Global Journal of Computer Science and Technology
Volume 12 Issue 4 Version 1.0 February 2012
Type: Double Blind Peer Reviewed International Research Journal
Publisher: Global Journals Inc. (USA)
Online ISSN: 0975-4172 & Print ISSN: 0975-4350

Stable and Secured Routing Strategy for MANET with SSRP

By Sunil Taneja &Ashwani Kush
Kurukshetra University

Abstract - A Mobile Adhoc Network (MANET) is characterized by mobile nodes, multihop wireless connectivity, infrastructureless environment and dynamic topology. The adhoc environment is accessible to both legitimate network users and malicious attackers. Moreover, as the wireless links are highly error prone and can go down frequently due to mobility of nodes, therefore, stable and secure routing over MANET is still a very critical task due to highly dynamic environment. In this research paper, a new protocol SSRP (Stable and Secured Routing Protocol) has been proposed. An experimental analysis of proposed protocol (SSRP) and existing protocol (AODV) has been carried out using network simulator ns-2. An effort has been made to perform analysis using random way point mobility model. The results have been derived using a self created network scenarios for varying number of mobile nodes. The same scenario is executed for both the protocols to analyze the performance. The performance metrics used for evaluation are packet delivery ratio, average end to end delay, throughput, normalized routing load and packet loss. Based on the experimental analysis, recommendations have been made about the significance of either protocol in various situations. It has been concluded that the proposed protocol i.e. SSRP provides a robust, stable and secured routing strategy for mobile adhoc networks.

Keywords : *MANET, Routing, Secured, SSRP, Stable*

GJCST Classification: *C.2.1*

Strictly as per the compliance and regulations of:

Stable and Secured Routing Strategy for MANET with SSRP

Sunil Taneja[α] & Ashwani Kush[σ]

Abstract - A Mobile Adhoc Network (MANET) is characterized by mobile nodes, multihop wireless connectivity, infrastructureless environment and dynamic topology. The adhoc environment is accessible to both legitimate network users and malicious attackers. Moreover, as the wireless links are highly error prone and can go down frequently due to mobility of nodes, therefore, stable and secure routing over MANET is still a very critical task due to highly dynamic environment. In this research paper, a new protocol SSRP (Stable and Secured Routing Protocol) has been proposed. An experimental analysis of proposed protocol (SSRP) and existing protocol (AODV) has been carried out using network simulator ns-2. An effort has been made to perform analysis using random way point mobility model. The results have been derived using a self created network scenarios for varying number of mobile nodes. The same scenario is executed for both the protocols to analyze the performance. The performance metrics used for evaluation are packet delivery ratio, average end to end delay, throughput, normalized routing load and packet loss. Based on the experimental analysis, recommendations have been made about the significance of either protocol in various situations. It has been concluded that the proposed protocol i.e. SSRP provides a robust, stable and secured routing strategy for mobile adhoc networks.

IndexTerms : *MANET, Routing, Secured, SSRP, Stable*

I. Introduction

MANET is a collection of wireless mobile nodes forming a temporary network without any fixed infrastructure where all nodes are free to move about arbitrarily and where all the nodes configure themselves. Unlike traditional networks whereby routing functions are performed by dedicated nodes or routers, in MANET, routing functions are carried out by all available nodes. There are no fixed base stations and each node acts both as a router and as a host. The mobile nodes in the adhoc network dynamically establish routing among themselves to form their own network 'on the fly'. In essence, the network is created in ad-hoc fashion by the participating nodes without any central administration. Further adhoc networks can be classified as single-hop or multi-hop. In single-hop adhoc networks, nodes are in their reach area and can communicate directly but in case of multi-hop, some nodes are far and cannot communicate directly.

The traffic has to be forwarded by other intermediate nodes. Adhoc networks are primarily meant for use by military forces or for emergency rescue situations. At the state of war an army cannot rely on fixed infrastructure because it is an easy and attractive target for the enemy. Adhoc networks are optimal solution in such cases. For civil use adhoc networks are crucial if the fixed infrastructure has been torn down by some natural disaster, like a flood or an earthquake. Then rescue operations could in such a situation be managed through utilizing adhoc networks. Security is an important issue for adhoc networks, especially for those security-sensitive applications. The dilemma is that how should it be judged whether the mobile adhoc network is secure or not. Some of the main security attributes [1, 2] that are used to inspect the security state of the mobile adhoc network are availability, integrity, Confidentiality, Authenticity, Non repudiation, Authorization and Anonymity.

In mobile adhoc networks, radio transmission is the most common means of communication. Eavesdropping on a node is far easier than in wired networks. Since intermediate nodes no longer belong to a trusted infrastructure, but may be eavesdroppers as well, consequent end-to-end encryption is mandatory. Next, as all nodes in an Adhoc network cooperate in order to discover the network topology and forward packets, denial of service attacks on the routing function are very easy to mount. Nodes may create stale or wrong routes, creating black holes or routing loops. Furthermore, in adhoc networks exists a strong motivation for non-participation in the routing system. Both the routing system and the forwarding of foreign packets consume a node's battery power, CPU time, and bandwidth, which are restricted in mobile devices. Consequently, selfish nodes may want to save their resources for own use. There are three main causes for a node not to work according to the common routing protocol. Malfunctioning nodes are simply suffering from a hardware failure or a programming error. Although this is not an attack, they may cause severe irritation in the routing system of an adhoc network. Selfish nodes try to save their own resources, as described above. Malicious nodes are trying to sabotage other nodes or even the whole network, or compromise security in

Author α : Department of Computer Science, Smt. Aruna Asaf Ali Government Post Graduate College, Kalka-133 302, Haryana, India affiliated to Kurukshetra University, Kurukshetra, India
(phone : +919467237272; fax : +91 1733 220019
E-mail : suniltaneja.iitd@gmail.com
Author σ : Department of Computer Science, University College, Kurukshetra University, Kurukshetra-132 119, Haryana, India
E-mail : akush20@gmail.com

some way. Before developing a security framework that prevents selfish or malicious nodes from harming the network, it is worthwhile to first create a structured overview on what kinds of attacks are possible in adhoc networks. Network security attacks [1, 2] are typically divided into two categories as passive and active attacks which has been shown in table 1.

In passive attacks, the malicious entity only listens to the traffic, without modifying or disturbing it in any way. It can be either eavesdropping or traffic analysis. In an active attack, the malignant node actively disturbs the normal operation of the network and an unauthorized party makes modifications to a message, data stream or file. Active attacks may take the form of one of four types: masquerading, replay, message modification, and denial-of-service (DoS).

Table 1 Passive Vs. Active Attacks

Passive attacks: Eavesdropping, traffic analysis
Active attacks: Masquerading, Replaying, Message modification, DoS

II. Security Challenges

Security has become a primary concern in order to provide protected communication between mobile nodes in a hostile environment. The salient features of adhoc networks pose both challenges and opportunities in achieving the aforementioned goals. *First,* use of wireless links renders an adhoc network susceptible to link attacks ranging from passive eavesdropping to active impersonation, message replay, and message distortion. Eavesdropping might give an adversary access to secret information, violating confidentiality. Active attacks might allow the adversary to delete messages, to inject erroneous messages, to modify messages, and to impersonate a node, thus violating availability, integrity, authentication, and non-repudiation. *Secondly*, nodes, roaming in a hostile environment e.g. in a battlefield with relatively poor physical protection, have non-negligible probability of being compromised. Therefore, one should not only consider malicious attacks from outside a network, but also take into account the attacks launched from within the network by compromised nodes. Therefore, to achieve high survivability, adhoc networks should have a distributed architecture with no central entities. Introducing any central entity into our security solution could lead to significant vulnerability; that is, if this centralized entity is compromised, then the entire network is subverted. *Thirdly,* an adhoc network is dynamic because of frequent changes in both its topology and its membership. Trust relationship among nodes also changes, for example, when certain nodes are detected as being compromised. Unlike other wireless mobile networks, such as mobile IP, nodes in an adhoc network may dynamically become affiliated with administrative domains. Any security solution with a static configuration would not suffice. It is desirable for our security mechanisms to adapt on-the-fly to these changes. *Finally,* an adhoc network may consist of hundreds or even thousands of nodes. Security mechanisms should be scalable to handle such a large network.

These challenges motivate for building multi fence security solutions that achieve both broad protection and desirable network performance. In this paper our focus is on the fundamental security problem of protecting the multihop network connectivity between mobile nodes in a MANET. Efforts are to review the state-of-the-art security proposals that protect the MANET link layer and network layer operations of delivering packets over the multihop wireless channel. The complete security solution should span both layers, and encompass all three security components of prevention, detection, and reaction.

III. Related Work

MANETs are extremely vulnerable to attacks due to their dynamically changing topology, absence of conventional security infrastructures and open medium of communication, which, unlike their wired counterparts, cannot be secured with ease. MANET security involves authentication, key establishment and distribution, and encryption. Despite the fact that security of adhoc routing protocols is causing a major roadblock in commercial applications of this technology, only a limited work has been done in this area. Such efforts have mostly concentrated on the aspect of data forwarding, disregarding the aspect of topology discovery. On the other hand, solutions that target route discovery have been based on approaches for fixed-infrastructure networks, defying the particular adhoc network challenges. To address these concerns, several secure routing protocols have been studied. Dahill et al. proposed ARAN [3], it assumes managed-open environment, where there is a possibility for pre-deployment of infrastructure. It consists of two distinct stages. The first stage is the certification and end-to-end authentication stage. In this, source gets a certificate from the trusted certification server and then using this certificate signs the request packet. Each intermediate node in turn signs the request with its certificate. The destination then verifies each of the certificates, thus the source gets authenticated and so do the intermediate nodes. The destination node then sends the reply along the route reverse to the one in the request; reply signed using the certificate of the destination. The second stage is a non-mandatory stage which is used to discover the shortest path to the destination but this stage is

computationally expensive. It is prone to reply attacks using error messages unless the nodes have time synchronization. Papadimitratos and Haas [4] proposed a protocol SRP that can be applied to several existing routing protocols. This protocol assumes a security association between source and destination nodes. Intermediate nodes do not need to cryptographically validate the control traffic. It adds a SRP header to the base routing protocol, DSR or AODV, request packet. SRP header has three important fields QSEQ which helps prevent replay of old outdated requests, QID and random number which helps prevent fabrication of requests and a SRP MAC which ensures integrity of the packets in transit. SRP requires that, for every route discovery, source and destination must have a security association between them. Furthermore, the paper does not even mention route error messages. Therefore, they are not protected and any malicious node can just forge error messages with other nodes as source. ARIADNE [5], is based on DSR [6] and TESLA [7]. It prevents attackers/compromised nodes from disrupting uncompromised routes comprising of benign nodes. It uses highly efficient symmetric key cryptography. It does not guard against passive attackers eavesdropping on the network traffic. It does not prevent an attacker from inserting data packets. It is vulnerable to active-1-1 attacker that lies along the discovered route, who does not forward packets and does not generate ERROR if it encounters a broken link. It also requires clock synchronization, which is considered to be an unrealistic requirement for adhoc networks. Perlman proposed a link state routing protocol [8] that achieves Byzantine Robustness. Although the protocol is highly robust, it requires a very high overhead associated with public key encryption. Zhou and Haas [9] primarily discussed key management. They devote a section to secure routing, but essentially conclude that "nodes can protect routing information in the same way they protect data traffic". They also observe that denial-of-service attacks against routing will be treated as damage and routed around. Some work has been done to secure adhoc networks by using misbehavior detection schemes [10]. This approach has two main problems: first, it is quite likely that it will be not feasible to detect several kinds of misbehaving; and second has no real means to guarantee the integrity and authentication of the routing messages. Looking at the work that has been done in this area previously, it seems that the security needs for adhoc networks has not been yet satisfied. Most of the work done around using Hashing techniques is around authenticating messages and route table entries. Bayya et al. [11] demonstrate the use of hashing as part of password based authenticated key exchange. The problems in this protocol are the need of a strong shared secret and the need to constantly change the shared secret which in turn may prove to be computationally expensive. Yih-Chun Hu et al. [12] used symmetric cryptography to secure adhoc networks by using one way hash chains or Markle hash tree as part of SEAD protocol for proactive routing. In this protocol the elements of hash chain are used directly to authenticate the sequence number and other metric in each entry. The problems identified with SEAD protocol are no provision of a secure initial key distribution, greater network traffic and count-to-infinity problem. Zapata [13] in its proposed protocol uses a new one-way hash chain for each Route Discovery to secure the metric field in an RREQ packet. It also uses asymmetric cryptography to initially authenticate participating nodes.

IV. Cryptographic Hashing

A 'hash' ('digest' and informally a 'checksum') is a kind of 'signature' for a stream of data that represents the contents. Cryptographic hashing [14] is used for data/user verification and authentication. A strong cryptographic hash function has the property of being very difficult to reverse the result of the hash and hence reproduce the original piece of data. The hash functions are typically defined by the way they create hash values from data. There are two main methodologies for a hash algorithm to implement as:

a) Additive And Multiplicative Hashing

The hash value is constructed by traversing through the data and continually incrementing an initial value by a calculated value relative to an element within the data. The calculation done on the element value is usually in the form of a multiplication by a prime number [14] as given in equations (1) to (4).

$$h(m) = h^{-1} \oplus (m \otimes p) \quad \text{----- (1)}$$

$$h(m) = \sum_{i=0}^{|m|} m_i \otimes p_i \quad \text{----- (2)}$$

$$h(m) = h^{-1} \otimes (m \otimes p) \quad \text{----- (3)}$$

$$h(m) = \prod_{i=0}^{|m|} m_i \otimes p_i \quad \text{----- (4)}$$

b) Rotative Hashing

This is same as additive hashing. In that every element in the data string is used to construct the hash but unlike additive hashing, the values are put through a process of bitwise shifting [14]. Usually a combination of both left and right shifts is used (shift amounts are prime). The result of each process is added to some form of accumulating count, the final result being the hash value is passed back as the final accumulation as given in equations (5) to (7).

$$h(m) = h^{-1} \oplus (m \ll p) \otimes (m \gg q) \quad \text{----- (5)}$$

$$h(m) = \sum_{i=0}^{|m|} (m_i \ll p_i) \otimes (m_i \gg q_i) \quad \text{----- (6)}$$

$$h(m) = \prod_{i=0}^{|m|} (m_i \ll p_i) \otimes (m_i \gg q_i) \quad \text{----- (7)}$$

The popular examples of hashing functions [11] are HMAC, SHA-1 and MD5.

V. Proposed Stable and Secured Routing Protcol

In AODV protocol, it is assumed that the malicious node has exceptionally large sequence number. Whenever a malicious node joins the network, the packets start dropping and link path breakage happens as shown in figure 1. The proposed scheme is based on modifications of existing AODV. Two parameters have been used, one for stability and other for security. The proposed protocol, SSRP, ensures stable and secure routing over the adhoc network.

Whenever a link path breakage occurs, stable routing is achieved with an alternate route selection using neighbour nodes. If there is an attack on security of the network, the hash key chain mechanism is used to ensure secure routing. This has been shown in figure 2.

The solution proposed stresses upon applying hashing techniques not only in prevention stage in the form of message and routing information authentication, but also in different stages of securing adhoc networks. A unique way of using hash functions as 'one way hash chain' has been used in the proposed work. Hash key chains are constructed by using only symmetric cryptographic primitives, namely hash functions. A hash key chain is configured as a recursive chain, where the node first chooses a random key, K_1. Subsequent keys [15, 16, 17] are calculated by calculating the one-way hash over the key as given in equation (8):

$$K_2=H[K_1],\ K_3=H[K_2],\ \ldots\ldots\ldots,\ K_N=[K_{N-1}] \quad \text{----- (8)}$$

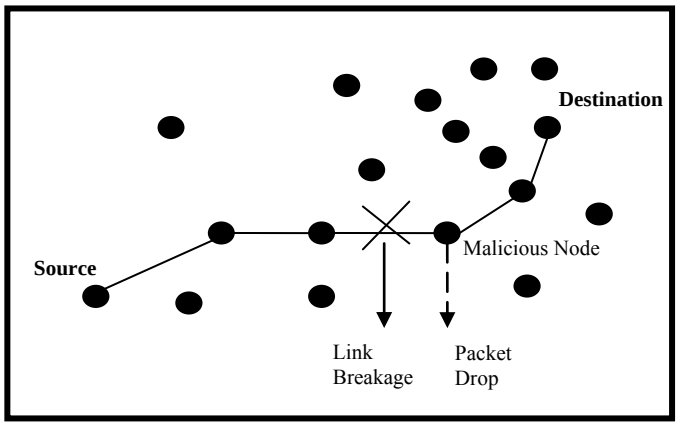

Figure 1: Route prior to malicious node entry in AODV

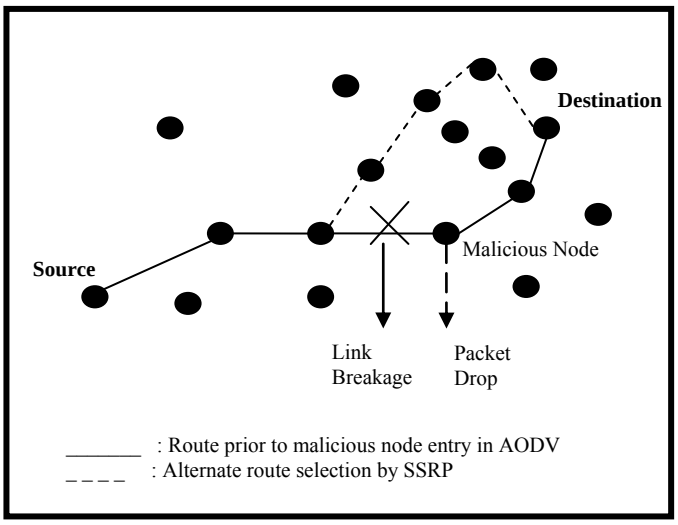

Figure 2 : Route recovery using alternate path in SSRP

The algorithm behind the proposed protocol is given below with the following assumptions:

Assumptions:

a) The source node 'S' and destination node 'D' are not malicious.
b) The sequence number is in ascending order from 'S' to 'D'.
c) One hop has increment of one sequence number.
d) Total hops can be counted before route selection.

Step 1: The sequence number of each and every node is updated at each beacon.

Step 2 : Whenever a node with exceptionally large sequence number is detected, it means malicious node has entered into the network.

Step 3 : If a malicious node has entered into the network during transmission of data from source to destination, an alternate route is selected.

Step 4 : The secure routing is ensured using equation 8 which is used by any node to authenticate any received value on the hash chain. If the computed value matches previous known authentic key value then the received key is authentic.

Step 5 : Each node discloses each key of its one-way key chain in a particular order, which is exactly reverse of the order in which the keys were generated. The key disclosure schedule and key generation schedule should be reverse For example if the keys were generated by a node in the order K_N; K_{N-1}; …..K_1; K_0 then the node disclose them in the order K_0; K_1; …. K_N. The rationale behind having the key disclosure schedule to be reverse of the key generation schedule is that K_N of a node is known to all other nodes and in such a situation they should be able to authenticate any subsequent keys that are disclosed. The use of one way hash function allows K_0; K_1 ; …. K_{N-1} to be authenticated using K_N but K_N cannot be authenticated

using any other key value. Hence the key disclosure schedule and key generation schedule is reverse.

Step 6: The transmission of route request and route reply is analyzed as under:

Transmission of RREQ packet

A parameter 'node_type' is used which is set to '0' (non-malicious) or '1' (malicious). RREQ packet is broadcasted by Source. If non-malicious, broadcast RREQ to current node else deactivate the node and don't broadcast RREQ. Process continues till it reaches Destination.

Transmission of RREP packet

Destination node 'D' rebroadcast the RREP packet like the RREQ. All the possible routes will be searched by RREP. If any node is out of signal range or dead from the network after getting RREQ then available route will be selected by the RREP broadcasting.

VI. Performance Metrics

RFC 2501 describes a number of quantitative metrics that can be used for evaluating the performance of a routing protocol for mobile wireless ad-hoc networks. Some of these quantitative metrics [18] are defined as follow:

a) Packet Delivery Fraction (PDF)

The packet delivery fraction is defined as the ratio of number of data packets received at the destinations over the number of data packets sent by the sources as given in equation (9). This performance metric is used to determine the efficiency and accuracy of MANET's routing protocols.

$$\text{Packet Delivery Fraction} = \frac{\text{Total Data Packets Received}}{\text{Total Data Packets Sent}} \times 100 \quad \text{----- (9)}$$

b) Average End_to_End Delay (AE2ED)

This is the average time involved in delivery of data packets from the source node to the destination node. To compute the average end-to-end delay, add every delay for each successful data packet delivery and divide that sum by the number of successfully received data packets as given in equation (10). This metric is important in delay sensitive applications such as video and voice transmission.

$$\text{Average End to End Delay} = \frac{\sum (\text{Time Received} - \text{Time Sent})}{\text{Total Data Packets Received}} \quad \text{----- (10)}$$

c) Network Throughput

A network throughput is the average rate at which message is successfully delivered between a destination node (receiver) and source node (sender). It is also referred to as the ratio of the amount of data received from its sender to the time the last packet reaches its destination. Throughput can be measured as bits per second (bps), packets per second or packet per time slot. For a network, it is required that the throughput is at high-level. Some factors that affect MANET's throughput are unreliable communication, changes in topology, limited energy and bandwidth.

d) Normalized Routing Load (NRL)

The normalized routing load is defined as the fraction of all routing control packets sent by all nodes over the number of received data packets at the destination nodes. In other words, it is the ratio between the total numbers of routing packets sent over the network to the total number of data packets received as given in equation (11). This metric discloses how efficient the routing protocol is. Proactive protocols are expected to have a higher normalized routing load than reactive ones. The bigger this fraction is the less efficient the protocol.

$$\text{Normalized Routing Load} = \frac{\text{Total Routing Packets Sent}}{\text{Total Data Packets Received}} \quad \text{----- (11)}$$

e) Packet Loss (PL)

Packet loss occurs when one or more packets being transmitted across the network fail to arrive at the destination. It is defined as the number of packets dropped by the routers during transmission. It can be shown by equations (12) to (14).

$$\text{Packet Loss} = \text{Total Data Packets Dropped} \quad \text{----- (12)}$$

$$\text{Packet Loss} = \text{Total Data Packets Sent} - \text{Total Data Packets Received} \quad \text{----- (13)}$$

$$\text{Packet Loss (\%age)} = \frac{\text{Total Packets Dropped}}{\text{Total Data Packets Sent}} \times 100 \quad \text{----- (14)}$$

In this research paper, performance of the proposed protocol 'SSRP' is evaluated with respect to 'AODV' using these performance metrics.

VII. Experimental Analysis Using Simulation

IEEE 802.11 is used as the MAC layer protocol. The simulation experiments are carried over network simulator 2 (version 2.34) installed in Fedora Linux 12. The results have been derived by writing a *tcl* script and generating corresponding *trace* and *nam* files. Varying number of UDP connections/traffic agents have been used to analyze the traffic. The mobility model used is random waypoint model in a square area. The area configurations used are 750 meter x 750 meter for 20 nodes, 1000 meter x 1000 meter for 50 nodes and 1500 meter x 1500 meter for 80 nodes. The packet size is 512 bytes. The simulation run time is 500 seconds during analysis of 20 nodes, 700 seconds for 50 nodes and 950 seconds for 80 nodes.

a) Snapshots of Simulation Environment

An extensive simulation model having scenario of 20, 50 and 80 mobile nodes is used to study inter-

layer interactions and their performance implications. Same scenario has been used for performance evaluation of both SSRP and AODV protocols at one time. Some of the snapshots of trace and NAM files created using AODV and SSRP protocols for 50 nodes are shown in figure 3 to 6.

```
+ -t 3.000000000 -s 0 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k RTR
- -t 3.000000000 -s 0 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k RTR
h -t 3.000000000 -s 0 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k RTR
+ -t 3.000115000 -s 0 -d -1 -p AODV -e 106 -c 2 -a 0 -i 0 -k MAC
- -t 3.000115000 -s 0 -d -1 -p AODV -e 106 -c 2 -a 0 -i 0 -k MAC
h -t 3.000115000 -s 0 -d -1 -p AODV -e 106 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963291 -s 9 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963308 -s 40 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963329 -s 20 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963360 -s 10 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963394 -s 30 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963397 -s 11 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963397 -s 21 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963441 -s 4 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963467 -s 1 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963504 -s 15 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963534 -s 19 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963558 -s 24 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963641 -s 7 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963646 -s 35 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963738 -s 43 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963741 -s 2 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963760 -s 37 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963763 -s 8 -d -1 -p AODV -e 48 -c 2 -a 0 -i 0 -k MAC
```

Figure 3 : NAM File using AODV (50 Nodes)

```
s 3.000000000 _0_ AGT  --- 0 cbr 512 [0 0 0 0] ------- [0:0 1:0 32 0] [0] 0 0
r 3.000000000 _0_ RTR  --- 0 cbr 512 [0 0 0 0] ------- [0:0 1:0 32 0] [0] 0 0
s 3.000000000 _0_ RTR  --- 0 AODV 48 [0 0 0 0] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
s 3.000115000 _0_ MAC  --- 0 AODV 106 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963291 _9_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963308 _40_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963329 _20_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963360 _10_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963394 _30_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963397 _11_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963397 _21_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963441 _4_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963467 _1_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963504 _15_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963534 _19_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963558 _24_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963641 _7_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963646 _35_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963738 _43_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963741 _2_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963760 _37_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963763 _8_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963776 _49_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963787 _13_ MAC  --- 0 AODV 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
```

Figure 4 : Trace File using AODV (50 nodes)

```
+ -t 3.000115000 -s 0 -d -1 -p SSRP -e 106 -c 2 -a 0 -i 0 -k MAC
- -t 3.000115000 -s 0 -d -1 -p SSRP -e 106 -c 2 -a 0 -i 0 -k MAC
h -t 3.000115000 -s 0 -d -1 -p SSRP -e 106 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963291 -s 9 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963308 -s 40 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963329 -s 20 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963360 -s 10 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963394 -s 30 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963397 -s 11 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963397 -s 21 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963441 -s 4 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963467 -s 1 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963504 -s 15 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963534 -s 19 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963558 -s 24 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963641 -s 7 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963646 -s 35 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963738 -s 43 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963741 -s 2 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963760 -s 37 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963763 -s 8 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
r -t 3.000963776 -s 49 -d -1 -p SSRP -e 48 -c 2 -a 0 -i 0 -k MAC
```

Figure 5 : NAM File using SSRP (50 Nodes)

```
r 3.000963291 _9_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963308 _40_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963329 _20_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963360 _10_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963394 _30_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963397 _11_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963397 _21_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963441 _4_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963467 _1_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963504 _15_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963534 _19_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963558 _24_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963641 _7_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963646 _35_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963738 _43_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963741 _2_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963760 _37_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963763 _8_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963776 _49_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963787 _13_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963817 _3_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000963820 _6_ MAC  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
r 3.000988291 _9_ RTR  --- 0 SSRP 48 [0 ffffffff 0 800] ------- [0:255 -1:255 30 0] [0x2 1 1 [1 0] [0 4]] (REQUEST)
```

Figure 6 : Trace File using SSRP (50 Nodes)

A graphical tool known as Network Animator is used to observe the visual representation of NAM files created during simulation of 50 nodes. The snapshots of visual representations taken at two different times t_1= 138.942153 Sec. and t_2= 138.974673 Sec. are given in figure 7 and 8.

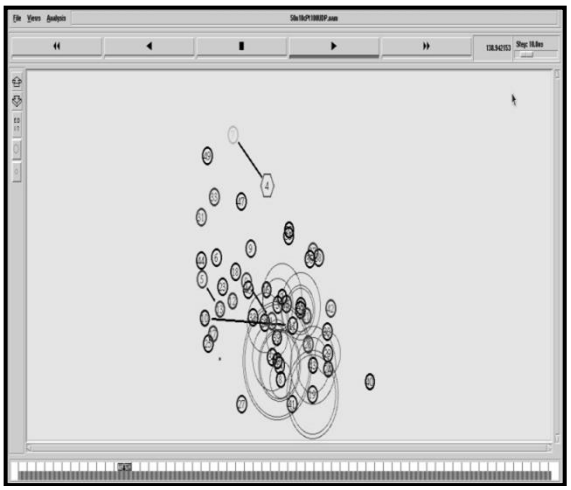

Figure 7 : Position at time t_1= 138.942153 Seconds (50 Nodes)

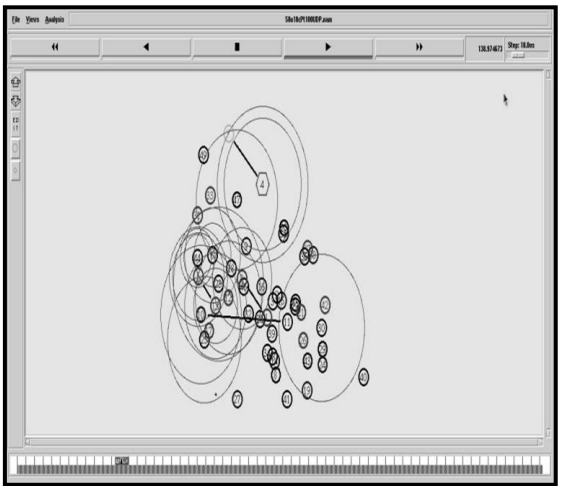

Figure 8 : Position at time t_2= 138.974673 Seconds (50 Nodes)

b) Simulation Results for 20 Nodes

All the performance metrics have been evaluated for SSRP and AODV protocols using 6 UDP connections. All nodes are moving at a fixed speed of 5 meters/second. Two malicious nodes have been introduced in the network scenarios which are moving at a speed of 1 meter/second. The pause time has been used as a varying parameter from 100 seconds to 500 seconds and the queue length is 150.

Figure 9 shows packet delivery fraction with respect to pause time. The observation is that SSRP gives high packet delivery fraction that AODV and there is a significant positive difference. Therefore, SSRP protocol outperforms AODV in terms of stable and secure routing over MANET. In figure 10, average end to end delay has been presented with respect to pause time. When the pause time is 100 seconds, AODV has high average end to end delay than SSRP but after that AODV and SSRP gives almost same results. On an average, SSRP outperforms AODV. The network throughput with respect to pause time has been shown in figure 11. The protocol having high network throughput is more efficient and in this figure, SSRP gives high throughput than AODV. Therefore, SSRP outperforms AODV in terms of throughput. Figure 12 shows normalized routing load by varying pause time. The bigger this fraction is the less efficient the routing protocol. When the pause time is between 100 seconds to 300 seconds, AODV shows bigger NRL than SSRP but after that both SSRP and AODV gives almost same results. On an average, SSRP outperforms AODV in terms of normalized routing load. In figure 13, the packet loss has been shown for both protocols. Higher the packet loss, less efficient is routing protocol and in this figure, AODV gives high packet loss than AODV. Therefore, SSRP outperforms than AODV in terms of packet loss.

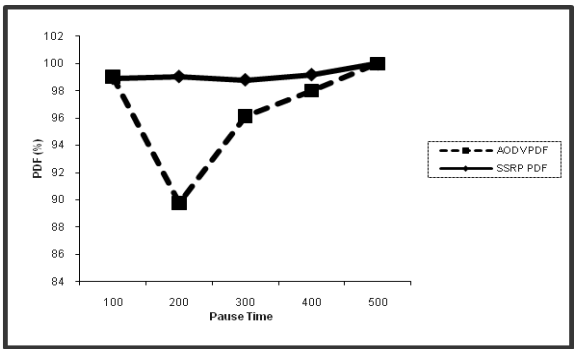

Figure 9 : Packet Delivery Fraction (20 Nodes)

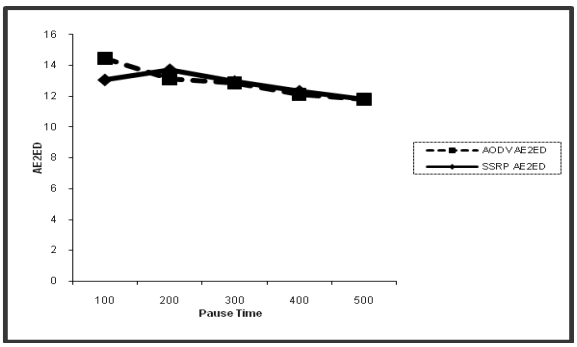

Figure 10 : Average End to End Delay (20 Nodes)

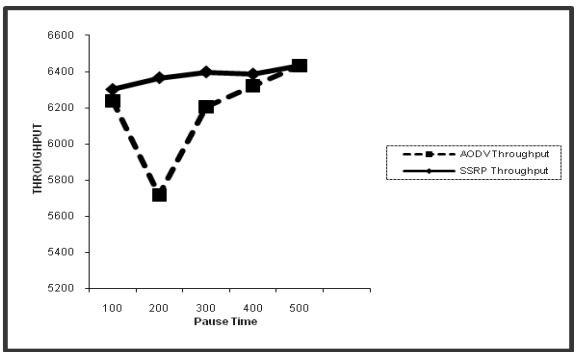

Figure 11: Network Throughput (20 Nodes)

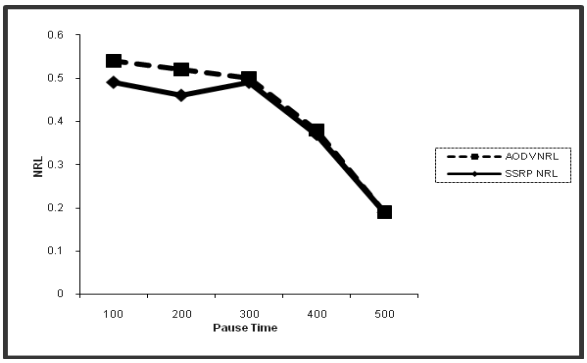

Figure 12 : Normalized Routing Load (20 Nodes)

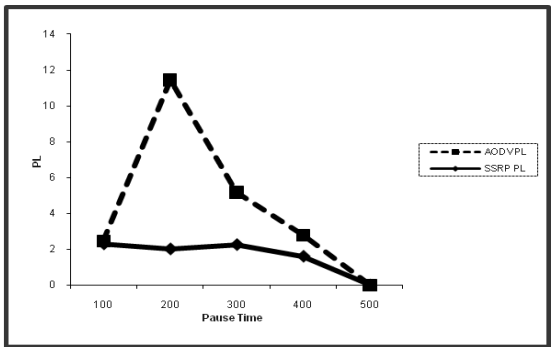

Figure 13 : Packet Loss (20 Nodes)

c) Simulation Results for 50 Nodes

All the performance metrics have been evaluated for SSRP and AODV protocols using 10 UDP connections. All nodes are moving at a fixed speed of 10 meters/second. Two malicious nodes have been introduced in the network scenarios which are moving at a speed of 5 meters/second. The pause time has been used as a varying parameter from 100 seconds to 700 seconds and the queue length is 150.

In figure 14, packet delivery fraction is shown with respect to pause time for SSRP and AODV. The observation is that SSRP gives high packet delivery fraction that AODV and therefore, SSRP protocol outperforms AODV in terms of better packet delivery.

In figure 15, average end to end delay has been presented with respect to pause time. When the pause time is between 100 seconds to 200 seconds, AODV has high average end to end delay than SSRP. When pause time is between 200 seconds to 400 seconds, SSRP has high average end to end delay than AODV. In end, when pause time is between 400 seconds to 500 seconds, SSRP has low average end to end delay than AODV. Therefore, on an average, SSRP outperforms AODV.

Network throughput with respect to pause time has been shown in figure 16. SSRP gives high throughput than AODV and therefore, SSRP outperforms AODV in terms of throughput.

Figure 17 shows normalized routing load by varying pause time. When the pause time is between 100 seconds to 300 seconds, AODV shows higher normalized routing load than SSRP but when the pause time is between 300 seconds to 400 seconds, SSRP gives higher normalized routing load than SSRP. In end, both SSRP and AODV give almost same results. Concluding, it is inferred that SSRP outperforms AODV in terms of normalized routing load.

In figure 18, AODV shows high packet loss than SSRP and therefore, SSRP outperforms than AODV.

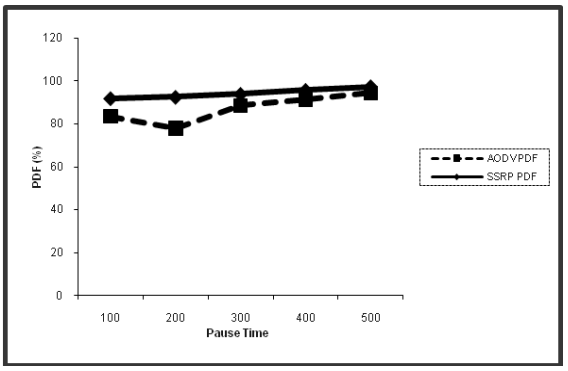

Figure 14 : Packet Delivery Fraction (50 Nodes)

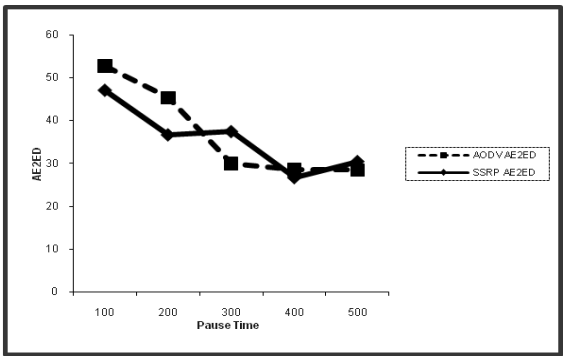

Figure 15 : Average END To End Delay (50 Nodes)

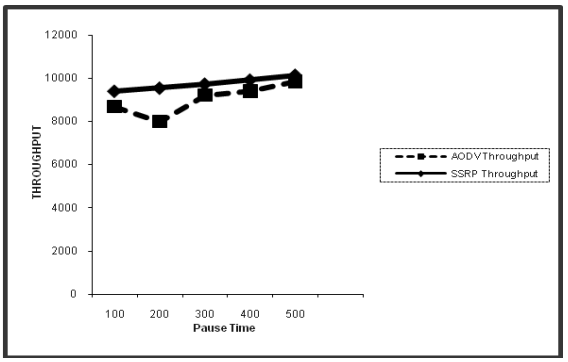

Figure 16 : Network Throughput (50 Nodes)

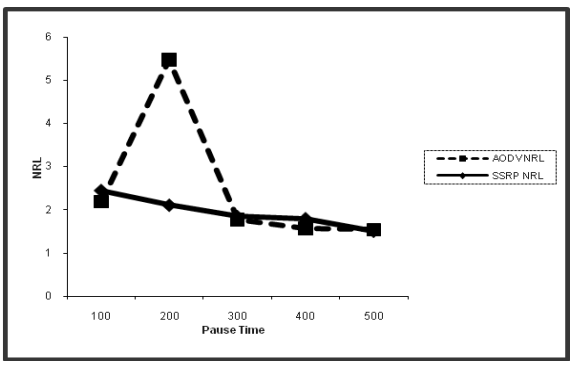

Figure 17 : Normalized Routing Load (50 Nodes)

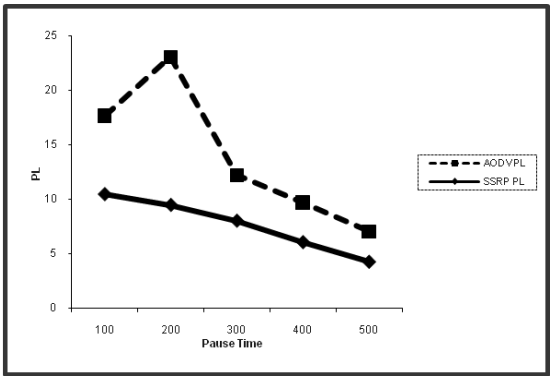

Figure 18 : Packet Loss (50 Nodes)

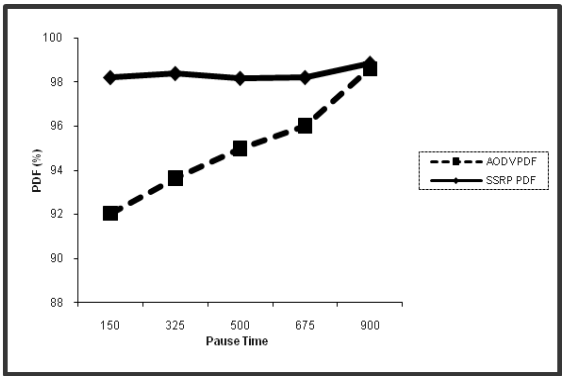

Figure 19 : Packet Delivery Fraction (80 Nodes)

d) Simulation Results for 80 Nodes

All the performance metrics have been evaluated for SSRP and AODV protocols using 14 UDP connections. All nodes are moving at a fixed speed of 10 meters/second. Two malicious nodes have been introduced in the network scenarios which are moving at a speed of 5 meters/second. The pause time has been used as a varying parameter from 100 seconds to 950 seconds and the queue length is 150.

Figure 19 shows that packet delivery fraction for SSRP is much higher than that of AODV for all pause times and hence SSRP outperforms AODV in terms of better packet delivery. In figure 20, average end to end delay has been presented with respect to pause time. When the pause time is between 100 seconds to 675 seconds, AODV has high average end to end delay than SSRP but when it is between 675 seconds to 950 seconds, SSRP gives high average end to end delay than AODV. Concluding SSRP outperforms AODV initially but in end AODV starts outperforming SSRP. This issue is still under consideration. Network throughput with respect to pause time has been shown in figure 21. SSRP gives high throughput than AODV for all pause times and hence SSRP outperforms AODV in terms of better throughput.

Figure 22 shows normalized routing load by varying pause time. The bigger this fraction is the less efficient the routing protocol. When the pause time is between 100 seconds to 250 seconds, SSRP shows bigger NRL than AODV; when it is between 250 seconds to 400 seconds, AODV shows bigger NRL than SSRP and when pause time is between 400 seconds to 950 seconds, SSRP shows marginal bigger NRL than AODV. Although both the protocols give almost same results but still due to marginal difference between the results, on an average, AODV outperforms SSRP. In figure 23, the packet loss has been shown for both protocols with respect to varying pause time from 100 seconds to 950 seconds. In all cases, SSRP gives very low packet loss than AODV and so SSRP outperforms AODV.

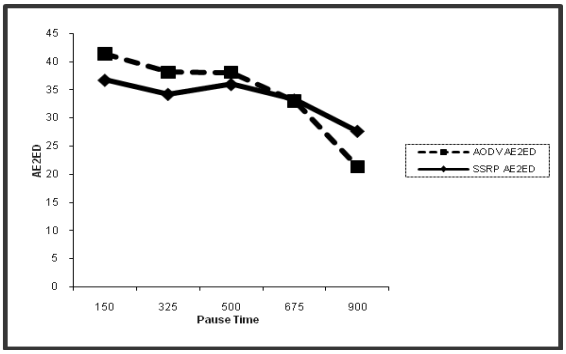

Figure 20 : Average End to End Delay (80 Nodes)

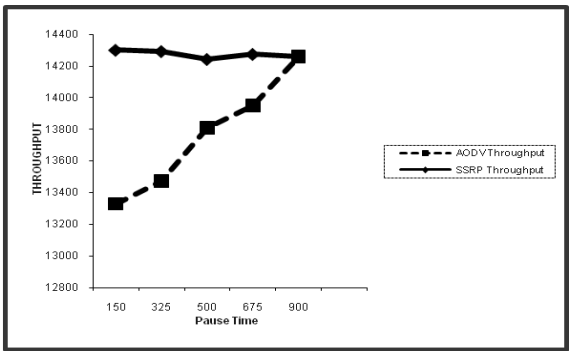

Figure 21: Network Throughput (80 Nodes)

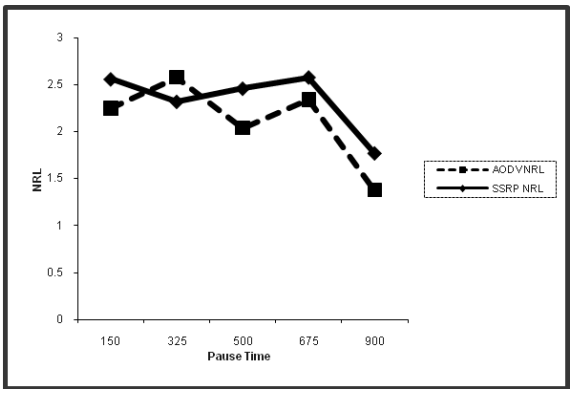

Figure 22 : Normalized Routing Load (80 Nodes)

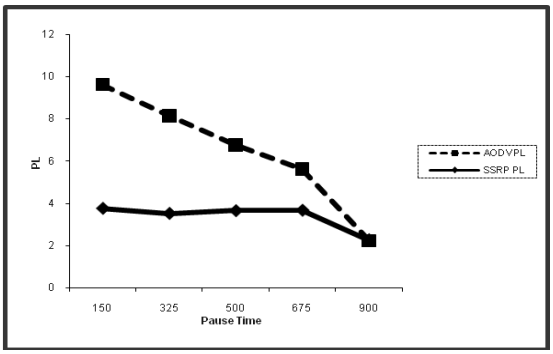

Figure 23 : Packet Loss (80 Nodes)

VIII. Conclusion and Future Scope

The existing routing protocols are typically attack-oriented. They first identify the security threats and then enhance the existing protocol to conquer such attacks. Since the solutions are designed explicitly with certain attack models in mind, they work well in the presence of designated attacks but may collapse under unanticipated attacks. Therefore, the ultimate goal for adhoc network security is to develop a multifold security solution that results in in-depth protection that offers multiple lines of defense against both known and unknown security threats. AODV is vulnerable to various kinds of attacks as it based on the assumption that all nodes must cooperate and without their cooperation no route can be established. In addition, when the malicious nodes enter into the network, various performance metrics begin decreasing for AODV. The objective of this research paper is to find a multifold security solution by developing a new on-demand stable and secure routing protocol, SSRP. The performance of this protocol has been evaluated with respect to AODV using five performance metrics viz. packet delivery fraction, average end to end delay, network throughput, normalized routing load and packet loss. It has been concluded that when the malicious nodes come into the way, the performance of SSRP is much better than that of AODV. Efforts are to increase the number of mobile nodes and to introduce more malicious nodes in the network scenario so that its impact on the network performance may be determined. The efforts can be made in the direction of improving hash functions to avoid collisions, using stronger hash keys by making them dependent on additional parameters like biometric credentials, passwords, IP addresses etc.

References Références Referencias

1. T. Karygiannis and L. Owens, "Wireless Network Security", *NIST Special Publication*, 2002.
2. William Stallings, "Cryptography and Network Security: Principles and Practice", *Prentice Hall*, 5th Edition, 2011.
3. B. Dahill, B. N. Levine, E. Royer and C. Shields, "A secure routing protocol for adhoc networks", *Technical Report UM-CS-2001-037*, University of Massachusetts, Department of Computer Science, 2001.
4. P. Papadimitratos and Z. J. Haas, "Secure Routing for Mobile Adhoc Networks", *SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002)*, 2002.
5. Adrian Perrig, D. B. Johnson, Yih-Chun Hu, "ARIADNE: A Secure On-demand Routing Protocol for Adhoc Networks", *ACM, Proceedings of the Eighth Annual International Conference on Mobile Computing and Networking (MobiCom 2002),* 2002.
6. D. B. Johnson, D. A. Maltz, Y.C. Hu, "The Dynamic Source Routing Protocol for Mobile Adhoc Networks", *Internet Draft*, MANET working group, 2003.
7. Perrig, R. Canetti, D. Song and D. Tygar, "Efficient and Secure Source Authentication for Multicast", *In Network and Distributed System Security Symposium (NDSS'01)*, 2001.
8. R. Perlman, "Fault-tolerant Broadcast of Routing Information", Computer Networks, No. 7, pp. 395–405.
9. L. Zhou and Z. J. Haas, "Securing Adhoc Networks", *IEEE Network Magazine*, 13(6), pp. 24–30, 1999.
10. S.Marti, T. J. Giuli, K. Lai and M. Baker, "Mitigating Routing Misbehavior in Mobile Adhoc Networks", *Proceedings of the Sixth Annual International Conference on Mobile Computing and Networking,* pp. 255–265, 2000.
11. Bayya Arun, "Security in Ad-hoc Networks", Computer Science Department, University of Kentucky.
12. Yih-Chun Hu, D. B. Johnson, and A. Perrig, "SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Adhoc Networks," *IEEE Workshop on Mobile Computing Systems and Applications (WMCSA)*, pp. 3-13, 2002.
13. Manel Guerrero Zapata, N. Asokan, "Securing Adhoc Routing Protocol", *WiSe 2002*.
14. Arash Partow, "General Purpose Hash Function Algorithms", www.partow.net.
15. Kush A., Hwang C., "Proposed Protocol for Hash-Secured Routing Adhoc Networks", *Masaum Journal Of Computing (MJC)*, Volume 1, Issue 2, pp. 221-226, 2009.
16. Kush A., Gupta P., Hwang C., "Secured Routing Scheme for Adhoc Networks", *International Journal of Computer Theory and Engineering (IJCTE)*, Volume 3. pp. 1793-1799, 2009.
17. L. Lamport, "Password Authentication with Insecure Communication", *Comm. of ACM*, 24 (11), pp. 770-772, 1981.

18. Georgios Kioumourtzis, "Simulation and Evaluation of Routing Protocols for Mobile Adhoc Networks", Thesis, Master of Science in Systems Engineering and Master of Science in Computer Science, *Naval Postgraduate School*, Monterey, California, 2005.

This page is intentionally left blank