Characterization of GCET Protocol in Ad Hoc Network

Anamika Rai, Ravikant Yadav, Mrs. Ruchi Asija

* Department of Computer Science & Engineering, Galgotias College of Engineering & Technology, Greater Noida

Abstract- An ad hoc wireless network is an autonomous self-organizing system of mobile nodes connected by wireless links where nodes not in direct range can communicate via intermediate nodes. A common technique used in routing protocols for ad hoc wireless networks is to establish the routing paths instead of continually maintaining a complete routing table. A significant concern in routing is the ability to function in the presence of byzantine failures which include nodes that drop, delay, or mis-route packets in an attempt to disrupt the routing service.

We propose a routing protocol for ad hoc wireless networks that provides resilience to byzantine failures caused by individual or colluding nodes. Our delay analysis technique detects a faulty link (mainly those which lead to packet drop, mis-route and delay). These links are then avoided by repetitively comparing their time delay and by using a beacon message in our route discovery protocol that leads to a least delay path to the destination.

Index Terms- Ad hoc network, Byzantine, Black hole Attack, Byzantine Wormhole Attack, Delay value analysis technique.

I. INTRODUCTION

Ad hoc wireless networks are self-organizing multi-hop wireless networks where all the hosts (nodes) take part in the process of forwarding packets.It is relatively better communication methodology which has a rapid growth of wireless gadget, such as laptop, PDAs wireless sensors and Wireless phones, shows the importance of wireless technology becoming more prominent day by day. The Infrastructure networks rely on a fixed base station or access point, where all the mobile nodes are connected to it. The infrastructure less networks is the ad hoc networks, where all the mobile nodes are connected to each other with the absence of an access point a centralized point of management. A mobile ad hoc network consists of nodes. Nodes within radio range of each other can communicate directly over wireless links, and those that are far apart use other nodes as relays. Each host in an ad hoc wireless network also acts as a router and routers are mostly multi hop. Ad hoc wireless network[1] is self –organized in such a way that a collection of mobile nodes without the help of any fixed infrastructure and central management is formed automatically. Each node is equipped with a wireless receiver and transmitter that communicate with other nodes which lie in its radio communication range. Ad hoc wireless network is dynamic in nature and they constantly move in and out of their network limits (range). Ad hoc networks can easily be deployed since they do not require any fixed infrastructure, such as base stations or routers. Therefore, they are highly applicable to emergency deployments, natural disasters, military battle fields, and search and rescue missions.

Initially, Ad hoc wireless network was designed for military applications, but later on it found its new usage. For example, search and rescue mission, data collection, virtual classes and conferences where laptops, PDA or other mobile devices are in wireless communication. A key component of ad hoc wireless networks is an efficient routing protocol, since all of the nodes in the network act as routers. Some of the challenges faced in ad hoc wireless networks include high mobility and limited power resources. Consequently, ad hoc wireless routing protocols must use battery power efficiently.

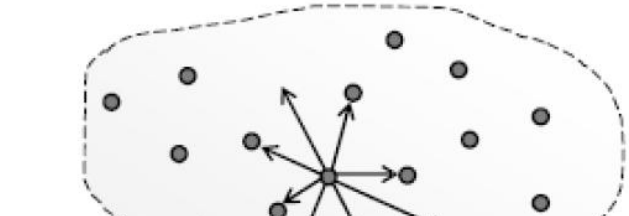
In this routing protocol, nodes initiate a route discovery process only when data packets need to be routed. Discovered routes are then analysed by time delay comparison for selecting an efficient path. Here we assume that there is proper security in the ad hoc network where this protocol is working. Although one might assume that once authenticated, a node should be trusted, there are many scenarios where this is not appropriate. For example, when ad hoc networks are used in a public Internet access system (airports or conferences), users are authenticated by the Internet service provider, but this authentication does not imply trust between the individual users of the service. Also, mobile devices are easier to compromise because of reduced physical security, so complete trust should not be assumed.

Since ad hoc wireless network is being used widespread, security has become a very important issue[2]. Therefore, only a node that compromises with an attacking node can cause the network to fail. The dynamic and cooperative nature of the ad-hoc routing infrastructure also imposes additional security threats. Dynamic changes to the network topology make it difficult to detect if a node providing false routing information is Byzantine[3] or is just out of sync with the topological changes. These security threats must be considered, when designing security mechanisms for a wireless ad-hoc network. This paper focuses mainly on routing process[4] assuming a pre-secured system that detects inappropriate or malicious activity on a computer or network. This paper is structured as follows. In section 2 we discuss about misbehaving or critical nodes in Ad hoc wireless network. In section 3 we present the algorithm of our protocol. In section 4 we discuss the detailed technique proposed for preventing byzantine faults and finally provide a comprehensive analysis of results in section 5.

II. PROBLEM DEFINITION

We consider only the source and destination to be trusted, and assume that there exists anon-adversarial path between source and destination. Any intermediate node on the path between the source and destination may exhibit Byzantine

behavior[7]. The goal of our protocol is to avoid Byzantine behavior, or bound its effect on the overall system. We assume that an intermediate node can exhibit such behavior either alone or in collusion with other nodes. We do not address resource consumption attacks where a node receives a high number of messages with incorrectly authenticated packets[8]. We define *Byzantine behavior* as any action by an authenticated node performed at the network layer that results in disruption or degradation of the routing service (i.e. the attacker does not have control over the MAC or physical layers). Attacks like eavesdropping, fabricating or modifying packets can be prevented by traditional encryption, authentication and integrity mechanisms. More complex attacks include manipulating the route discovery phase of the protocol to control the path establishment. Examples of such attacks are making a path appear either longer or shorter than it is. One simple way to do this is by modifying the information that propagates on the packet, such as the hop count or the list of nodes on the path. A basic Byzantine attack is referred to as a *black hole attack* [2,5] where the adversary drops data packets (entirely or selectively), while still participating in the routing protocol. As a result, whenever an adversarial node is selected on a path, data will be lost partially or entirely on that path. Authentication techniques cannot prevent the attack, since once a node is compromised and under adversarial control, all the cryptographic keys are available to the attacker. Thus, the attacker can generate messages that appear to be authentic. Besides attacks that can be performed individually by one malicious node, we consider stronger forms of Byzantine attacks that involve colluding malicious nodes which coordinate their actions. Examples of such attacks are the *Byzantine wormhole* attack[9] and its stronger variant the *Byzantine overlay network wormhole*. In a *Byzantine wormhole a*ttack two colluding adversaries cooperate by tunneling packets between each other in order to create a shortcut (or wormhole) in the network. This tunnel can be created by using a private communication channel, such as a pair of radios and directional antennas, or by using the existing ad hoc network infrastructure. The adversaries can then use the low cost appearance of the wormhole in order to increase the probability of being selected on paths, and then attempt to disrupt the network by selectively dropping the data packets or to perform traffic analysis. Note that for a Byzantine wormhole, the wormhole link exists between two compromised (adversarial) nodes, while in a traditional wormhole two honest nodes are tricked into believing that there exists a direct link between them. Wormhole detection techniques proposed against traditional wormholes, are ineffective in the case of Byzantine wormholes due to the trust of the wormhole link end points (which are adversarial).A *Byzantine overlay network wormhole* attack is a more general (and stronger) variant of the previous attack and occurs when several nodes are compromised and form an overlay network. By tunneling packets through the overlay network, the adversaries make it appear to the routing protocol that they are all neighbors, which considerably increases their chances of being selected on routes and facilitates further attacks[9].We do not consider general attacks such as Sybil and node replication attacks.



III. GCET ROUTING PROTOCOL

Our protocol establishes a reliability metric based on past delay value analysis and uses it to select the best path. The metric is represented by a list of delay values where high delay values correspond to low reliability. Each route in the network maintains its own list, referred to as a reliability list, and dynamically updates that list when it detects faults.

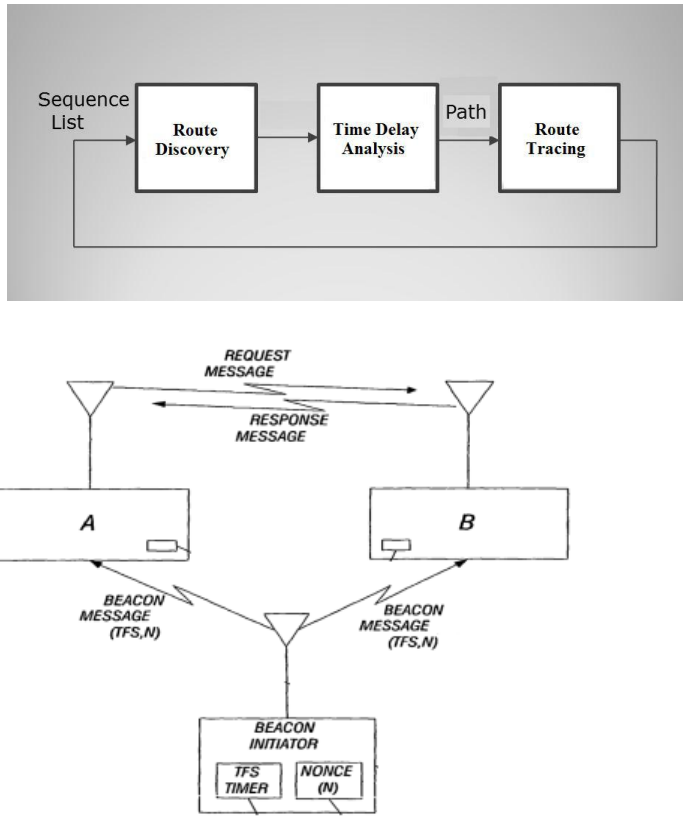


Fig. 1 Routing Protocol Phases

These links are avoided using a secure route discovery protocol that incorporates the reliability metric. More specifically, our routing protocol can be separated into three successive phases, each phase using as input the output from the previous.

1) Route discovery: Using flooding and beacon message, this phase sends and outputs the initial path from the source to the destination.

Fig.2 Flooding

2) *Delay value Analysis:* This phase analyses the delay value involved for respective beacon messages and maintains a delay value list for each pair nodes. The delay value list is used by the efficient route tracing phase to avoid faulty path.

3) Efficient Route tracing: This phase maintains a reliability list of links discovered by the time delay management algorithm. The delay value list is used by the route discovery phase in cyclic manner to generate effective route.

1) ***Route Discovery:*** When a source node wants to transmit data to a destination node and if it doesn't has a path to destination node in the routing table, it broadcasts the beacon message to establish a main route. When intermediate nodes receive the beacon message and they are not the destination that is determined in the packet, take action similar to conventional AODV[6] protocol. If the destination node receives beacon, sends the reply to the source node, and after net traversal time (in milliseconds) sends reply to the source node. If the source node receives the reply, the main route is established and starts data transmission.

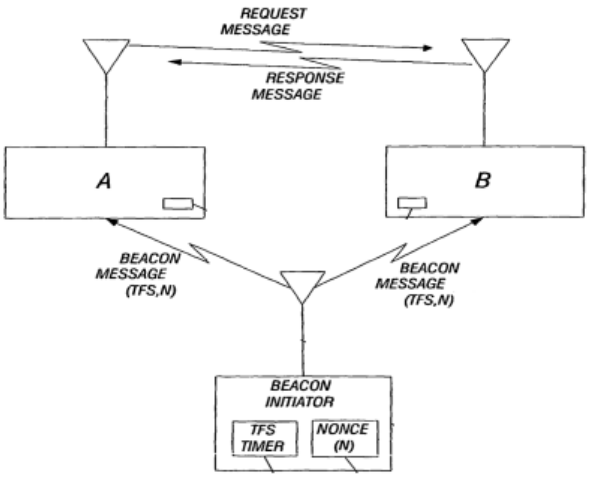


Fig.3 Transmission of Beacon message

2) ***Delay Value Analysis:*** This phase aims to form a list of the delay values involved in route discovery phase. Beacon message transmission acts a medium for this goal. The delay values are compared with optimum threshold value which determines whether the link is efficient or not. In case, the delay value is greater than the threshold value, the link is rejected. Else, the link proves to be efficient as it is within the acceptable range. The threshold value is determined as per analysis of data obtained in initial stages.

3) ***Route Tracing:*** After analyzing the links on basis of delay values, a list efficient links is maintained. These links are used further to establish efficient route for data transmission. The route is traced on the basis of original algorithm. This route has minimum risk from the foreseen byzantine failures such as packet drop, delay, mis-route. Thus, it is used for successful transmission of data packets.

IV. ALGORITHM

1. Assume the Beacon message is pre-encrypted.
2. Broadcast it using flooding technique.
3. When all the nodes in the network are available.
4. Max distance between any two nodes is fixed at some standard (let D).
5. So max time of message delivery is estimated which is used as threshold.
6. Now when reply is generated then the time delay between the links is compared to obtain the best nodes (in reply time) in the network.
7. Hence a route with least time delay (let t_l) is traced to send a packet to the destination.
8. Further, when packet passing is done then the initial node chooses that neighbor node which has least time delay for reply (t_l= min $\{t_0, t_1 t_2 t_3 \ldots\ldots\}$).
9. In this way least delay links are chosen among the nodes.
10. Further the same process is repeated for each pair of nodes to obtain a route.
11. If an error comes then error message (Rerr) is generated towards the source.
12. Finally, the packet is successfully transmitted to the destination.

V. ANALYSIS & RESULTS

The Implementation of the GCET routing protocol is done on NS2[11]. The Network Simulator (ns2) network simulation package will be the supported tool for this class. You may run ns2 using the ECE Graduate Workstation Laboratory or you can install it on your own Linux. Tcl, Tk, Otcl, and Tclcl[12] are support programs that are used by ns2, which is the simulation program. Nam is the network simulator which provides visual views of the network simulation.

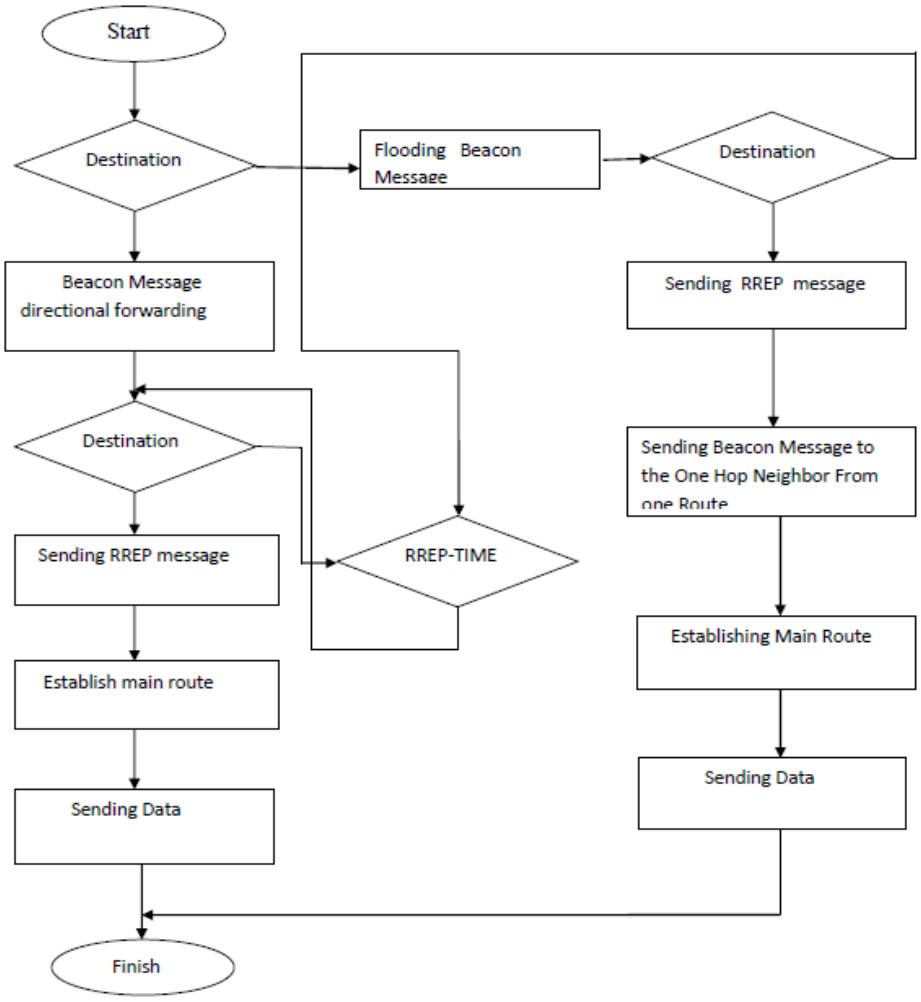


Fig.4 Routing Flowchart

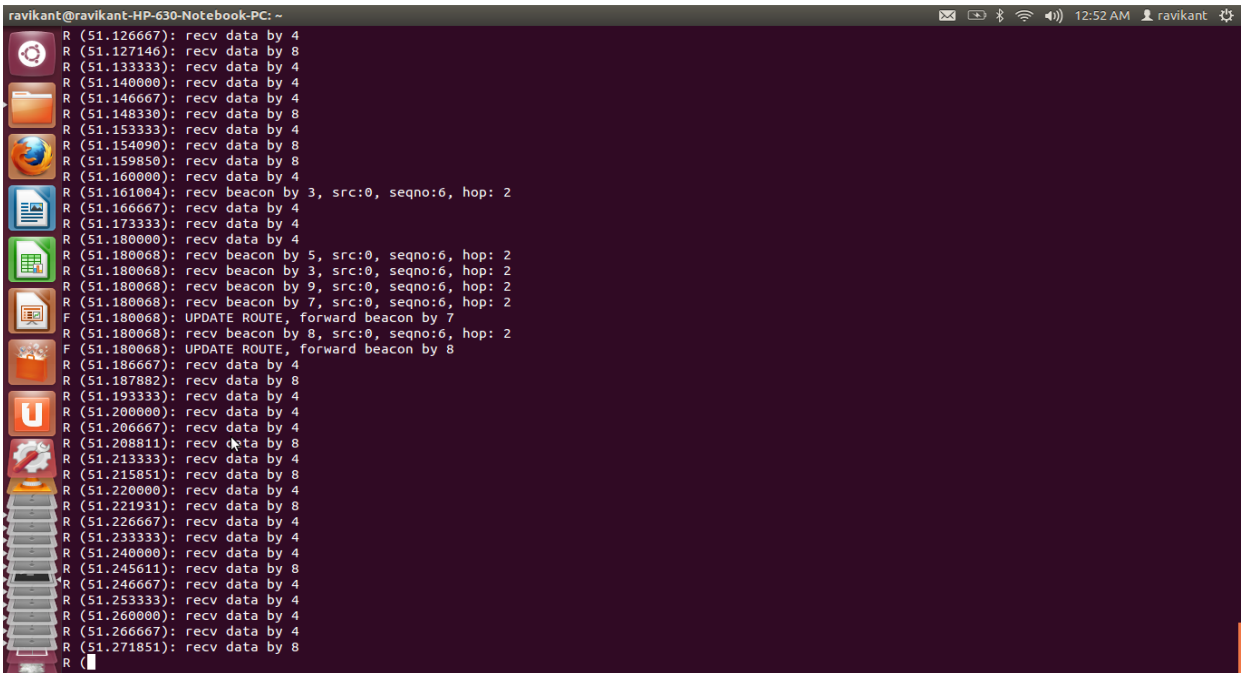


Fig.5 Processing of TCL file on Ubuntu Terminal by NS2

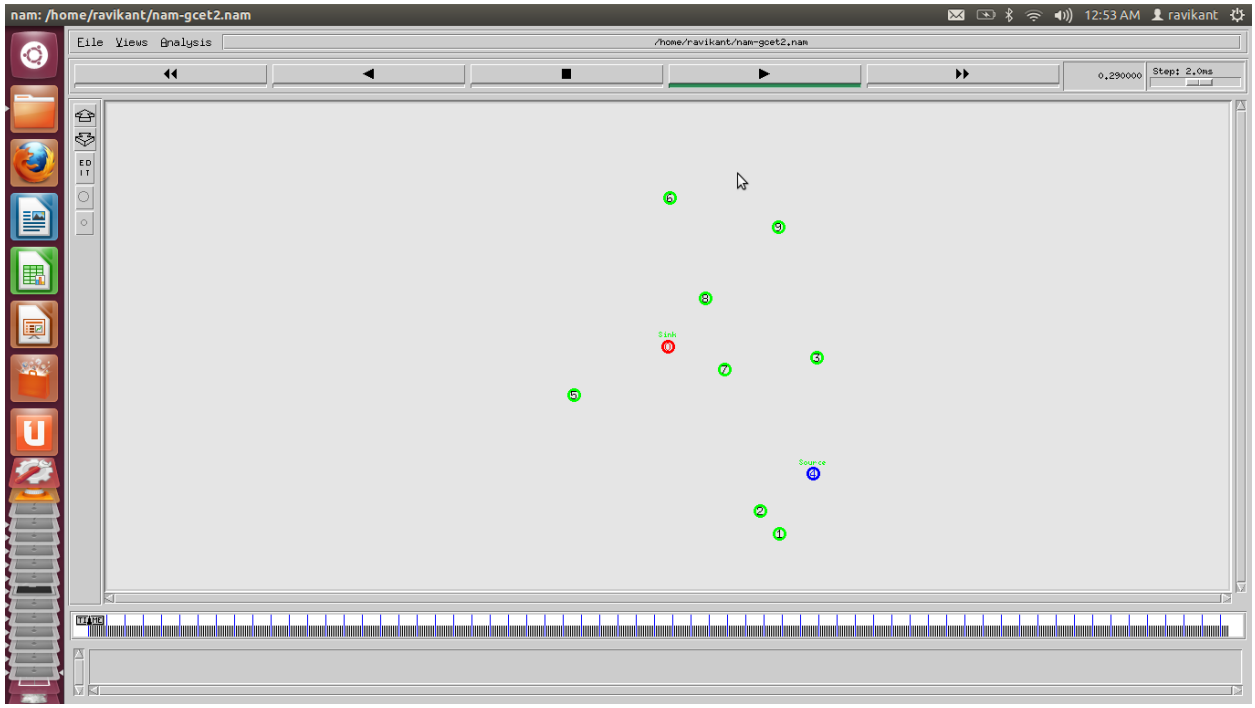


Fig.6 Visual of NAM Trace initially. Here sink0, source-4

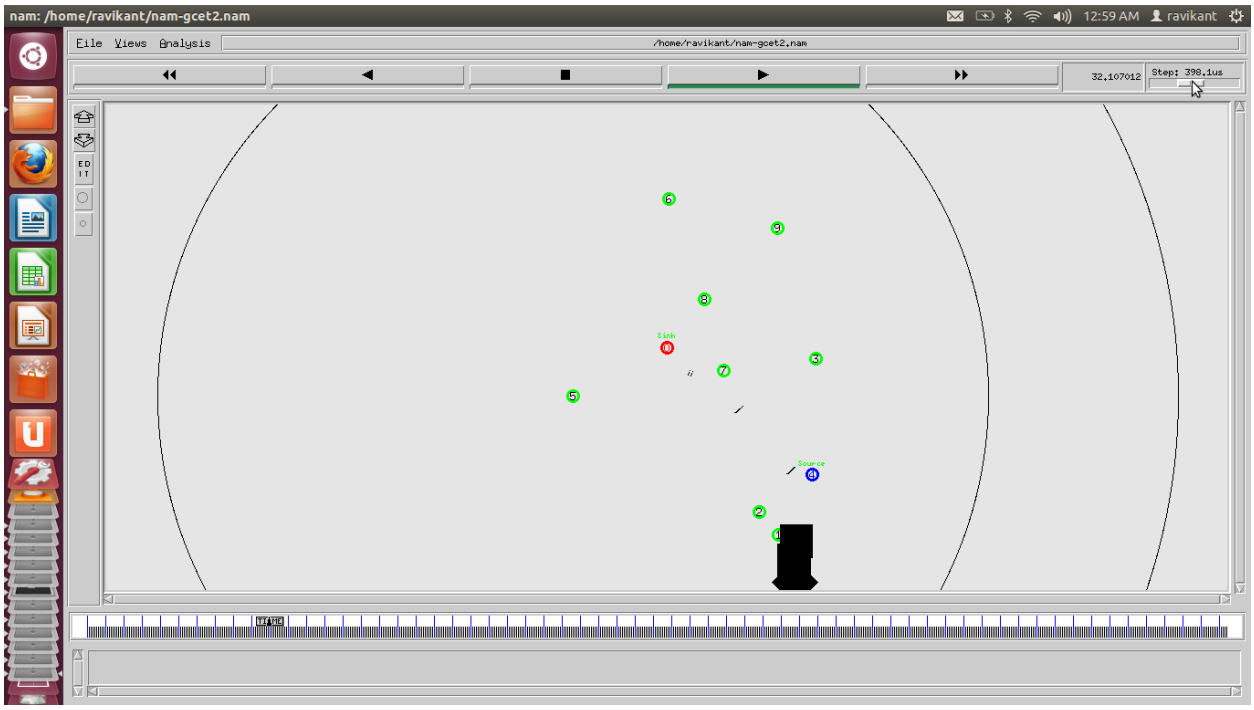


Fig.7 Visual of Beacon Message transmission in Network

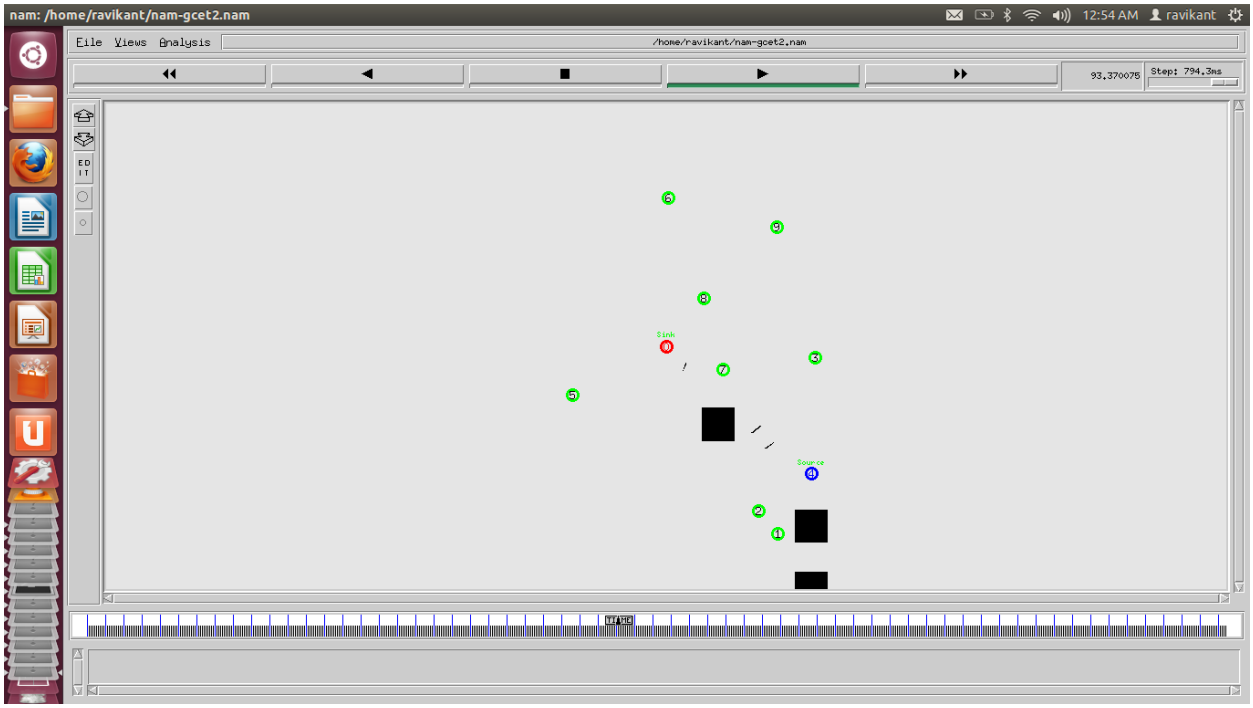


Fig.8 Data Transmission to sink using Droptail

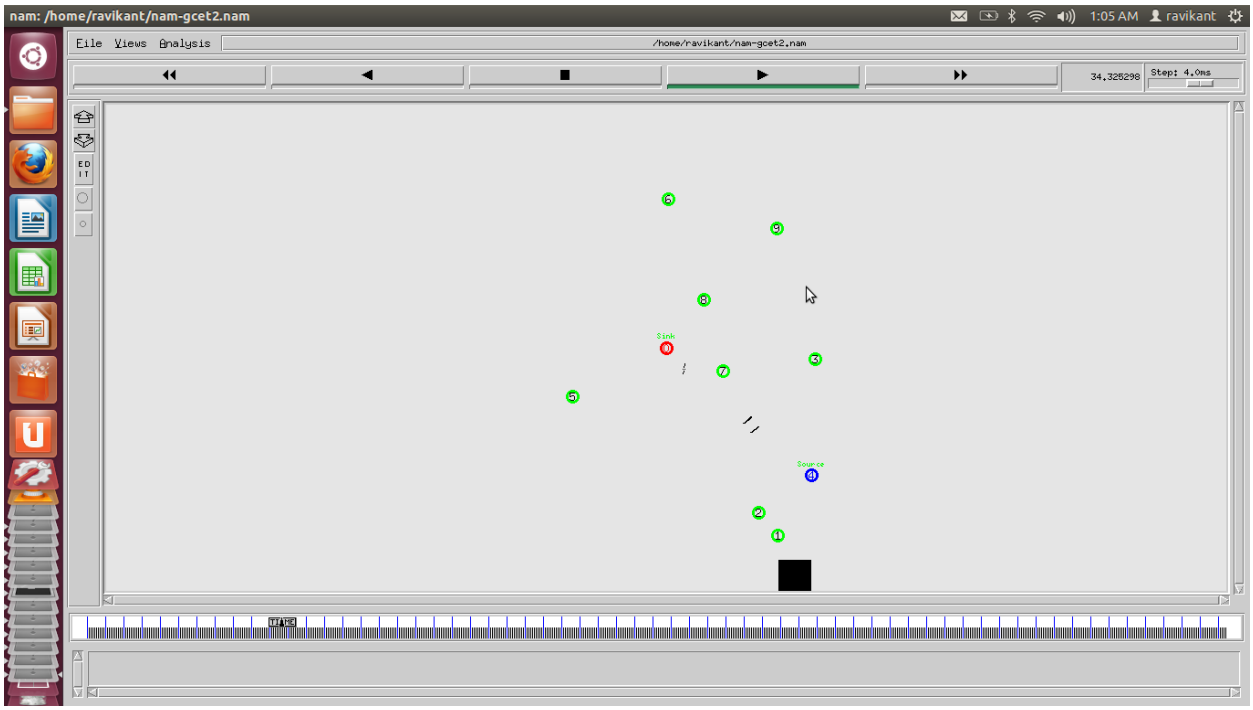


Fig.9 Again transmission of data continues from Source to Sink

VI. CONCLUSION

In this paper, we have illustrated the GCET Routing protocol which uses an on-demand approach for finding routes, that is, a route is established only when it is required by a source node for transmitting data packets. It employs destination sequence numbers to identify the most recent path.

In Ad Hoc wireless networks, due to high network traffic. A number of recent works have been studied before implementing this new methodology. The implemented solution unlike some of its predecessors does not depend on distance metrics but mainly focus on transmission time which leads to an efficient system formation. Currently more studies are being done to implement probing method based on Divide and Conquer technique to add tolerance of several errors to the protocol.

ACKNOWLEDGMENT

This research paper is made possible through the help and support from everyone, including our parents, teachers, family and friends.

We would like to thank Mrs. Ruchi Asija for her support and encouragement and thanks to our parents, family, and friends, who provide the advice and support. This research paper would not be possible without all of them.

REFERENCES

[1] P. Yau and C. J. Mitchell, "Security Vulnerabilities in Ad hoc Network".

[2] Y. A. Huang, W. Fan, W. Lee, and P. S. Yu, "Cross-feature analysis for detecting ad-hoc routing anomalies," in The 23rd International Conference on Distributed Computing Systems (ICDCS'03), pp. 478-487, May 2003.

[3] LAMPORT, L., SHOSTAK, R., AND PEASE, M. 1982. The byzantine generals problem. *ACM Trans. Program. Lang. Syst. 4*, 3, 382–401.

[4] Y. A. Huang and W. Lee, "Attack analysis and detection for ad hoc routing protocols," in The 7th International Symposium on Recent Advances in Intrusion Detection (RAID'04), pp. 125-145, French Riviera, Sept. 2004.

[5] M. A. Shurman, S. M. Yoo, and S. Park, "Black hole attack in wireless ad hoc networks," in ACM 42nd Southeast Conference (ACMSE'04), pp. 96-97, Apr. 2004.

[6] C. E. Perkins, E. M. B. Royer, and S. R. Das, Ad hoc On-Demand Distance Vector (AODV) routing, RFC 3561, July 2003.

[7] Mohapatra, P., Gui, C., and Li, J., "Group Communications in Mobile Ad Hoc Networks", University of California, Davis.

[8] K. Sanzgiri, D. LaFlamme, B. Dahill, B. N. Levine, C. Shields, and E. M. B. Royer, "Authenticated routing for ad hoc networks," IEEE Journal on Selected Areas in Communications, vol. 23, no. 3, pp. 598-610, Mar. 2005.

[9] Y.-C. Hu, A. Perrig, and D. B. Johnson, "Packet leashes: A defense against wormhole attacks in wireless ad hoc networks," in *Proceedings of the 22nd Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*, April 2003.

[10] K. Sanzgiri, B. Dahill, B. N. Levine, C. Shields, and E. Belding-Royer, "A secure routing protocol for ad hoc networks," in 10th IEEE International Conference on Network Protocols (ICNP'02), November 2002.

[11] "The network simulator - ns2." http://www.isi.edu/nsnam/ns/.

[12] Webopedia, An Internet Dictionary, http://www.webopedia.com/.

AUTHORS

First Author – Anamika Rai, Student, Department of Computer Science & Engineering, Galgotias College of Engineering & Technology, Greater Noida, India

Email- anamikarai2010@gmail.com
Second Author – Ravikant Yadav, Student, Department of Computer Science & Engineering, Galgotias College of Engineering & Technology, Greater Noida, India
Email- ravikant.rvkt@yahoo.in
Third Author – Mrs.Ruchi Asija, Astt. Professor, Department of Computer Science & Engineering, Galgotias College of Engineering & Technology, Greater Noida, India
Email- er.ruchiasija@gmail.com

Correspondence Author – Ravikant Yadav
Email- ravikant.rvkt@yahoo.in
Alt. Email- rvktydv@hotmail.com
Contacts- +91-8574680023, +91-9718280473, +91-9415042761