International Journal of Applied Engineering Research
ISSN 0973-4562 Volume 10, Number 7 (2015) pp. 18079-18087

http://www.ripublication.com

Securing DSR Protocol From Selfish Node Attack

[#1]S.Sundar,[*2]R.Kumar, [#3]Harish M.Kittur
[#]School of Electronics Engineering, VIT University Vellore – 632014, Tamilnadu, India.
**Senior Consultant, WIPRO Technologies Chennai, Tamilnadu, India.*
E-mail: [1] sundar.s@vit.ac.in,[2] rajagopal.kumar@wipro.com, [3]kittur@vit.ac.in

Abstract

Mobile ad-hoc networks (MANET) are self-configuringwireless networks without any particular infrastructure. MANET's are highly vulnerable to attacks due to continuously changing network topology, lack of central monitoring and lack of efficient defense mechanism. Selfish nodes in MANET's are the defective nodes which drop the packets that are not intended to them and presence of these nodes will impact on performance degrade of MANET's. In this paper, DSR (Dynamic Source Routing)protocol is used and a malicious selfish node is introduced in the network to analyze the selfish node attack and a prevention algorithm for selfish node attack is also suggested .Network parameters meters like throughput, end to end delay and packet delivery ratio are used for evaluation and comparison.Simulation tool used in this paper is NS2.

Keywords: MANET, DSR, Selfish Nodes,ABDSR, Throughput and Delay

Introduction

MANET's are a group of mobile nodes without any fixed infrastructure and nodes can communicate with other nodes in the network with the help of routing protocols. The protocol should not only aiming to find shortestpath between a source and destination but also satisfying meet certain other parameters like throughput, power, energy etc... [1] It is desirable that the protocol should be adaptive to the changes caused by mobility of nodes.In distance communication, MANETs uses intermediate nodes as relay nodes for communication. But these intermediate nodes are not always reliable and trusted. Especially in civilian communication such as provisioning of communication service to remote locations, the intermediate nodes may be from other service providers. In such situations there is no guarantee that the intermediate nodes can always be trusted.

According to [2],in MANETs critical functions like routing and forwarding performed by less trusted and less secured nodes. The intermediate nodes can act like selfish nodes in such a way that the nodes in order to save their battery life instead of forwarding sometimes could drop the packets.

The ad hoc networks are so popular since they do not need any fixed infrastructure. [3]Due to popularity of ad hoc networks, there are some nodes in the network acts in a negative way such that they can consume the resources of the network and these nodes are called as malicious nodes.

In the literature, there are many routing protocols have been proposed and in that AODV [4] and DSR [5] are popular among the researchers. In references [3] and [6] it is given that, these protocols are not possessing guards against attacks since they rely on principle of trust your neighbor's relationships.Hence presence of selfish nodes and malicious nodes can degrade the performance of these protocols. In this paper, DSR protocol is chosen for testing as it is simple reactive protocol which does not need periodical hello *(beacon)* packet which is used by a node to inform its neighbors of its presence.

This paper organized by discussing about DSR protocol in chapter II, Selfish node discussion in chapter III, Discussion about the proposal in chapter IV,Results in chapter V and Conclusion in chapter VI.

DSR Protocol

Dynamic source routing protocol (DSR) is a reactive protocol that is known as simple and efficient, specially designed for the multi-hop mobile ad hoc network [7]. Often called "On-demand" routing protocol as it involves determining the routing on demand unlike the pro- active routing protocols that has periodic network information. Network nodes use multiple- hops to communicate, DSR protocol plays a key role in determining and maintaining all the routing automatically as the number of hops needed changes at anytime and the mobile nodes involved may leave or join the network. DSR protocol involves two major mechanisms to establish the routing process. These are route discovery and route maintenance, which is the main mechanisms of the DSR protocol, allows the discovery and maintenance of source routes in the ad hoc network. DSR does not rely on functions like periodic routing advertisement, link status sensing or neighbor detection packets and because of the entirely on demand behavior, the number of overhead packets caused by DSR scales down to zero.

As DSR works entirely on demand and as nodes begin to move continuously, the Routing packet overhead automatically scales to only that needed to react to changes in the route currently in use. In response to a single Route Discovery if a node learns and caches multiple routes to a destination, it can try another route if the one it uses fails. The overhead incurred by performing a new Route Discovery can be avoided when the caching of multiple routes to a destination occurs. In wireless networks, differing antenna, propagation patterns or sources of interference can cause the link between two nodes to not work efficiently in either direction.

DSR Route Discovery

The header of the packet, which originates from a source node S to a destination node D, contains the source route, which gives the sequence of hops that the packet should traverse. A suitable source route is found normally when searching the Route Cache of routes obtained previously but if no route is found then the Route Discovery protocol is initiated to find a new route to D. Here S is the initiator and D the target.

The sequences of action is best explained in Fig 2.1 and in this,node A transmits a ROUTE REQUEST message, which is received by all the nodes in the transmission range of A [7]. Each ROUTE REQUEST message identifies the initiator and target of the Route Discovery and also contains a unique request ID, determined by the initiator of the REQUEST. Each ROUTE REQUEST also contains a record listing the address of each intermediate node through which this particular copy of the ROUTE REQUEST message has been forwarded. The initiator of the Route Discovery initializes the route record to an empty list. [7] When the target node receives the ROUTE REQUEST message, it returns a ROUTE REPLY message to the ROUTE Discovery initiator with a copy of the accumulated route record from the ROUTE REQUEST. This route is cached in the Route Cache when the initiator receives the ROUTE REPLY and is used in sending subsequent packets to this destination. When the target node finds a ROUTE REQUEST message from the same initiator bearing the same request ID or if it finds its own address is already listed in the route record of the ROUTE REQUEST message, it discards the REQUEST. If the target node does not find the ROUTE REQUEST message from the initiator, then it appends its address to the route record in the ROUTE REQUEST message and propagates it by transmitting it as a local broadcast packet. When Route Discovery is initiated the copy of the original packet is saved in a local buffer called Send Buffer. The Send Buffer contains copies of each packet that cannot be transmitted by the sending node. The packets are kept until a source route is avail-able or a timeout or Send Buffer overflow occurs. As long as a packet is in the Send Buffer, the node should initiate new Route Discovery until time out occurs or overflow of Buffer occurs. An exponential Back off algorithm is designed to limit the rate at which new ROUTE Discoveries may be initiated by any node for the same target.

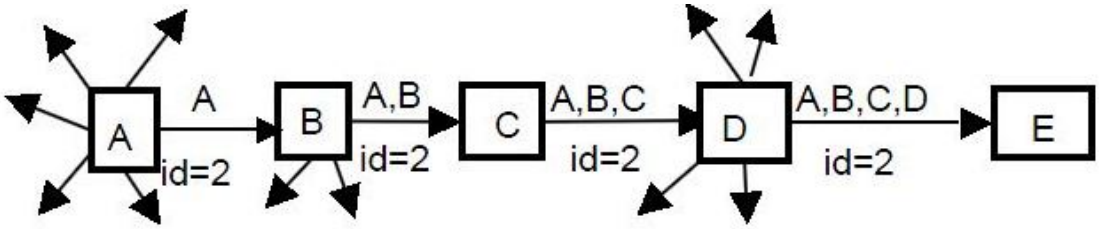

Figure 2.1: Node A Is The Initiator And Node E Is The Target

DSR Route Maintenance

When a packet with a source route is forwarded, each node in the source route makes sure that the packet has been received by the next hop in the source route. The

confirmation of receipt will be received only by re-transmitting the packet for a number of times. In Fig 2.2, it is shown that node A is the originator of a packet to the desired destination E. The packet has a source route through intermediate nodes B, C and D. Node A is responsible for receipt of the packet at B, node B at C, node C at D and node D at E. Node B confirms receipt of packet at C by overhearing C transmit the packet to forward it to D. The confirmation of acknowledgement is done by passive acknowledgements or as link-layer mechanisms such as option in MAC protocol. The node receiving the packet can return a DSR specific software acknowledgement if neither of the acknowledgements is available. This is done by setting up a bit in the packet's header and then requesting a DSR specific software acknowledgement by the node transmitting the packet. When a node is unable to deliver a packet to the next node then the node sends a ROUTE ERROR message to the original sender of the packet. The broken link is then removed from the cache by the originator of the packet and retransmissions to the same destination are done by upper layer protocols like TCP [7].

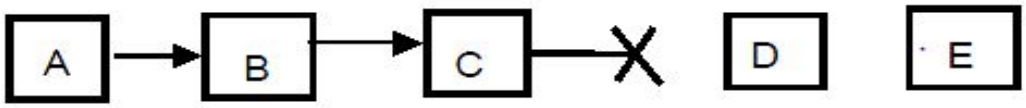

Figure 2.2: Node C Is Unable To Forward A Packet From A To E Over The Next Node D

Selfish Node Attack

Selfish node aims to save its resources to the maximum. This type of misbehaving node discards all incoming packets (control and data) except those which are destined to it. By dropping control packets, the nodes would not be included in the routing and then be released from being requested to forward data packets. The similarity of these two types of misbehaving is that they both use the network to forward their own packets but refuse to provide the same services back. Misbehaving nodes can significantly degrade the performance of a MANET [8][9]. Simulation shows that the percentage of misleading nodes can decrease the number of packets that are successfully delivered in the network. Selfish nodes on the other hand, have an impact on PDR. However, this type of misbehaving can increase the average end to end delay and PDR [4]. As the number of selfish nodes been increased, the source node will have less option on which route the data packets should travel. As a result, less attractive route will be selectedwhich means longer delays, and the selfish nodes will drop the packets. In this paper, we present a system to detect selfish nodes in a MANET [8]. An algorithm to overcome the negative effects of selfish node attack is also explained.

Proposed Algorithm

In this paper, the proposed algorithm and related simulationsareexplained in following subsections

A. Implementation of DSR protocol

In the first scenario, we are going to establish a communication using DSR protocol considering 26 nodes. Once the communication is initialized, RREQ messages are sent from Source node requesting for a route to the destination. The remaining nodes of the network sent RREP messages in reply to RREQ messages showing that they have a route to the destination. After the RREP messages are received at the source node Source node sends the data packets. This process is carried out using DSR protocol. The tcl commands for analyzing DSR protocol are executed by giving all the required parameters in NS2 and the output is obtained and it is shown below Fig 3.1

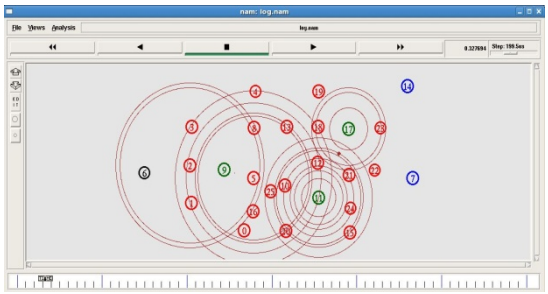

Figure 4.1: DSR Simulated Result For 26 Nodes

B. Simulation of Selfish Attack

Now we are going to introduce a malicious node into the network. In order to implement the black hole attack we have to make the some modifications. We can easily add the Black Hole behavior to any node. We Configured node 3 as black hole node by using necessary commands which areadded in TCL script of DSR protocol. Then we have to make changes in the backend C++ level by modifying *dsragent.cc* file we have to add is,

```
if (malicious == true)
{
// Drop Data packets to perform packet drop
drop (p.pkt, DROP_RTR_TTL);
return;
}
```

The above lines make the node to drop the packets which acts as a selfish node.

In this scenario consider the RREQ function because selfish node behavior is carried out as the selfish node receives an RREQ packet. When selfish node receives an RREQ packet it immediately sends RREP packet as if it has fresh enough path to the destination. Malicious node tries to deceive nodes sending such an RREP packet .When the source node receives this RREP messages, they will send the data

packets in that path. Once the data packets reach the malicious node, it will drop the packets leading to communication failure in the network as shown below figure

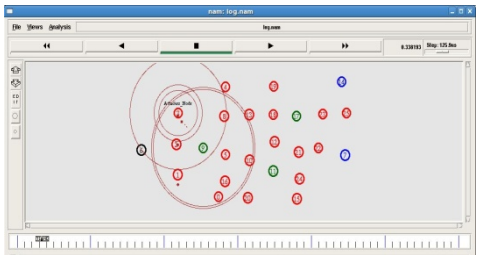

Figure 4.2: Simulation With Malicious Nodes

C. Prevention of Selfish Node Attack

In order to overcome the selfish node attack a new protocol called “ABDSR” is added in the NS2 directory.ABDSR means Association based DSR.This protocol uses the association between nodes to detect selfish nodes.We start the work by duplicating DSR protocol in the NS directory and change the name of the directory to “abdsr” such as *abdsragent.cc,* and *abdsragent.h*.We have changed all classes, functions, structs, variables and constants names in all the files in the directory except struct names that belong to dsrpacket. h code. We have designed DSR and ABDSR protocols to send each other dsr packets.

Detection of malicious node is done based on packet drop ratio at each node after every hop. A threshold value for PDR is set up and for every node PDR is calculated for the next hop.

If (PDR >threshold)
{
Node =selfish;
//choose alternate path
}
else if(PDR<threshold)
{Node=not selfish;
//forward packets through the node
The simulated results are shown below in Fig3.3,
}

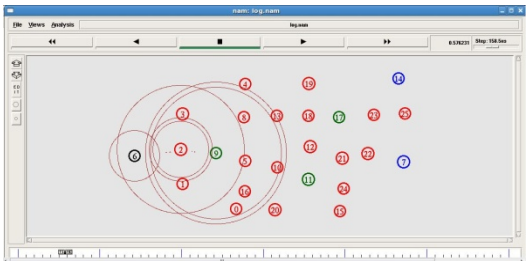

Figure 4.3: Implementation of ABDSR protocol

Results

All the three above mentioned scenarios are simulated in NS2 [10]. Throughput, end to end delay and packet loss ratio are calculated and graphs are plotted using xgraph in NS2.

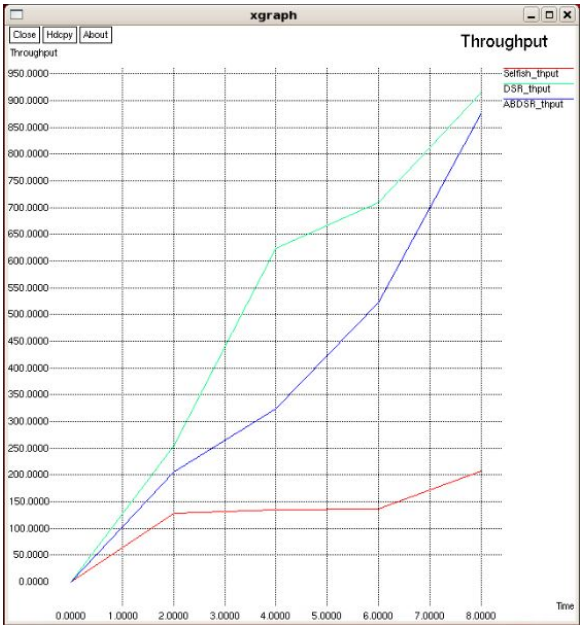

Figure 5.1: Comparison of Throughput

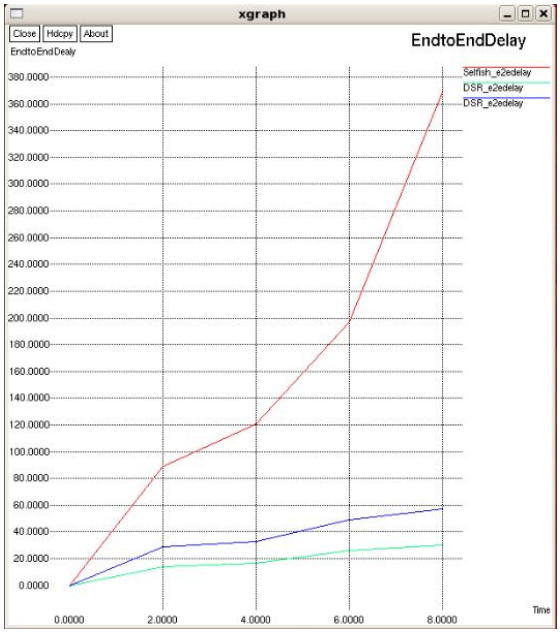

Figure 5.2: Comparison of End To End Delay

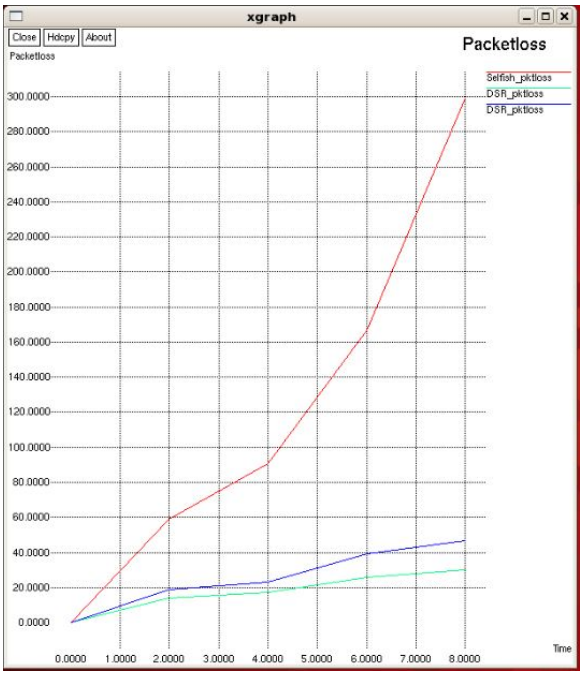

Figure 5.3: Comparison of Packet Loss Ratio

Conclusion

In this paper, the effect of selfish node attack in DSR was analyzed. For this purpose a DSR protocol that behaves as selfish in NS2 was implemented. Three scenarios which have 26 nodes that use DSR protocol was simulated and same scenarios after introducing one selfish node into the attack.Initially there is very less data loss in the DSR network. If a selfish Node is introduced in this network data loss is increased . When ABDSR protocol was used in the same network, the data loss is decreased. These results show that this solution reduces the selfish effect in a network using ABDSR protocol and improvement is shown in terms of parameters such as Throughput, Delay and improvement in Packet loss.

References:

[1] C.Siva Ram Murthy, B.S.Manoj, “Ad HOC Wireless Networks Architecture and Protocols”, Prentice Hall Communication Engineering and Emerging Technologies series.

[2] J.Vijithanand, K.Sreerama Murthy, “A Survey on Finding Selfish Nodes in Mobile Ad Hoc Networks” International Journal of Computer Science and Information Technologies, Vol. 3 (6), 2012, 5454-5461.

[3] Yasser khamayseh, Ruba Al-Salah, Muneer Bani Yassein, “Malicious Nodes Detection in MANETs: Behavioral Analysis Approach”, Journal of Networks, Vol. 7, No. 1, January 2012.

[4] Belding-Royer, E., Das, S., and Perkins, C., "Ad hoc On Demand Distance Vector (AODV) Routing", IETF Internet Draft, www.ietf.org, 1997.

[5] D.B. Johnson, and D.A.Maltz, ,"The Dynamic Source Routing in Ad Hoc Wireless Networks ",Mobile Computing, Kluwer academic publishers , vol 353,pp 153 to 181,1996.

[6] Kravets, R., Naldurg, P. and Yi, S., "Security-aware Ad Hoc Routing for Wireless Networks", In the ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC01), Long Beach, CA, 2001.

[7] Ana Cavalli, Cyril Grepet and Stephane Maag,"A Validation Model for the DSR protocol",GET / Institut National des Télécommunications, Distributed Computing Systems Workshops, 2004. Proceedings. 24th International Conference, 2004, Page(s): 768 - 773.

[8] Gagandeep, Aashima, Pawan Kumar, "Analysis of Different Security Attacks in MANETs on Protocol Stack A-Review", International Journal Of Engineering and Advanced Technology (IJEAT) ISSN: 2249 – 8958, Volume-1, Issue-5, June 2012.

[9] Dipali Koshti, Supriya Kamoji, "Comparative study of Techniques used for Detection of Selfish Nodes in Mobile Ad hoc Networks", International Journal of Soft Computing and Engineering (IJSCE) ISSN: 2231-2307, Volume-1, Issue-4, September 2011.

[10] F. J. Ros and P. M. Ruiz, "Implementing a New Manet Unicast Routing Protocol in NS2",http://masimum.inf.um.es/fjrm/wp-uploads/nsrt-howto/html/, December 2004.