International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2015): 6.391

Gray Hole Detection and Removal in MANET by Pool Tile Method

Supriya Pustake[1], Dr. S. J. Wagh[2], D. C. Mehetre[3]

[1]Department of Computer Engineering, K J College of Engineering and Management Research, Pune, India

[2,3]Professor, Department of Computer Engineering, K J College of Engineering and Management Research, Pune, India

Abstract: ***Nowadays Mobile Ad-hoc networks are widely in use. Such networks are infrastructure less which invites multiple threats to the network. As a result it decreases the performance level of network. Among these threats gray hole is the silent but more vulnerable attack which keeps attacking the network in the repetition mode. Many techniques invented to detect and remove these gray holes, where nodes in the pool themselves need to work together to find out the malicious node. If the neighbor nodes of the gray hole node itself is malicious or it is not functioning due to some other technical reasons then it is hard to detect the gray hole node in the network pool. So a pool tile method is introduced where tile represent the time "t" in which a pool manager of the pool performs iterations in "t" to keep the record of routing history and to decide gray hole node.***

Keywords: Gray hole, network pool, pool manager, tile, request, response, routing, MANET

1. Introduction

Mobile ad-hoc network (MANET) is formed by mobile devices by connecting through wireless media. MANET suits for military operations and the emergency rescue that need to perform special purpose in urgency. As MANET doesn't require any infrastructure, it can be installed anytime anywhere. Because of its easy installation features, it is easy to use in personal area networks and so on.

However due to the dynamic network topology, no infrastructure property, there is high risk of security in MANET [1]. DSR (Dynamic Source Routing) protocol is an on-demand routing protocol. It uses source routing, instead of using routing table at every hop. Its operation is divided into two parts, route discovery and route maintenance.

Here whenever any node has a data to transfer, it first checks its route cache for the route to destination node. If path is available in route cache then it will directly transfer the data over the path. But if route is not available in route cache then it initiates the Route Discovery process. Node who initiates the process becomes the source node. Now the source node will broadcast the Route Request (RREQ) packet over the network. It contains the destination node and the unique identifier from the source.

Every node who receives RREQ checks for the unique identifier, if it has already seen then request is rejected. Otherwise it appends its own address to the list and rebroadcast the request over the network.

When RREQ is received by the destination node it will unicast the Route Reply (RREP) back to the source node, and it appends the list of addresses received from route request. When RREP is received by source node it updates it's cache with a new route.

Route Maintenance helps to inform the source node regarding the unavailability of route, link breakage etc. In such situation route error (RERR) message is transferred to notify the unavailability of path [14].

While performing the inter node communication many nodes are pretending to be the honest node and receives the data and drops at their end. This actually creates hole where all the data packets come and sinks and won't be available to the further nodes for the communication. This type of sinking data packets in MANET is known as black hole attack. Whereas the gray hole attack is periodically keep dropping the packet data and makes even more difficult to identify the gray hole attack. Gray hole may drop data for particular IP, while transfer for others in the network. It selectively drops the data which makes it difficult to identify [2].

So many systems are proposed to handle the gray hole attack in which all the actions are need to be taken by the nodes themselves to detect gray hole. This actually affects the routing speed and performance of the network itself. So we come out with an idea in which we are creating a pool which actually contains many number of mobile nodes and this pool also contains a pool manager which actually a system which identifies the gray hole node and removes it too.

The remaining paper is organized as follows. Section 2 discusses some related work and section 3 presents the design of our approach. The details of the results on this approach are discussed in section 4 as Results and Discussions. Sections 5 provide hints of some extension of our approach as future work and conclusion.

2. Related Work

Number of methods are discovered to detect gray hole node in the network. Some of those work during route discovery process while others at the time of data transmission. Those who work during route discovery are more advantageous, because they don't lose the data. While on the other hand, how much we can try there should be minimum of data loss

during detection of the gray hole at the time of data transmission.

The method proposed by S D Khatawkar [1], makes use of mobile agents (MA) to detect gray hole using the code migration facility. MA consists of program code and program execution state. Here the performance decreases with random mobility and also with mobility of nodes, it has some false detection.

The method proposed by N. Dharini [2], uses light weight learning based energy prediction algorithm. By comparing consumed energy with predicted energy, gray hole is detected. Less energy consumption means node has not transmitted data. Proposed method achieves energy saving so as it increases network lifetime.

While the method proposed by Parineet D. Shukla [3] works by using the probability for dropping the packets and getting a false reply from the next node, the gray hole can be detected. Probability for getting a false reply from the node, act as a threshold value for deciding the behavior of a node.

The method proposed by Seemita Pal [4], detects gray hole by observing delay in packet arrival by calculating slope of the delay over a given window. Based on difference in slope after a packet loss and the slope of the next coming packet, it decides the reason behind the packet loss. This method exploits the correlation between packet delays and packet loss due to congestion.

The main idea behind the method proposed by Qiang Liu [5] is, it combines downstream assessment and end-to-end assessment to detect gray hole attack. Method uses fast hashing and digital signature techniques to protect packet against manipulation, replay and masquerading attacks at mesh routers. Method minimizes false positive and false negative rates.

The method proposed by Jiwen CAI [6] deals with network layer and MAC layer. Here they focus on the path of transmission to detect a gray hole by observing the next hop's actions not all neighbors. This increases system performance. But still there is a problem of false positive probability.

The method proposed by Devu Manikantan Shila [7], uses channel aware detection algorithm. It adopts two strategies for detection, hop-by-hop loss observation by downstream nodes and traffic monitoring by upstream nodes. Here control packets are more which causes overhead in the network.

The scheme proposed by Jaydip Sen [8], first collects the data routing information in a routing table. Then they detect the presence of a gray hole locally. But sometimes there might be chances of declaring an honest node as malicious node. So to avoid the chances of false positive it is once again checked by the nodes in the network cooperatively. If here node declared as a gray hole, then alarm is sent over the network, which informs all the nodes in the network about the gray hole. In this way they take care that gray hole should be separated from the network.

The method proposed by Gao Xiaopeng Chen Wei [9], use aggregation signature algorithm to produce evidence on forwarded packets, these evidences helps to detect malicious node. It uses three algorithms, the creating proof algorithm which creates the proof based on aggregation signature. The checkup algorithm is invoked if source suspects some malicious activity. Finally the diagnosis algorithm detects the gray hole from the evidences provided by checkup algorithm.

3. Proposed Work

This section narrates the thorough understanding of the proposed system which is depicted in the figure 1 and 2.
Proposed system is properly elaborated with the below mentioned steps.

*Step 1:*For better connectivity and flow, all the nodes have been connected to a pool which is powered by the wireless router. For our experiment a two antenna Digisol router is used for the creation of the pool with its head as pool manager as shown in the figure 2.

*Step 2 :*For easy understanding let's consider that node 1 wants to transfer the data to the node 4 in the network. As soon as node 1 selects node 4 as its destination a shortest path calculation job will be conducted by the pool manager after receiving Route request from the source node (RREQ), that is node 1.

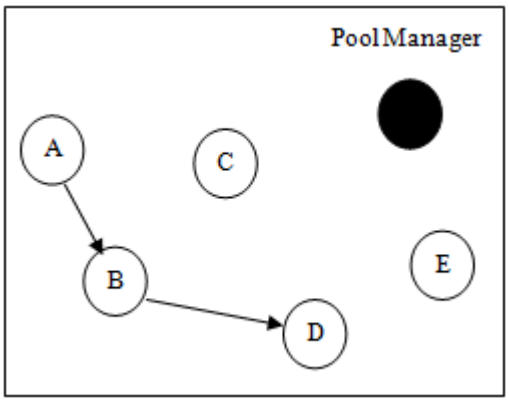

Figure 1: Basic Idea of proposed system

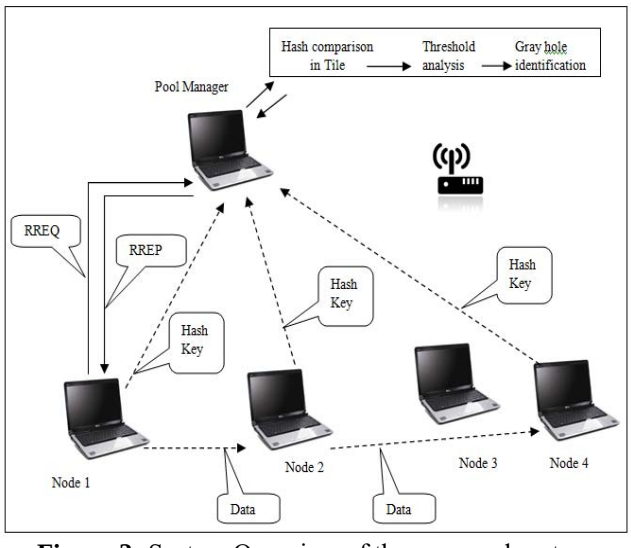

Figure 2: System Overview of the proposed system

Pool manager takes the input as the source and destination nodes and identify the possible paths for routing. Once the possible paths of routing is been identified then the time delay is calculated for all the identified paths. So the path which eventully yields the least time delay for identified nodes is considered as the shortest path. This shortest path will be returned to the request source node as Route reply (RREP).In our case of narration the path will be Node 1-> Node 2-> Node 4.

*Step 3:*Here in this step on receiving of the shortest path from the pool manager source node starts routing the data to the next hop. While on this process it does two importatnt things,

Firstly it sends the hash key of the data which is been generated by the MD5 Algorithm to the pool manager.

Secondly it transfer the original data to the hop node , in our case it is node 2.

*Step 4:*Here in this step the hop node is characterized by the concept of sending the hash key of the received data generated by the MD5 Algorithm to the pool manager.As soon as pool manager receives both the hash keys for the verification, it checks the data authenticity on the said Tile.

Then by comparing both the hash keys from the source node and the destination node of the instance, pool manager will check for the avalanche effect of the hash keys. The avalanche effect will be created if any slight drop or malfunctioning of the data is happened at the instance destination node.

If any effect is identified then this is considered as the gray hole and then pool manager will check for its threshold of 2. In our experiments the threshold of 2 is set due to two hops that encountered in the process of data transimisison.

Once any node is identified as the gray hole on considering of its threshold then this node will be black listed in the database for any of it's role in the future routing process. And this will abort the routing process of the data by rising proper alert .

*Step 5:*If in any case hash keys are not found for any kind of avalanche effect this indicates no drop of data or any malfunctioning of the data at the Destination node.

And now destination node is characterized to become source node for the instance and transfer the data to the next hop, in our case path will be from node 2 to node 4.

The complete process can be depicted in the below Algorithm

Algorithm : Gray Hole Identification Using Pool Tile Method

// Input : Sender Data **D**
// Destination Node $\mathbf{D_n}$
// Output : Successful identification of Gray hole node
Step 0: **Start**
Step 1: Add Pool managers $\mathbf{M_1}$ into pool $\mathbf{P_1}$, and so on $\mathbf{M_n}$ to $\mathbf{P_n}$
Step 2: Add node $\mathbf{N_n}$ to pool $\mathbf{P_n}$
Step 3: Set Tile **T** in all Pool managers $\mathbf{M_n}$(Where tile is Time in Seconds)
Step 4: Activate all pool managers $\mathbf{M_n}$
Step 5: Select Data **D** by source node $\mathbf{S_n}$
Step 6: Select Destination node $\mathbf{D_n}$
Step 7: Identify the shortest path $\mathbf{P_{th}}$
Step 8: **WHILE** $\mathbf{D \notin D_n}$
Step 9: for each tile T
Step 10: $\mathbf{P_d \rightarrow S_{nt} \in D}$ ($\mathbf{P_d}$= previous data in Hash and $\mathbf{S_{nt}}$= Run time Source node)
Step 11: $\mathbf{C_d \rightarrow D_{nt} \in D}$ ($\mathbf{C_d}$ = Current data in hash and $\mathbf{D_{nt}}$ = Run time Destination node)
Step 12: check **IF** $\mathbf{C_d \neq P_d}$ (Avalanche Effect)
Step 13: Identify the Node for $\mathbf{D_{nt}}$
Step 14: Label it as Gray hole $\mathbf{G_h}$
Step 15: Vote all the Nodes in the pool $\mathbf{P_n}$ for Gray Hole $\mathbf{G_h}$
Step 16: **END IF**
Step 17: **END WHILE**
Step 18: **Stop**

---

4. Results and Discussions

For the experimental process of the system 5 java based windows machines are taken with Netbeans as the development IDE. System is put under hammer for number of tests for the proper identification of the gray holes in many scenarios.

Experiment is conducted on the performance time to identify the gray holes with other system proposed by [1]. In [1] gray holes are been identified in the mobile network based on the cluster analysis process. When the system incorporates the method stated in [1] for the sake of comparison then the performance time noted and shown in the below table.

Table 1: Gray Hole identification performance in Time

Pool Tile Method (Milli second)	Cluster Analysis Method (Milli second)
64	497
52	398
52	405
68	412

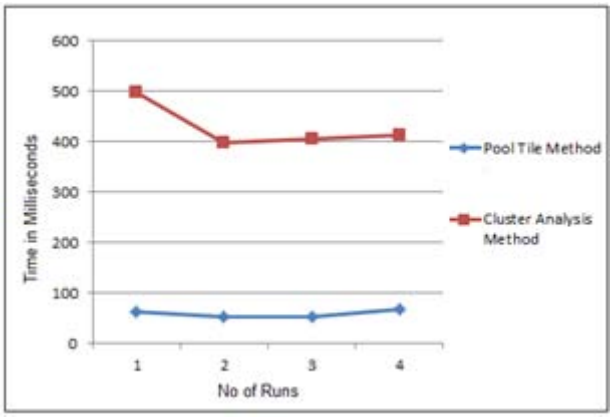

Figure 3: Gray Hole identification comparison in time

When the values of the table 1 are drawn in a plot as shown in figure 3 we found the fact that system proposed by using pool tile method excels then that of the system proposed through the Cluster analysis process. Time taken by the pool tile method is less compared to that of [1]. This clearly indicates that the decision of identifying gray holes is centric and takes off the burden on the routing nodes, this eventually fastens the system.

5. Conclusion and Future Work

The proposed system successfully identifies the gray holes in the mobile ad-hoc network in a comparatively less time using pool tile method. System effectively makes use of pool managers to identify the gray hole. Pool tile method properly keeps checking the data dropping at every node on each hop and thereby detects and removes the gray hole node and intimate all the nodes in the pool about the gray hole successfully.

As the future scope to our proposed methodology we can enhance our system for huge hybrid architecture of the internet of things with many hierarchies of pool managers.

References

[1] S D Khatawkar, NitinTrivedi, "Detection of Gray hole in MANET through Cluster Analysis" IEEE 2015 2nd InternationalConference on Computingfor SustainableGlobal Development(INDIACom), pp.1752-1757

[2] N. Dharini, RanjithBalakrishnan and A. PravinRenold,"Distributed Detection of Flooding and Gray Hole Attacks in Wireless Sensor Network" IEEE 2015 International Conference on Smart Technologies and Management for Computing, Communication, Controls, Energy and Materials (ICSTM), pp.178-184.

[3] Parineet D. Shukla, Ashok M. Kanthe,Dina Simunic, "An Analytical Approach for Detection of Gray Hole Attack in Mobile Ad-hoc Network (MANET)" 2014 IEEE International Conference on Computational Intelligence and Computing Research

[4] Seemita Pal, Huijiang Li, BiplabSikdar and Joe Chow,"A Mechanism for Detecting Gray Hole Attacks on Synchrophasor Data" IEEE ICC 2014 - Selected Areas in Communications Symposium, pp.4131-4136

[5] Qiang Liu, Jianping Yin, Victor C. M. Leung, and ZhipingCai,"FADE: Forwarding Assessment Based Detection of Collaborative Grey Hole Attacks in WMNs" IEEE Transactions on Wireless Communications, Vol. 12, No. 10, October 2013, pp.5124-5137

[6] Jiwen CAI, Ping YI, Jialin CHEN, Zhiyang WANG, Ning LIU,"An Adaptive Approach to Detecting Black and Gray Hole Attacks in Ad Hoc Network" 2010 24th IEEE International Conference on Advanced Information Networking and Applications, pp.775-780

[7] Devu Manikantan Shila, Yu Cheng∗ and Tricha Anjali,"Channel-Aware Detection of Gray Hole Attacks in Wireless Mesh Networks" IEEE "GLOBECOM" 2009.

[8] Jaydip Sen, M. Girish Chandra, Harihara S.G., Harish Reddy, P. Balamuralidhar,"A Mechanism for Detection of Gray Hole Attack in Mobile Ad Hoc Networks" ICICS 2007 IEEE.

[9] Gao Xiaopeng Chen Wei,"A Novel Gray Hole Attack Detection Scheme for Mobile Ad-Hoc Networks" 2007 IFIP International Conference on Network and Parallel Computing – Workshops IEEE, pp.209-214

[10] R. H. Jhaveri, S. J. Patel, and D. C. Jinwala, "A novel approach for grayhole and blackhole attacks in mobile ad hoc networks," in Advanced Computing & Communication Technologies (ACCT), 2012 Second International Conference on. IEEE, 2012, pp. 556–560.

[11] J. Sen, M. G. Chandra, S. Harihara, H. Reddy, and P. Balamuralidhar, "A mechanism for detection of gray hole attack in mobile ad hoc networks," in Information, Communications & Signal Processing, 2007 6th International Conference on. IEEE, 2007, pp. 1–5.

[12] S. Banerjee, "Detection/removal of cooperative black and gray hole attack in mobile ad-hoc networks," in proceedings of the world congress on engineering and computer science, 2008, pp. 22–24.

[13] O. V. Chandure and V. Gaikwad, "Detection & prevention of gray hole attack in mobile ad-hoc network using aodv routing protocol," International Journal of Computer Applications, vol. 41, no. 5, 2012.

[14] D. G. Kariya, A. B. Kathole, and S. R. Heda, "Detecting black and gray hole attacks in mobile ad hoc network using an adaptive method," international journal of emerging technology and advanced engineering, vol. 2, no. 1, 2012.