ISSN:0975-9646

Precheck Algorithm with Success Rate Table in Handshake for Detecting and Removing Gray and Black Hole Attack in MANET

K. R. Viswa Jhananie [1] , Dr. C. Chandrasekar [2]

[1]*Department of Computer Science,*
Sheshadripuram Academy of Business Studies,
Bangalore, Karnataka, India.

[2]*Department of computer science,*
Periyar University,
Salem, Tamil Nadu, India.

Abstract - **An Ad-hoc network is a collection of nodes that do not rely on predefined infrastructure to keep the network connected. Nodes help each other in conveying information about the topology of network and share responsibility of managing the network. Thus in addition to acting as hosts, each mobile node does the function of routing and relying messages for other mobile nodes. Since MANETs are mobile in nature, they are susceptible to various attacks like black hole and gray hole, etc. They will either lead to loss of data or lead to the wrong route from source node. In this paper, we propose an algorithm for detecting malicious node[s] that misguide the source node and remove it from routing table.**

Keywords - **Black hole attack, MANET, success rate table, malicious node.**

I. Introduction

MANET is a self configuring network of mobile nodes connected by wireless links to form an arbitrary topology. The nodes are free to move randomly. Thus the network topology is unpredictable and may change rapidly. MANETs are suitable for military, natural disasters, emergency medical situation because of their quick deployment, mobility, minimal configuration and no centralized management. MANETs are vulnerable to many attacks, since the nodes themselves act as routers for finding a route to the destination.

In this paper, We propose an acknowledgement based approach to detect and remove malicious node that attacks in the form of black hole and gray hole. In this, we introduce a new table called success rate table to track the success rate of each node in the route. The source node broadcasts the data count to the destination and all the nodes in the route, before it starts forwarding the data to the destination node. Any node in the route can update the table if the data count is less than the expected (failure case). Also broadcast the same to the source node. Source node decides to initiate malicious node detection and removal process with the support of success rate table.

The detection and removal process will be initiated by the source node. Our algorithm takes T(n) time on average to find the chain of malicious nodes. Black hole attack either advertises itself as having a valid route to the destination node or the malicious node consumes the intercepted packets. A variation of black hole attack is the gray hole attack, in which the data packets are dropped selectively. In this the malicious node behaves like a normal node and suddenly turns malicious and start dropping data packets.

In this paper, We propose an algorithm that detects and removes the black hole or gray hole attack in MANET. The literature review is done in section II. In section III, black hole and gray hole attacks are discussed. In section IV, the assumptions are given. In section V, the methodology is discussed. Section VI gives the conclusion and future work.

II. Literature Review

In [1], Geetha et al has used AOMDV – Ad-hoc On Demand Multipath Distance Vector to improve the security of MANETs against black hole attack. In [2], Kamatchi et al has used a polynomial to prevent MANET from black hole attack. In [3], Naseera et al has described black hole attack that can be mounted against MANET and proposed a solution for it in AODV protocol. In [4], Neelam et al has detected a single black hole and co-operative black hole attack using AODV protocol that can detect black hole even when the nodes are idle. A frame for route error is maintained that contains unreachable destination IP address and unreachable destination sequence numbers.

In [5], Pratiba Bhat et al has proposed a new algorithm to detect and remove black and gray hole attack. In [6], Shalini Jain et al has used an algorithm to detect and remove malicious node by dividing the total traffic in to some small sized blocks. End to end checking is done by sending and receiving messages from both source and destination nodes. In [8], Sherril Sophie Maria Vincent et al has used channel adaptive version of AOMDV routing protocol that uses specific channel quality information for path availability.

In [9], Vipan Chand Sharma et al has given a solution by modifying the working of a source node with request reply and waiting time to the data structures in the AODV protocol. In [10], Vishnu et al has established a backbone network of trusted nodes over ad-hoc network. When a node wants to transmit a data, it sends IP address along with RREQ to the destination node. Based on RREP along with IP address, the source node identifies the black hole.

III. Black And Gray Hole Attacks

Since ad-hoc networks are dynamic and has autonomous topology with distributed environment, MANETs can change locations and configure itself wherever it is required. These characteristics constitute more challenges for security. MANETs suffer from various security attacks, like black hole attack, gray hole attack, worm hole attack, flooding attack, etc. In black hole attack, the malicious node waits for the neighbors to initiate route request data packet. As the node receives route request packet, it will immediately send a false route reply packet with a modified sequence number.

This makes the source node to assume that the source node is having a fresh route towards destination. Thus the other routes to the destination node is discarded by the source node and the source node starts transmission through the malicious node route. The malicious node thus gathers all the data packets and drops them. Thus the transmission between source node and destination node gets disconnected.

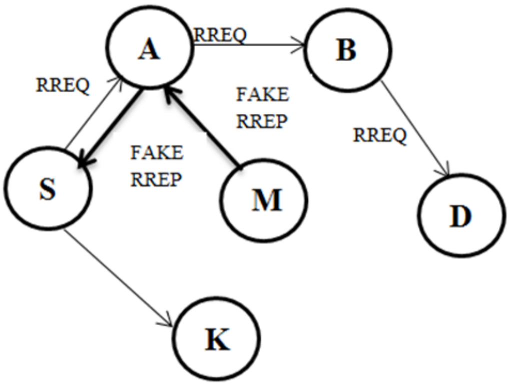

Fig 1 Formation of black hole attack in MANETs

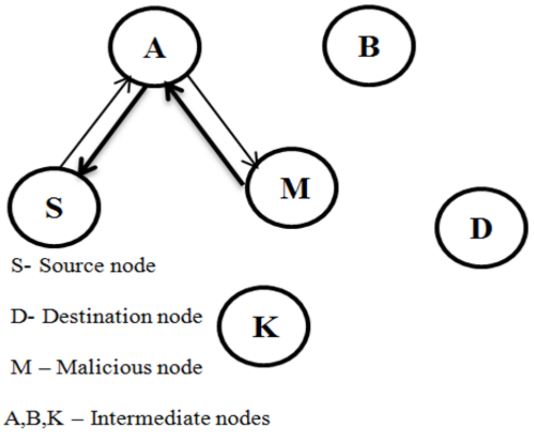

Fig 2 Nodes getting disconnected from source to destination

Sometimes, the malicious nodes attacks the nodes selectively, either by dropping the data packets for a particular destination, or dropping only the selected portion of the packets[7]. This is termed as gray hole attack and is difficult to detect. Since the malicious node drops over selective data packets, some traffic will continue on the network still which makes harder to identify the malicious node. This kind of attack is slightly different from black hole attack. In this attack, the malicious node actually interrupts the data packets in the route.

IV. Assumptions

In our approach, all the nodes in the route updates the success rate table for success or failure and broadcast the same to the source node. The source node updates the continuous column of success rate table if the data loss is continuous, which says that the nodes are 100% malicious. This is updated in continuous count column of the table accordingly. The intermediate node which is next to the data received node broadcasts the data count (assume dcount) to the source node.

In our algorithm, the source node will send a query message to detect malicious node only when it finds that the number of packets received by destination is significantly less than the number of packets actually sent by source node (say d_i). (ie) (dcount <= d_i). In malicious node detection process, when there is a data loss with the previous node id, the received node updates the success rate table as failure and thus the transmission is stopped.

Then the source node decides whether it needs to continue sending the data in the same route or not. The source node initiates the malicious node detection process with the help of success rate table, expiry response time (ertime), threshold of packets dropped and continuous column value. This process cannot be done straight forward, because the data loss in Mobile Ad-hoc network can happen due to packet overload or lack of CPU cycles.

Let us assume the threshold probability of non malicious packet drop by each node be P. When the source node checks whether (d_i (1-P) <= dcount), then it is not a malicious node. If the threshold probability of non malicious packet drop at source node is $\acute{p}$ and not P, then the source node will start gray/black hole removal process and also it checks for (dcount <= d_i (1- $\acute{p}$). This can be calculated from P as follows.

When the data loss in the initial node in the route is P, then the volume of data sent to the neighbor node is d_i (1-P). In the same way, the neighbor node data loss is P, then the next neighbor node sends d_i (1-P) (1-P) volume of data. Therefore, at the destination node, the total number of data loss due to malicious node is ($d_i – d_i (1-P)^n$, where ‘n’ is the total number of nodes in the route. Hence, $\acute{p} = 1- (1-P)^n$.

V. Methodology

The aim is to find a list of malicious nodes globally to all the nodes in the route through the success rate table. The behavior of each node in the route is monitored by next immediate node. Here, we divide the total traffic in to set of small data blocks. So, the malicious node can be captured in between communication of two such data blocks. The source node (S) sends a *precheck* message to all the nodes in the route, even to the destination node (D) before starting the communication. This *precheck* message makes all the nodes in the route alert about the incoming data packets.

The intermediate nodes and the destination node sends a timer for the end of incoming transmission and keeps a count of number of data packets received. Once the timer expires, the destination node sends a *postcheck* message to the source node. This message has the count of data packets

received by it. The source node, after sending the *precheck* message, broadcasts *verify* message to every next immediate nodes instructing them to check the action of its previous node in the route and start transmitting data.
Once the transmission is over, the source node sets a *timeover* for the receiving of *postcheck* message. If the source node receives the *postcheck* message before *timeover* expires and the number of data packets received by destination is same as the number of data packets sent by source or number of data lost is within the tolerable range, then the source node starts the transmission of the next data block. Otherwise, the source node starts the process of detection and removal of malicious node in the route or even when the source node receives data loss message from any of the intermediate nodes in the route.
Since the assumption is made as P for threshold data loss rate for each node and ṕ for total data loss rate, choosing the value of P is more important. It should be taken in to account that the lower value of P will detect any malicious node and also the total data loss rate should not be higher (not in unacceptable range > ṕ). We first assume the maximum value of ṕ depending on the hop count (ie) length of the path, then the value of P is calculated from ṕ.
Whenever the source node wants to find the gray/ black hole detection and removal process, it sends a query message to all the respective nodes and sets a *timeover* for the receipt of reply message from them. Therefore all the nodes in the route updates the success or failure rate in the table.

In this, the success case is when data sent and received are same (ie) dcount = = d_i or response is not within expiry response time (>ertime). The source and other nodes will increase the count of respective columns for success or failure. When the source node finds malicious node in any of the routes, then it updates *findattack* column of the routing table as true. Since, the data loss may happen due to resource limitation or lack of CPU cycles or buffer space or bandwidth, we deduct the malicious node when the data loss is continuous, using the success rate table. The continuous data loss leads to increase in failure rate. In this method, the malicious node is identified because the gray/ black hole attack will lead to continuous data loss or when the data loss exceeds the threshold, then it is identified as malicious node and remove it from routing table.

The status is updated as attacker node in the routing table and this is broadcasted to the entire network. Because of this, the algorithm suites best for gray/black hole attacks. In our method, we modify AODV protocol by introducing one more table maintained at each node. The success rate table is used to maintain the success rate of each node. We also modify the routing table of AODV by adding a new field called *findattack* which is set true when a malicious node is found in the route. Figure-3 shows the network scenario and each of the above tables are depicted below.

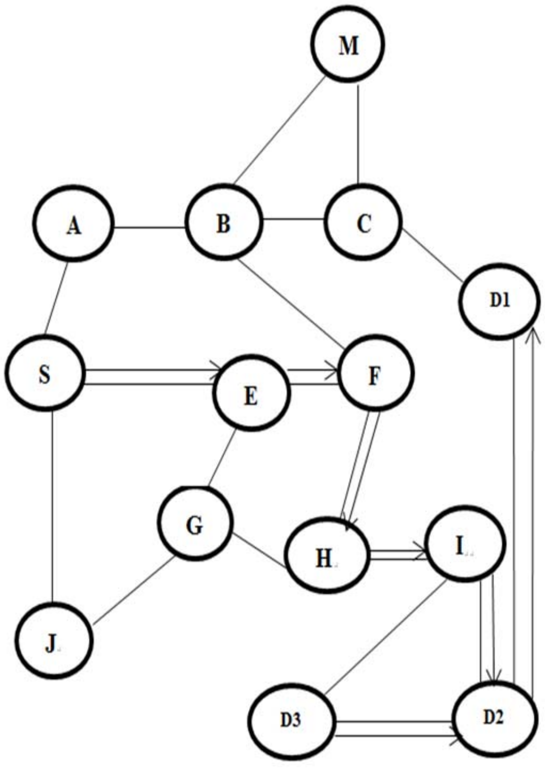

Fig 3 Network Scenario

S- Source node D1,D2,D3- Destination node
———Connections ——> Path

TABLE I
DATA ROUTING TABLE

Destination Node	Route	Find Attack
D1	E,F,H,I,D2	False
D2	A,B,F,H,I	False
D3	J,G,H,I	False

TABLE II
SUCCESS RATE TABLE

Route Id	Node Id	Failure Rate	Continuous Count
1	**E**	**1**	**0**
1	**F**	**0**	**0**
2	**H**	**4**	**4**

Algorithm for detecting Gray/ black hole attack:
Step -1: The data packets to be sent are divided in to
M equal parts.
Data[1,2,….M];
Initialize i=1;
Step-2: send *precheck* (S,D,d_i) message to the
destination node D,where d_i is the number of
data packets to be sent (current block).

```
Step-3: broadcast verify (S,D,PN) message to all the
        nodes, instructing neighbors of each node to
        monitor the previous node in the route(PN).
Step-4: forwarding of data packets starts from block
        data[i] to D.
Step-5: sets timeover, tv for the receipt of
        postcheck (D,S,dcount) message containing
        dcount, number of data packets received by D.
Step-6: if tv not expired and postcheck message
        received, (ie) if ($d_i$ (1-P) <= dcount)), go to
        step-8.
        else
        start gray/black hole removal process.
        (where ṕ is the threshold value ranging
        between 0 and 1)
Step-7: if tv expired and postcheck message not
        received then start gray/ black hole removal
        process.
Step-8: go to step-2 and continue when i<=M.
Step-9: stop source node, S action.
```

Action by Destination node:

```
Step-1: find the value of $d_i$, after receiving precheck
        (S,D,$d_i$) message.
        Initialize dcount = 0;
Step-2: sets timeover, tv for the receipt of current
        data sample and waits for data packets.

Step-3: When tv not expired, and data packet
        received, then update and send postcheck
        (D,S,dcount) message.
Step-4: when tv expired, send postcheck (D,S,dcount)
        message to S.
Step-5: stop destination, D action.
```

Action by Source node S:

```
Step-1: find the value of dcount after receiving
        precheck (S,D,$d_i$) message from D.
        initialize $d_i$ = dcount;
Step-2: compare data sent($d_i$) and data received
        (dcount)
        If (dcount == $d_i$)
            Update success table as success for all the
        nodes in the route.
        Else
            Start gray/black hole removal process.
Step-3: terminate source, S action.
```

Gray/ black hole removal process:

```
Step-1: get failure rate, continuous count of the node
        from success rate table.
Step-2: if failure rate> threshold and continuous
        count is true then remove node from routing
        table.
Step-3: broadcast to network.
Step-4: terminate the removal process.
```

Action by intermediate nodes:

```
Step-1: find the value of $d_i$ after receiving precheck
        (S,D,$d_i$) message from S.
        Initialize dcount = 0
Step-2: sets timeover, tv for the receipt of current data
        sample and waits for the data packets.
Step-3: when tv not expired then the data packet
        received is updated.
        If (dcount==$d_i$) then (where dcount is data
        received) forward data to next node.
        Else
          Update failure status in the table and send
        postcheck (D,S,dcount) message.
Step-4: when tv expired, send postcheck (D,S,dcount)
        message to S.
```

VI. Simulation Results

We have used NS-2.34 for our simulation. The network was constructed for the simulation purpose and then monitored for a number of parameters. The parameters used are given in the table.

Channel	Wireless
Propagation	Two ray ground
Network Type	Wireless
Traffic Source	CBR
Number of Nodes	60
Maximum Packets	50
Simulation Period	100msec

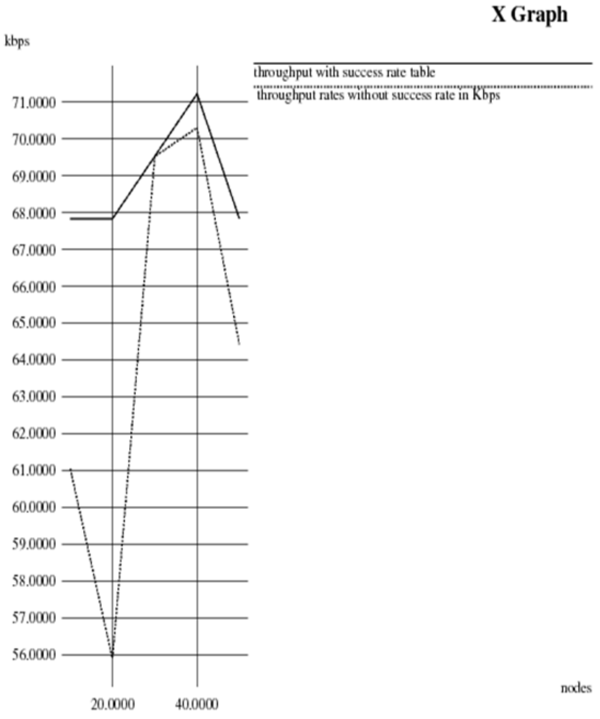

Fig 4 Throughput

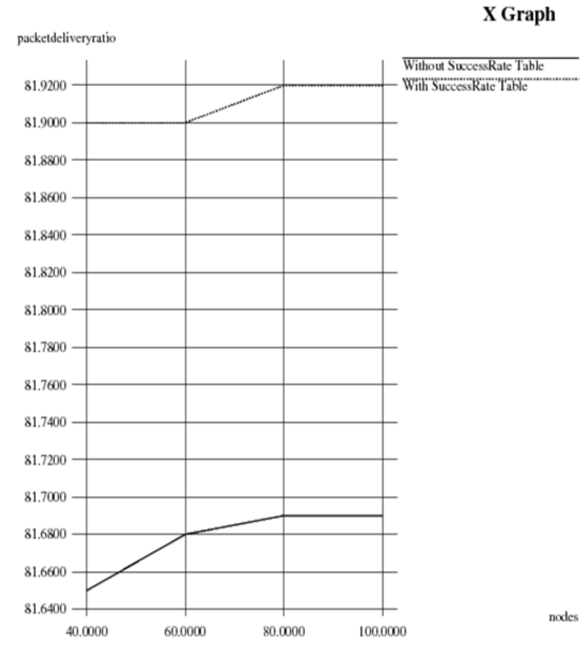

Fig 5 Packet Delivery Ratio

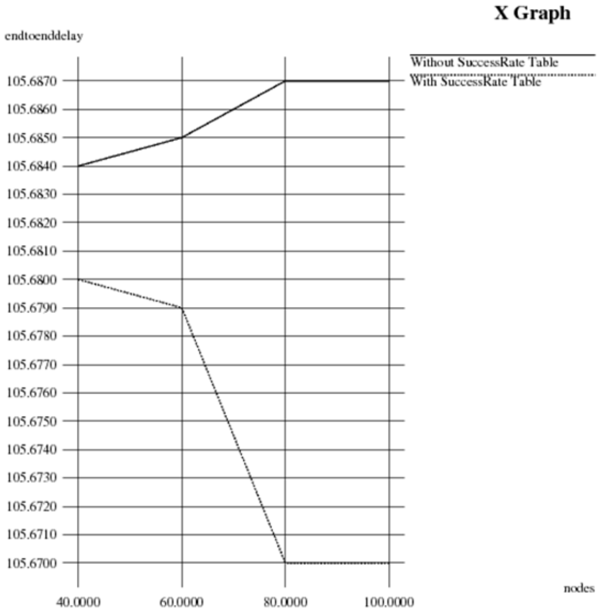

Fig 6 End to End Delay

The above Fig-4 shows the throughput rates for various number of nodes. The average rate of successful packets delivered are less when handshake concept was not used. But, when handshake – precheck algorithm with success rate table was used, the average rate of successful packets delivered are consistently more in number.

Fig-5 shows the number of packets delivered. The number of data packets dropped are very high without using handshake. It is observed that after using handshake concept with success rate table, the number of packets dropped are drastically reduced.

Fig-6 shows the delay which is constant for a long duration. After using handshake with success rate table, the delay is reduced to a large extent till it reaches the destination.

VII. CONCLUSION

In Mobile ad-hoc networks, gray hole and black hole attacks are most significant attacks. Even though many research work is done on black hole attack, We successfully attempted to detect and prevent gray/black hole attacks. We propose an algorithm which gives a feasible solution for detecting and removing malicious node with the help of success rate table. By adding a new column in routing table, We can let other nodes to know about malicious nodes in the network. This helps in finding a secure path for transmission.

REFERENCES

[1] D.Geetha and B.Revathi, "*AOMDV Routing Based Enhanced Security for Black hole attack in MANETs*", ICRTCT- 2013.

[2] Kamatchi.V, Rajeswari Mukesh and Rajakumar, "*Black hole attack Prevention using random dispersive routing for Mobile Ad-hoc Networks*", IJANS, Vol- 2,No-4,oct-2012.

[3] Naseera K.M and Dr.C.Chandrasekar, "*Prevention of black hole attack Using AOMDV*", IJERA, Vol-3, Issue 6, 2013.

[4] Neelam Khemariya and Ajay Khuntetha, "*An Efficient algorithm for Detection of Black hole attack in AODV based MANETs*", IJCA, Vol-66, No- 18,March-2013.

[5] Prathibha Bhat S , Vijaya Murari. T, "*Detecting and Removing Cooperative black or grey hole attacks in MANET*", IJETAE,Vol-4, Issue-7, July 2014.

[6] Shalini Jain, Mohit Jain and Himanshu Kandwal, " *Advanced Algorithm for detection and prevention of cooperative black and gray hole attacks in Mobile Ad-hoc Networks*", IJCA 2010,Vol-1,No-7.

[7] Shnamuganathan V, T.Anand, "*A Survey on Gray hole Attack in MANET*", IJCNWC, Vol- 2, Issue-6,2012.

[8] Sherril Sophie Maria Vincent, W.Thamba Meshach, "*Preventing Black hole attack in MANETs Using Randomized Multipath Routing Algorithm*", IJSCE, Vol-1,January- 2012.

[9] Vipan Chand Sharma, Atul Gupta and Vivek Dimri, "*Detection of black hole attack in MANET under AODV Routing Protocol*", Vol-3, Issue-6, Jun- 2013.

[10] Vishnu.K and Amos J.Paul, "*Detection and Removal of Cooperative black or Grey hole attack in Mobile Ad-hoc Networks*", IJCA, Vol-1.

[11] Network Simulator, www.isi edu/nsnam/ns.