International Journal of Engineering Applied Sciences and Technology, 2018
Vol. 2, Issue 12, ISSN No. 2455-2143, Pages 69-72
Published Online April 2018 in IJEAST (http://www.ijeast.com)

ENHANCED DETECTION OF NODES IN MOBILE ADHOC NETWORK

AMAN
M.Tech Scholar
SKITM , Bahadurgarh, India

Shabnam Sangwan
Asst. Professor CSE
SKITM, Bahadurgarh India

***Abstract*— It is a scheme for selfish node detection in MANET by overhearing other nodes. A buffer is maintained by each node for the packets sent recently and the packets within the buffer are compared with overhearing packet to check if there is a duplicate. Then the packet in the buffer is discarded and blank out by the watchdog. If the packet has stayed longer than a certain time-out in the buffer, then the watchdog will increase the fault count for the node culpable for sending the packet. If the count crosses some threshold, the node is considered to be misbehaving and a message about this node is sent to the source. in this paper Watchdog is presented in every node in the network Total number of packets incoming are equal to total number of packets out-going in watchdog**

***Keywords*— WATCHDOG, DSR,**

1. INTRODUCTION

Watchdog is presented in every node in the network. In the following Fig 3.1. Node S is a source and node D is a destination. Node S forward the packets to node Watchdog present in node S overhears the neighbor node.

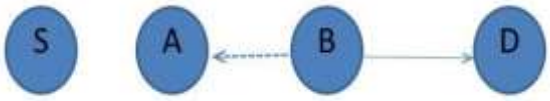

Figure 1: Watchdog

A whether it forward the packets to neighbor node B. Here node A forward the pack-ets to node B. Similarly, watchdog present in node A overhears whether node B forward the packets to node D. The problem with watchdog is partial dropping, false misbehavior, limited transmission power, receiver collisions and ambiguous collisions might not be detected. Path rater in the path rater includes the knowledge of link reliability data and misbehaving nodes to nd the most reliable route. Every node in the network maintains a metric for all the nodes it knows about. It measures a metric for path by balancing the node ratings in the route. Path with higher rating is chosen if multiple paths are there to same destination.

II. 2ACK Method

The 2ACK scheme [2] is used for detecting misbehaving link rather than detecting selfish nodes. For the existing routing protocols like DSR it can be used as an add-on. A xed route of 2 hops (3 nodes) in the direction that is opposite to the direction of data traffc is assigned to a 2Ack packet

At whatever point a route must be framed from the source to the destination, we rst utilize the essential directing protocol like DSR. To apply the 2ACK strategy, we picture the whole route as set of sequential covering triplets.

For example, if 1-A-B-C-D-E-2 represents a route from source to destination, then the 2Ack technique is applied to every triplet of the set: (1, A, B) (A, B, C) (B, C, D) (C, D, E) (D, E,2). Working of the technique shown below:

We consider triplet (A, B, C) for which the algorithm is applied, A sends a data packet to node B which has to be forwarded to node C along the route. Node A must be guaranteed of the effective gathering of the packet by node C through the acknowledgement packet 2Ack from C to B and from B to A. As such a reverse 2hop route is followed by 2Ack packet. The node C in the triplet is called 2Ack sender and node A is called 2Ack receiver. The timely and successful entry of 2Ack packets for each transmission guarantee node A that the link B-C is working well and not misbehaving.

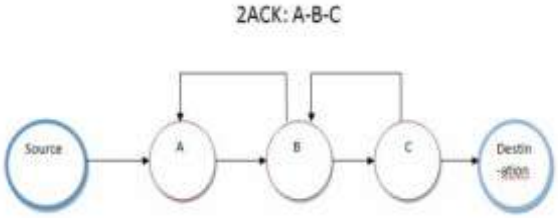

Figure 2: 2ACK

For all the triplets in the path 2Ack transmission takes place. Hence, the destination and just last node before the destination will not serve as a 2Ack receiver and very next router to the source node will not act as a 2Ack sender. Only some chunk of data packets is acknowledged for reducing the additional overhead in routing.

III. DISTRIBUTED APPROACH FOR DETECTING AND DELETING SELFISH NODES

The data processing and gathering module of the framework gather information in two ways [6], rst it generally runs an observing methodology to get the conduct data of neighbor

International Journal of Engineering Applied Sciences and Technology, 2018
Vol. 2, Issue 12, ISSN No. 2455-2143, Pages 69-72
Published Online April 2018 in IJEAST (http://www.ijeast.com)

nodes and besides it trades this data with different nodes checked data.

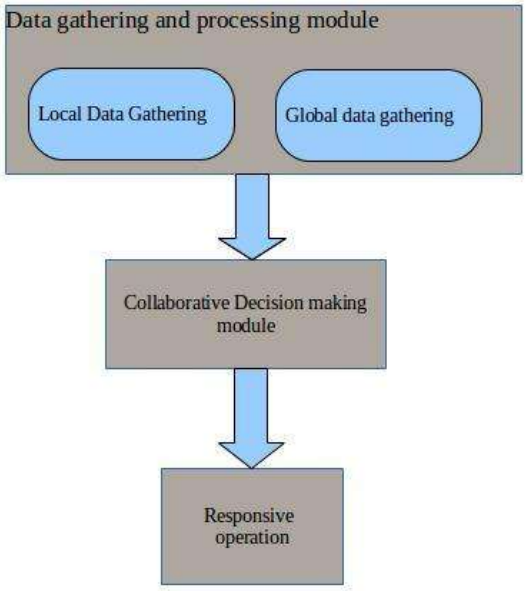

Figure 3: A Distributive approach

IV. AODV ROUTING PROTOCOL

With AODV algorithm multihop, self-starting, dynamic routing can be enabled between the mobile nodes that wish to maintain and establish an ad hoc network. It permits and helps mobile nodes in acquiring routes rapidly for new destinations, and does not oblige devices to keep up routes to destinations that are not in dynamic communication. This protocol enables mobile devices to react to the changes in network topology and link breakages in a timely and effcient way. In case if a link breaks, AODV helps in notifying the set of nodes that are affected so that the routes using the lost link can be invalidated.

In AODV four control messages are defined for maintaining routes to the desti-nation. These contol messages [16] include RREQ(RouteRequest) message, Hello message, RERR(RouteError) message and RREP(RouteReply). Periodically a hello message is broadcasted by every node in the network to all its neighbors to tell that it is alive. Whenever a neighboring node receives a hello message, the neighbor node includes the data about the node which sends a hello message into its routing table.

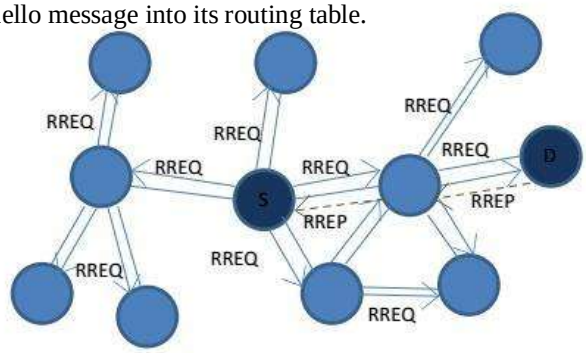

Figure 4: AODV Routing

Route request (RREQ) packet is broad-casted by the source node to all its neighbors in case if the routing table does not contain destination node [1]. Every neighboring node likewise rebroadcasts the gained route request (RREQ) messages to its neighbors. Through along these lines over and over until the destination node is reached. If the neighbor node accepts the route reply packet (RREP), it likewise replies conversely the Route reply packet to the former neighbor node as per the data in its routing table.

The trans-mission path can be created at the point when the route reply (RREP) message is sent again to the originating node. Throughout the information transmission, if in this transmission way a node is not able to communicate with the neighbor nodes, then a route error(RERR) message is sent by this node to the source node and the data that belongs to this transmission way is deleted from its routing table. The source node will retransmit RREQ packet for building a new transmission path when it receives a route error(RRER) message considering that the transmission path to the desired destination node has broken.

V. PROPOSED METHOD

In AODV, every selfish device merely means to spare its own resources for itself, it is simple for the node to turn into a selfish node overlook all messages (control and data) which are not intended to it. The nodes which don't send RREQ packets don't impact the network, this sort of selfish nodes can increase end to end delay because the number of nodes in the transmission path will increase.

In AODV routing protocol, a hello message is sent to obtain the neighbors information. Connectivity can be determined [20] by two variables using hello messages. ALLOWED HELLO LOSS and HELLO INTERVAL. Duration between the two hello messages of a node is known as the HELLO INTERVAL.

ALLOWED HELLO LOSS points out the greatest number of times of HELLO INTERVAL to hold up without getting a hello message before discovering a loss of connection to a neighbor. The prescribed worth for ALLOWED HELLO LOSS is two seconds and for HELLO INTERVAL is one. As it were, if a hello message is not accepted from a neighbor inside two seconds of the last message, connectivity lost is determined to that neighbor node.

In my proposal, every checking node works in promiscuous mode and might mon-itor the neighboring nodes which don't forward RREQ packet. Every checking node will maintain an entry for each of its neighboring nodes. In original AODV each node will contain the neighbor node address and the neighbor node expire time, newly added fields in the neighboring table are

International Journal of Engineering Applied Sciences and Technology, 2018
Vol. 2, Issue 12, ISSN No. 2455-2143, Pages 69-72
Published Online April 2018 in IJEAST (http://www.ijeast.com)

a. last helloTimer
b. last serviceTimer
c. node status

Neighbor nodes last service timer: Last service time is the time in which last time the neighboring nodes provided service to the network, providing services includes sending/forwarding RREQ packets, sending RREP/RRER and data packets.

Neighbor nodes last hello timer: Last hello time is the time recorded when the neighbor node has last sent the hello packet.
Neighbor nodes status: Status is the neighbor nodes current behavior recorded. Initially status of the neighbor nodes is initialized to zero, which is the behavior of the node is unknown.
The two fields last ServiceTimer and last HelloTimer are updated for every action performed. If the di erence between the neighbor nodes last HelloTimer and the last ServiceTimer is with in some threshold, then the node is considered as normal.
If it drops or do not react, then the checking node will mark the doubtful node as sel sh. In this proposed technique, every checking node will only regard its own data and will not claim with others, which removes false parsing and false accusation attacks.
The checking nodes will wait for this doubtful node to rebroad-cast the Route request message before some timeout. If the suspected node reacts, then the last service timer is updated and the node is considered as well behaved

Step by step procedure

STEP 1:
If a monitoring node hears a neighboring nodes data packet to forward it will check the difference between the last helloTimer and last serviceTimer.

STEP 2:

IF The di erence between the timers is within the threshold (last hello Timer - last service Timet threshold)

THEN The node is considered as normal and the last service time is updated (last serviceTime = CURRENT TIME).

ELSE The node is considered as suspicious node and further testing is conducted.

STEP 3: The monitoring node will broadcast a fake RREQ packet (with TTL=1 to reduce ooding) and waits for the doubtful node to rebroadcast the Route Re-quest message before time out.

STEP 4: IF The suspicious node responds before time out

THEN the last service timer (last serviceTimer = CURRENT TIME) is updated and labeled as normal node.

ELSE The suspicious node is labeled as selfish node (status = selfish).

VI. CONCLUSION

We notice that when the count of selfish nodes that don't transmit others route request packets are more than the TDR is less this is because when this kind of nodes are more in MANET, then most of the neighbor nodes will be selfish, and the normal nodes which are in the range of these selfish nodes cannot be identi ed. Hence, this will lessen the TDR of selfish nodes in the network.

VII. REFERENCE

1. Fahad, Tarag, and Robert Askwith. "A Node Misbehaviour Detection Mech-anism for Mobile Ad-hoc Networks." proceedings of the 7th Annual PostGrad-uate Symposium on The Convergence of Telecommunications, Networking and Broadcasting. 2006.

2. Banerjee, Sukla. "Detection/removal of cooperative black and gray hole at-tack in mobile ad-hoc networks." Proceedings of the World Congress on Engi-neering and Computer Science. 2008.

3. Bakar, Khairul Azmi Abu, and James Irvine. "A Scheme for Detecting Sel sh Nodes in MANETs using OMNET++." Wireless and Mobile Communications (ICWMC), 2010 6th International Conference on. IEEE, 2010.

4. Koshti, Dipali, and Supriya Kamoji. "Comparative study of Techniques used for Detection of Sel sh Nodes in Mobile Ad hoc Networks." International Jour-nal of Soft Computing and Engineering (IJSCE) ISSN (2011): 2231-2307.

5. Das, Samir R., Elizabeth M. Belding-Royer, and Charles E. Perkins. "Ad hoc on-demand distance vector (AODV) routing." (2003).

6. Roy, Debdutta Barman, and Rituparna Chaki. "Detection of Denial of Service Attack Due to Sel sh Node in MANET by Mobile Agent." Recent Trends in Wireless and Mobile Networks. Springer Berlin Heidelberg, 2011. 14-23.

7. Lin, H., Jos G. Delgado-Frias, and Sirisha Medidi. "Using a cache scheme to detect sel sh nodes in mobile ad

hoc networks." Communications, Internet, and Information Technology. 2007.

8. Chakeres, Ian D., and Elizabeth M. Belding-Royer. "AODV routing protocol implementation design." Distributed Computing Systems Workshops, 2004. Proceedings. 24th International Conference on. IEEE, 2004.

9. Chakeres, Ian D., and Elizabeth M. Belding-Royer. "The utility of hello mes-sages for determining link connectivity." Wireless Personal Multimedia Communications, 2002. The 5th International Symposium on. Vol. 2. IEEE, 2002.

10. Balakrishnan, Kashyap, Jing Deng, and Pramod K. Varshney. "TWOACK: preventing sel shness in mobile ad hoc networks." Wireless Communications and Networking Conference, 2005 IEEE. Vol. 4. IEEE, 2005.

11. Buchegger, Sonja, and Jean-Yves Le Boudec. "Performance analysis of the CONFIDANT protocol." Proceedings of the 3rd ACM international sympo-sium on Mobile ad hoc networking and computing. ACM, 2002.

12. Gupta, Shailender, C. K. Nagpal, and Charu Singla. "IMPACT OF SELFISH NODE CONCENTRATION IN MANETS." International Journal of Wireless and Mobile Networks 3.2 (2011).

13. Vijayan, R., V. Mareeswari, and K. Ramakrishna. "Energy based Trust solu-tion for Detecting Sel sh Nodes in MANET using Fuzzy logic." International Journal of Research and Reviews in Computer Science 2.3 (2011).

14. Wang, Yongwei, Venkata C. Giruka, and Mukesh Singhal. "A fair distributed solution for sel sh nodes problem in wireless ad hoc networks." Ad-Hoc, Mobile, and Wireless Networks. Springer Berlin Heidelberg, 2004. 211-224.