop Acknowledge Scheme (CHEMAS) to detect the selectively forwarding attack in WSN. Under this scheme some node on the route to base station will be designated as checkpoint that will send the acknowledgement of the packet received to

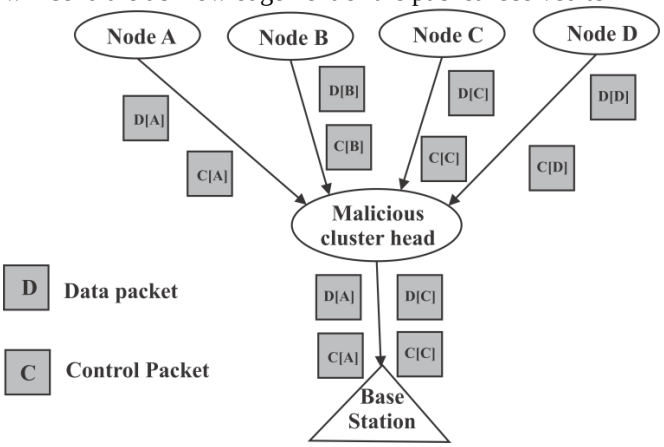

Figure 1: Greyhole attack based on the source of the packet

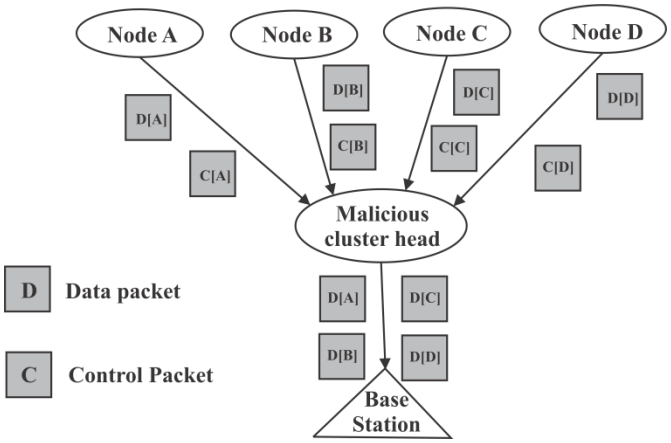

Figure 2: Greyhole attack based on the type of the packet

sender. If any checkpoint does not receive the acknowledgement in given time it will suspect the attack and send the alert message for same

Krontiris [5] has designed the Intrusion Detection Scheme (IDS) to detect the blackhole and greyhole attack. Author has used the watchdog approach [6] to monitor the behavior of the node in network. IDS have used the combination of specification based approach and cooperative decision making in order to detect the attack.

Reddy [7] has proposed the theory to detect the selective forwarding attack using two players zero-sum game. Here the game is played between the intrusion detection system and attacker node. Payoff will decide presence of attack.

Sultan [8] has used provenance to find the node that is implementing the selective forwarding attack. The author has used the inter-packed delay for detection of the greyhole attack.

Cho [9] has discussed the scheme to detect the packet drop attack using trust mechanism. It has used traditional beta trust model and entropy trust model with modification to estimate the trust value of its neighboring node

In the above mention literature it is being observed that this protocol require participation of more than one node for detection of attack which can increase the energy consumption of the node and reduce the throughput. Besides this the above mention detection mechanism is not completely foolproof, for instance in the scheme mention by Cho [9] if attacker gets to know the threshold value use to detect the attack, then attacker can manipulate its attack so that the trust value is less then threshold and attacker goes undetected. In this paper we have try to address the above mention two drawbacks.

4. Proposed System

In this section we will discuss our proposed scheme to detect and mitigate the greyhole attack using trust mechanism. We will first describe the network model and see how adversaries can launch the attack and then discussed the how to calculate the trust value in order to detect the attack.

4.1 Network model

Network consists of typical wireless sensors nodes who have limited battery life. These nodes are deployed to monitor the environmental parameters in its neighborhood. Parameters to monitor are decided based on the application for which network is deployed. Sensor node monitors the environment and report the change in the environment to base station. Once deployed this node are unattended hence vulnerable to malicious activity. The network of the node is organized using the LEACH [10] protocol.

Base station is the root node or sink node. All nodes in the network can reach base station. Unlike sensor node base station have unlimited battery life and cannot be compromised by any adversary. A typical network model is demonstrated using figure 3.

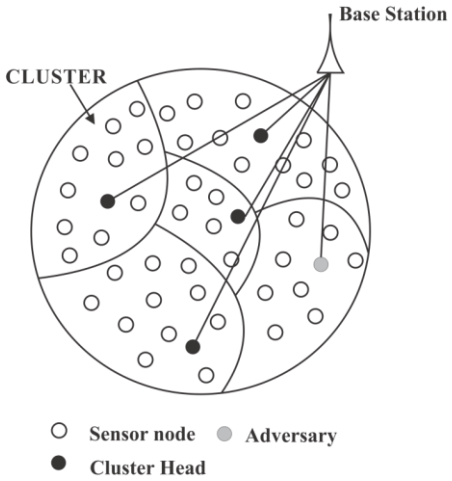

Figure 3: Network model

4.2 Adversary Model

The adversary cannot compromise the base station. But an adversary can compromise the cluster head to launch the greyhole attack. Once the cluster head is compromised then it can drop any number of packets it wants.

The basic assumptions made while developing the detection mechanism are as follow. First it is assumed that there can be only one adversary present in the network or a cluster. That is there are no colluding adversaries present in the network. At present there is single level of hierarchy in network. That is the hierarchy will consist of base station – cluster head – node. The network is congestion free because of the LEACH protocol. Every node will sign its message using it own private key and send this digital signature along with original message. This will prevent any packet manipulation attack that an adversary is implementing.

4.3 Detection Technique

In this section we have provided the overview of the proposed scheme to detect the malicious node that is implementing the greyhole attack. As mention before we have used the trust mechanism to detect the malicious node. The method to calculate the trust value of the node is proposed by Yao [11], Bao [12], Shaikh [13]. In our proposed detection mechanism we have used the concept of Bao [12] with some modification as per our problem. Thus our proposed detection mechanism is as follow.

Every node in cluster will calculates the trust value for the cluster head. As mention in [12] the trust value can be calculated by considering the two components of trust i.e. *social trust* and *QoS trust*. *Social trust* includes the parameter like intimacy, honesty, privacy etc. Where else *QoS trust* include parameter like cooperativeness, reliability, energy consumption, task completion etc. In our proposed scheme we have used *energy consumption* and *task completion* as the parameter to calculate the trust value. The detection mechanism is as follow.

The first task in the detection mechanism is to calculate the trust value for the cluster head. Therefore every node in cluster will calculate the trust value for the cluster head of its cluster. To calculate the trust value every node will monitor the behavior of the cluster head. And then calculate the trust value using the data collected by monitoring the behavior of the node. The trust value calculate by each node is independent of each other. That is trust value calculated by certain node in cluster will not influence the trust value calculated by other node in same cluster. Trust values are calculated at two level, node level and cluster level. At node level, node calculate the trust value of cluster head for each node in cluster separately and at cluster level, trust value calculate is the aggregate trust value of cluster head for the cluster. Cluster level trust value is calculated for the following two reasons; one is to have the aggregated view about the performance of cluster head. And other is to have this trust value as the backup for the node level trust value, i.e. if the attack goes undetected then it can be detected at cluster level.

Further two subsections we will specify how to calculate the trust value of cluster head with respect to task completion and energy consumption. Table 2 specify the symbol used through the paper.

Table 2: List of Symbol

Symbol	Description
CH	Cluster Head
N	Set of nodes in the cluster under consideration
M	Set of nodes in network
BS	Base station
Tc	Task Completion
Ec	Energy Consumption
$T_{tc}^{i}(j)$	Trust value of cluster head calculated by node j for node i on bases of task completion, where i, j belong to same cluster.
U_i	Number of packet transmitted by node i to CH
F_i	Number of packet belonging to i forwarded by CH
θ_i	amount of energy consumption of CH estimated by node j
$T_{ec}(j)$	Trust value of cluster head calculated by node j on the bases of energy consumption
$E_1(j)$	The residual energy of CH estimated by node j when the cluster is form.
$E_2(j)$	The residual energy of CH estimated by node j just before the new cluster will be form
$T^i(j)$	Complete trust value of CH calculated by node j for node i.

W_{tc}	The weight assign for task completion.
W_{ec}	The weight assign to energy consumption.
T(j)	The cluster level trust value calculated by node j
TH	Predetermine threshold value for the node to generate the alert message to base station
s_i	Status of the node
$T^i(BS)$	The node level trust value of CH calculated by base station for node i
T(BS)	The cluster level trust value of CH calculated by base station.

4.3.1 Calculating the Trust Value Based on Task Completion

Task of the cluster head is to forward the packet send by the node in its cluster. Due to the presence of adversary cluster head may drop some packed instead of forwarding. Thus by finding out how many packet the cluster head has forwarded we can find out how much task the cluster head has completed. To calculate the trust value with respect to task completion every node will continue to listen to its environment. Every node will keep the count of number of packet that other nodes in the cluster has transmitted to cluster head. In wireless sensor network all the packets are actually broadcasted therefore it is easy for all other node to listen the packet transmitted to cluster head. Beside this every node will observe the number of packet forwarded by the cluster head. Let U_i be the number of the packet transmitted by node i to CH, and F_i be the number of packet belonging to node i forwarded by CH to BS. Therefore the trust value of CH calculated by node j for node i on bases of task completion is given by following equation.

$$T_{tc}^{i} = \frac{F_i}{U_i} * 100 \qquad (1)$$

If the cluster head is not compromised then it will forward almost all the packet transmitted by the node in its cluster and because of this the trust value will be near 100. But if the cluster head is malicious and is maliciously dropping the packet then the trust value will start decreasing. Thus lower value of trust value will indicate that packet are being drop by the cluster head.

4.3.2. Calculating the Trust Value Based on Energy Consumption

The amount of energy required by CH is proportional to number of packets forwarded by it BS. Some energy is also required to process the packet but is quite less than the energy required to send the packet over the medium. If the node is implementing the greyhole attack then the energy required is less than the energy required by the normal node. And this difference in the amount of energy actually consumed by cluster head is used to calculate the trust value based on energy consumption. This trust value is calculated as follow.

After listening to the total number of packet send to CH by all nodes in cluster, node *i* will estimate the amount of energy that CH will require to forward all packet to BS. Let the amount of energy consumption of CH estimated by node j be θ_j. To calculate the actual energy consumption every node will find out the residual energy of CH when the cluster is form and residual energy just before the new cluster will be form. Let $E_1(j)$ be the residual energy of CH estimated by node j when the cluster is form. And let $E_2(j)$ be the residual energy of CH estimated by node j just before the new cluster will be form. The residual energy can be found out using method mention in [14]. The trust value based on energy consumption is given by following equation.

$$T_{ec}(j) = \frac{\theta_j - |E_2(j) - E_1(j)|}{\theta_j} * 100 \qquad (2)$$

Once the node has calculated the trust value based on both the component it will then combine both the values to calculate the total trust value.

4.3.3 Calculating the Total Trust Value

Once all nodes have calculated the trust value based on task completion and energy consumption we now combine this value to form the complete trust value. While combining the two trust value we will assign the weight to both the trust value. The total trust value is calculated as follow.

$$T^i(j) = W_{tc} * T_{tc}^i(j) + W_{ec} * T_{ec}(j) \qquad (3)$$

Where $W_{tc} + W_{ec} = 1$

In the above equation $T^i(j)$ is the complete trust value of CH calculated by node *j* for node *i*. W_{tc} is the weight assign for task completion and W_{ec} is the weight assign to energy consumption. The values to these weights are decided by network administrator.

Once the trust value is calculated at node level we will now calculate the trust value at cluster level. To calculate the trust value at cluster level we find out the average of all the node level trust value. Let N be the set of nodes belong cluster with cluster head CH. Let *T(j)* be the cluster level trust value calculated by node *j*. Then the average is calculated as follow.

$$T(j) = \frac{\sum_{i \in N} T^i(j)}{|N|} \qquad (4)$$

4.3.4 Generating the Alert Message

Once the node has calculated the trust value of its cluster head, node will now determine whether the cluster head is malicious or not. For this node will compare its calculated trust value with the predetermine threshold TH. Every node in cluster will compare both its node level trust values and the cluster level trust value with threshold.

If any node discovers the trust value below *TH* then that respective node will raise alert message claiming that the cluster head may be malicious node implementing the greyhole attack and node will directly broadcast to BS. Alert message will include ID of the node who have generated the message, Cluster head ID, array of trust values calculated by the node, and message signature to authenticate the message.

Alert message = {node ID||CH ID||trust values||Digital Signature}

Nodes may required to store their calculated trust value for certain time period

4.3.5 Detecting the Adversary at Base Station

Once the base station receive the alert message from node, base station has to verify whether the cluster head is malicious or not as claimed in alert message received. For this base station will maintain two data structure, one array to keep the status of the node in the network and other array to keep the aggregated trust value found out in past before the arrival of the alert message. The status of the node indicates whether the node was detected malicious or uncertain or normal. Uncertainty arises when malicious node try to increase the trust value by reducing the number of packet it is dropping, so that it goes undetected. Initially when the network is setup the status of all the node is initialized as normal. Status of the node is assigned the value as 0 for malicious, 0.5 for uncertain node and 1 for normal node. Status array is represented as follow.

$S = \{ s_1, s_2, s_3, \ldots\ldots, s_{|M|} \}$

$$\text{Where } s_i = \begin{cases} 0 & \text{if node i is malicious} \\ 0.5 & \text{if node i is uncertain} \\ 1 & \text{if nide i is normal} \end{cases}$$

The trust value array is represented as follow

$T = \{T_1, T_2, T_3, \ldots\ldots, T_{|M|}\}$

When the base station receive the alert message from any node, first it will find out the ID the sender of the message and ID of the cluster head of the cluster it belong to. It may happen that the node sending the alert message is the malicious node and it is trying to bad mouth the cluster head to reduce the reputation of the cluster head. So to overcome this bad mouthing attack, BS will broadcast the request message to all the nodes under the cluster head *CH* to send the trust value calculated by them for the *CH* of the cluster they belong to. Upon receiving the request from the base station all the node those were under the cluster head *CH* will send all the trust values calculated by them to *BS*. The reply message will contain the node ID and the array containing the trust value calculated by node.

When the base station receives the reply from all the other nodes in the cluster it will then combine the trust value it has received including the trust value from the node which has generated the alert message. When the trust values are combined, the trust value is weighted by the status value of the node. The equation to combine is given as follow.

$$T^i(BS) = \frac{\sum_{j \in N}(s_j * T^i(j))}{|N|} \qquad (5)$$

In the above equation $T^i(BS)$ is the node level trust value of cluster head calculated by base station for node *i*. The following procedure is applied for all $j \in N$. After calculating the trust value at node level *BS* will calculate the trust value at cluster level. The equation to calculate the trust value at cluster level is as follow.

$$T(BS) = \frac{\sum_{j \in N}(s_j * T(j))}{|N|} \qquad (6)$$

Where *T(BS)* is the cluster level trust value value of cluster head calculated by base station. Base station will then update its own list of trust value. Thus new value of $T_{CH} = T(BS)$.

After combining the trust value, base station will compare the trust value with the thresholds to determine whether the cluster head is normal, malicious, or uncertain, and it will then update the status of the node. For this we have two threshold value TH1, TH2 such that TH1<TH2. If the trust value is less then TH1 then the node is malicious. If the trust value is greater than or equal to TH1 but less then TH2 then there is uncertainty. And if the trust value is greater than or equal to TH2 then the node is normal. The base station will compare the thresholds with the trust value at both node level trust values and cluster level trust value. The above condition is represented as follow.

$$s_{CH} = \begin{cases} 0\,(malicious) & if\ T(BS) < TH1 \\ 0.5\,(uncertain) & if\ TH1 \leq T(BS) < TH2 \\ 1(normal) & if\ TH2 \leq T(BS) \end{cases} \qquad (7)$$

Above condition applies for node level trust values.

4.4 Mitigating the Attack

Once base station as determine whether the cluster head is malicious or uncertain, next is to solve uncertainty if it is there. If the node is malicious then base station will simply remove it and install the new node to replace it. But when there is uncertainty, BS has to verify whether the node is normal or malicious.

In order to solve uncertainty every node in the network will store the list of IDs of the packet it has forwarded to their cluster head. These IDs will be stored in chronological order. LEACH algorithm use TDMA to transmit the message to the cluster head. Cluster head accumulate this message and then forward this message to base station in the manner they have arrived. So whenever the uncertainty occur base station request the list of IDs of the packet that they have send to cluster head. Base station will then check the packed that it has received and mark it as “received”. And make the new list which will indicate the packet that it has received and that have been drop in chronological order in which they were send by sender.

After forming the list of packet dropped and the packed received, as shown in following diagram for illustration, the base station will initialize the window of particular size in the following diagram window size is of nine packets. Once this is done base station will calculated the percentage of packet received within the window. Base station repeats this procedure by moving window by one position at time. Here the percentage will act as trust value. If there is malicious activity then there would be atleast one instant where the trust value is less than TH1.

The figure 4 explains above process. In the figure 4 diagram R stand for packet received and D stand for packet drop. If we observe the diagram then at the end of the attacker has stop dropping the packet to prevent itself from being detected. If we ignore the energy consumption then we would get the trust value as 63.63. Assuming our threshold is TH1=50, TH2=70 then we can see that this scenario lead to uncertainty. Assuming position of the window the percentage of packet dropped is 44.44 which is less than TH1. Therefore we can verify that the node is malicious. If the window size is small then the chance of making correct decision is increased.

Even after this if the uncertainty is not solved then base station will inform all nodes in the network about this uncertainty. Next time if the same node is selected as cluster head the other nodes in the network will not cooperate with that node for next few rounds.

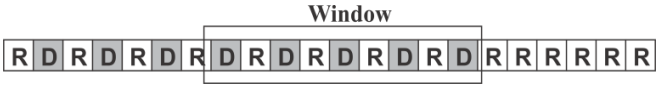

Figure 4: Example for detection using window

4.5 Updating the Threshold Values

Our major challenge while detecting the attack is that smart attacker will manipulate its attack in such a way that the trust value remains above the threshold, for this it is necessary to update the threshold values. The threshold value is updated as follow. Let M be the set of node in complete network.

$$TH1 = \frac{1}{2}\left(\frac{\sum_{i \in M} T_i}{|M|}\right) \quad (8)$$

$$TH2 = \frac{2}{3}\left(\frac{\sum_{i \in M} T_i}{|M|}\right) \quad (9)$$

Base station will broadcast TH2 to all nodes in network and this will new value of TH for nodes to generate the alert message.

5. Conclusion

In this paper we have proposed the method to detect the greyhole attack using the trust mechanism. Our proposed system does not require extra node to monitor the behavior of cluster head and to calculate the trust value. The attack detection is done by base station and nodes only calculate the trust value and forward it to base station. Thus energy required by the node for the attack detection will be reduced. We have also discussed situation where the attacker uses smart tricks to hide the attack in such a way that it remains undetected. We have also discussed the method to update the threshold value in order to increase the accuracy of detection mechanism.

This proposed scheme is designed with assumption that there will be only one adversary present at any point of time. Further scope of our report will be to design the detection mechanism when there are colluding adversaries.

References

[1] Youngho Cho, Gang Qu, Yuanming Wu. "Insider Threats against Trust Mechanism with Watchdog and Defending Approaches in Wireless Sensor Networks." IEEE Symposium on Security and Privacy Workshops, 2012, pp. 134-141.

[2] Harpal Singh, Vaibhav Pandey. "A Defence Scheme to Detect Selective Forwarding Attack in WSN." International Journal of Advanced Research in Computer Science and Software Engineering, Volume 4, Issue 8, August 2014

[3] Chris Karlof, David Wagner. "Secure routing in wireless sensor networks: attacks and countermeasures." Ad Hoc Networks, Volume 1, Issues 2–3, September 2003, Pages 293–315

[4] Bin Xiao, Bo Yu, and Chuanshan Gao. "CHEMAS: Identify suspect nodes in selective forwarding attacks." *J. Parallel Distrib. Comput.* 67, 11, November 2007, 1218-1230.

[5] Krontiris, I., Dimitriou, T., Freiling, F.C.: "Towards intrusion detection in wireless sensor networks." In: Proceedings of the 13th European Wireless Conference, Paris, France April 2007.

[6] Sergio Marti, T. J. Giuli, Kevin Lai, and Mary Baker. "Mitigating routing misbehavior in mobile ad hoc networks." In *Proceedings of the 6th annual international conference on Mobile computing and networking* (MobiCom 2000). ACM, New York, NY, USA, 255-265.

[7] Yenumula B. Reddy and S. Srivathsan. "Game theory model for selective forward attacks in wireless sensor networks." In *Proceedings of the 2009 17th Mediterranean Conference on Control and Automation* (MED '09). IEEE Computer Society, Washington, DC, USA, 458-463.

[8] Salmin Sultana, Elisa Bertino, and Mohamed Shehab. "A Provenance Based Mechanism to Identify Malicious Packet Dropping Adversaries in Sensor Networks." In *Proceedings of the 2011 31st International Conference on Distributed Computing Systems Workshops* (ICDCSW '11). IEEE Computer Society, Washington, DC, USA, 332-338.

[9] Youngho Cho, Gang Qu. "Detection and Prevention of Selective Forwarding-Based Denial-of-Service Attacks in WSNs." Hindawi Publishing Corporation International Journal of Distributed Sensor Networks Volume 2013, Article ID 205920, 16 pages

[10] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. "Energy-Efficient Communication Protocol for Wireless Microsensor Networks." In *Proceedings of the 33rd Hawaii Internatio nal Conference on System Sciences-Volume 8 - Volume 8* (HICSS '00), Vol. 8. IEEE Computer Society, Washington, DC, USA, 8020-8030.

[11] Yao, Z., Kim, D. and Doh, Y. "PLUS: Parameterized and Localized Trust Management Scheme for Sensor Networks Security." 2006 IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS), Vancouver, October 2006, 437-446.

[12] Fenye Bao, Ing-Ray Chen, MoonJeong Chang, and Jin-Hee Cho. "Hierarchical Trust Management for Wireless Sensor Networks and its Applications to Trust-Based Routing and Intrusion Detection", IEEE Transactions On Network And Service Management, Vol. 9, No. 2, June 2012, 169-183.
[13] Shaikh, R.A. ; Jameel, H. ; d'Auriol, B.J. ; Heejo Lee ; Sungyoung Lee ; Young-Jae Song. "Group-Based Trust Management Scheme for Clustered Wireless Sensor Networks." IEEE Transactions On Parallel And Distributed Systems, Vol. 20, No. 11, November 2009, 1698 – 1712
[14] Y. J. Zhao, R. Govindan, and D. Estrin, "Residual energy scan for monitoring sensor networks," in Proc. 2002 IEEE Wireless Commun. Netw. Conf., pp. 356–362.