Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X
IMPACT FACTOR: 5.258

IJCSMC, Vol. 5, Issue. 7, July 2016, pg.92 – 99

Path Aware Secure Routing for MANETs

Sunetra P. Salunkhe[1], Dr. Hitendra D. Patil[2]

[1]Master Student, Computer Engineering, SSVPS's B.S.Deore College of Engineering, India
[2]Professor and Head, Computer Engineering, SSVPS's B.S.Deore College of Engineering, India
[1] salunkhesunetra.p@gmail.com; [2] hitendradpatil@gmail.com

Abstract— Mobile ad hoc network (MANET) is wireless connection of mobile nodes which provides the communication and mobility among wireless nodes without the need of any physical infrastructure or centralized devices. The communications in MANET is done by routing protocols. At present MANET is used in many real time applications and hence such networks are vulnerable to different kinds of security threats. MANET networks suffered more from security attacks due to use of free wireless communication frequency spectrum and dynamic topology. Therefore it becomes very tough to provide secure to MANET under different adversarial environments like battlefields. For MANET, anonymous communications are vital under the adversarial environments, in which the identification of nodes as well as routes is replaced by pseudonyms or random numbers for the purpose of protection. There are many protocols presented for anonymous communication security for MANET, however suffered from limitations like worst delay, vulnerable to DoS attacks etc. This paper presents Delay Efficient Authenticated Anonymous Secure Routing [DEAASR] which is extension of existing AASR approach presented recently. The main aim of DEAASR protocol is provide secure data communication with goal of improving performance packet delay and routing efficiency for different attacks in MANET.

Keywords— MANET, Security, Anonymous Routing, Path aware routing

I. INTRODUCTION

Mobile Ad hoc Network (MANET) is an autonomous system of mobile nodes connected via wireless links without using any pre-established network infrastructure. Every node in the network acts as both a host and a router and forwards each other's packets to enable the communication between nodes, not directly connected by wireless links. The development of dynamic routing protocols that can efficiently find routes between the communicating nodes is the central challenge in the ad hoc network design. The routing protocol must keep the high degree of node mobility that often changes the network topology. It is difficult to provide trusted and secured communications in adversarial network. The nodes inside a network are not always trusted because a node within a network may become malicious. The adversaries outside a network may deduce the information about the communicating nodes or traffic flows by passive snooping. A secured routing protocol should be provided whenever nodes want to communicate with each other. End-to-end security mechanisms can provide some level of security for the data, valuable information such as identity and traffic of the communicating nodes may be easily determined from data analysis. An anonymous routing based technique should be modified to provide anonymity and to overcome attacks. Anonymity is a combination of unidentifiability and unlinkability. Unidentifiability indicates that the identities of the source and destination nodes should not be revealed to the other nodes in the network. Unlinkability indicates that the route and traffic flows between the nodes cannot be uncovered to the network.

Related Work

Various methods to deal with the anonymity for MANETs have been proposed.

J. Kong, X. Hong, and M. Gerla, proposed An identity-free and on-demand routing scheme against anonymity threats in mobile ad hoc networks (ANODR) [6]. ANODR works on the mechanism of broadcast and trapdoor information. The drawback of this approach is that every forwarding node in the path has to generate a new public/secret key pair for every RREQ message. The cost of generating key pairs increases due to overhead. A. Boukerche, K. El-Khatib, L. Xu, and L. Korba proposed An efficient secure distributed anonymous routing protocol for mobile and wireless ad hoc networks (SDAR) [5]. SDAR permits only the reliable nodes to participate in transmission. This message achieves the destination and it gets decrypted in each intermediate node and achieves the source. Source node obtains complete information about the intermediate nodes. Neighborhood nodes IDs are potentially uncovered. SDAR is not secured against Denial of Service attack. Messages are vast and rely on the quantity of bounces. R. Song, L. Korba, and G. Yee proposed AnonDSR: efficient anonymous source routing for mobile ad hoc networks [7]. In AnonDSR anonymous route establishment relies upon the quantity of jumps between the source and the destination, but it allows the destination nodes to know all intermediate node IDs. Y. Zhang, W. Lou, and Y. G. Fang proposed MASK: Anonymous On-Demand Routing in Mobile Ad Hoc Networks [8]. MASK provides anonymity depending on a unique sort of open key cryptosystem, the pairing-based cryptosystem, to accomplish unknown correspondence in MANET but it fails at the destination nodes because the destination node ID is present in every RREQ message in plain text. L. Yang, M. Jakobsson, and S. Wetzel proposed Discount anonymous on demand routing for mobile ad hoc networks [9]. It has the advantage of accomplishing considerably lower computation and correspondence complexities at the cost of expense of a slight lessening of security insurances. Route requests in Discount-ANODR and in ANODR are parallel but the limitation is that intermediate nodes know the destination of the request and the identity of the previous intermediate node but not the source node. J. Paik, B. Kim, and D. Lee proposed A3RP: Anonymous and Authenticated Ad hoc Routing Protocol [10] which provides security to data packets by group signature but the A3RP used secure hash function to calculate the anonymous route using the real IDs of the destination node but it is not scalable as encrypted onion mechanism.

The existing protocols are vulnerable to the denial-of-service (DoS) attacks. The node IDs are exposed, which do not provide anonymity to the nodes in the adversarial network. Generating new pair of public/private key for each node makes the operation expensive. To overcome the problems associated with existing method recently AASR method was proposed by Wei Liu and Ming Yu, in "AASR: Authenticated Anonymous Secure Routing for MANETs in Adversarial Environments". MANETs in adversarial environments, the public and group key can be initially deployed in the mobile nodes. It was assumed that there is no online security or localization service available when the network is deployed. Therefore authenticated anonymous secure routing (AASR) is proposed to overcome the pre-mentioned problems. AASR method adopts a key-encrypted onion to record a discovered route and design an encrypted secret message to verify the RREQ-RREP linkage. Group signature is used to authenticate the RREQ packet per hop, to prevent intermediate nodes from modifying the routing packet. AASR is suffering from the worst delay performance and that is the main research problem.

II. PROPOSED WORK

Here node model and protocol design is given.

A. Node representation

The information of the destination node is stored in destination table. This neighbor pseudonym and session key information is stored in the neighborhood table. A new entry is created in the routing table for each generated or forwarded route request message to other node. The switching information of the established routes are stored in the forwarding table. Hop comparison array is maintained at each node of current communication path. Each packet carries a "hop count (HC)" field in its header. HC is initialised to zero at source node and gets incremented by one at every hop. Format of hop comparison array is <Src, Dest, HC, Neigh>, where Src is source address, Dest is destination address and Neigh is neighbour's address from which packet was broadcast.

B. Protocol Design

An on-demand routing protocol is modified to design Path Aware Delay Efficient Authenticated Anonymous Secure Routing (DEAASR) which includes route discovery, data transmission and route maintenance. In route discovery, RREQ packet is broadcasted by source node to all the nodes those are within range. When destination receives RREQ packet, it will reply a RREP back along the same incoming path of the RREQ. A five node network is considered in Fig. 1 as an example to show the anonymous routing. Here A is the source node, E is the destination node, and B, C and D are intermediate nodes.

Dest. pseudonym	Dest. string	Dest. public key	Session key
N_E	dest	K_{E+}	K_{AE}

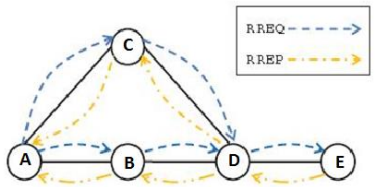

Fig 1: Example Network Topology

1) Anonymous Route Request:

Source Node: The assumption is that the source node A knows the information about the destination node. If there is no valid session key K_{AE} between A and E, then A will create a new session key for communication between source and destination. The following data will be updated in the forwarding table.

Source A will assemble and broadcast the RREQ packet in the given format:

A:- [RREQ, N_{sq}, V_E, V_{AE}, Onion (A)] G_{A-}

Where, RREQ: Packet type identifier; N_{sq}: Sequence number generated randomly by source node A for this RREQ packet; V_{AE}: An encrypted message to verify the route at intermediate nodes; V_E: An encrypted message to verify the destination at the destination node; Onion (A): key encrypted onion of node A; G_{A-}: The group private key of A to sign the RREQ packet.

The V_E and V_{AE} works as the trapdoor function. V_{AE} is defined as the following:

$$V_{AE} = (N_v)K_v$$

Where, N_v: one time randomly generated nonce; K_v : symmetric key used to encrypt N_v.

The encrypted message V_E is defined as:

$$V_E = \langle Nv, Kv, dest\rangle K_{AE,} \{K_{AE}\}K_{E+}$$

If E is the destination then it can decrypt K_{AE} using K_{E-}. Then it can decrypt the first part of the message using the K_{AE}.

Onion (A) created by A contains the following:

$$Onion\ (A) = O_{Kv}\ (N_A)$$

Where, N_A is the one time nonce created by A, K_v is the key used to encrypt this nonce. It can be decrypted only by E. After sending the RREQ, a new entry is created in the routing table of A:

Req pseudonym	Dest. pseudonym	Ver. Msg.	Next hop	Status
N_{sq}	N_E	V_{AE}	N/A	Pending

Intermediate Node: The RREQ packet of A is broadcasted into the network. It is assumed that B has neighbour relationships with A and D. B knows where the RREQ packet comes from. Node B will store the following entries in its neighbouring table:

Neigh. pseudonym	Session key
N_A	K_{AB}
N_D	K_{BD}

After receiving the RREQ packet, B first verify the packet using its group private key. If the packet is signed by a valid node then the RREQ will be processed else it will be discarded and will be dropped as malicious.

The N_{sq} and the time stamp are checked in order to determine if the packet has been processed earlier. If the N_{sq} is a new entry to the routing table then it is a new RREQ. If the N_{sq} is already present in the routing table with an old time stamp then it is a repeated request. If the N_{sq} exists with a new time stamp then it is an attack. B first tries to decrypt V_E with its private key. On unsuccessful decryption, B understands that it is not an intended destination; it will assemble and broadcast another RREQ packet in the following format:

B:- [RREQ, N_{sq}, V_{AE}, V_E, Onion (B)] G_{B-}

Where, N_{sq}, V_E and V_{AE} are kept same as received from A. It only updates the key-encrypted onion part and it be signed by B's private group key G_{B-}. The updated onion will be like:

$$\text{Onion (B)} = O_{KAB} (N_B, \text{Onion (A)})$$

Where, N_B is the one time nonce created by B to indicate itself.

When the RREQ of B reaches to D, D will perform same action, it will update the onion by adding one layer.

$$\text{Onion (D)} = O_{KBD} (N_D, \text{Onion (D)})$$

The routing tables of B and D will be updated with the new entry of the RREQ packet.

Req. pseudonym	Dest. Pseudonym	Ver. Msg.	Next hop	Status
N_{sq}	N/A	V_{AE}	N/A	Pending

Destination Node: When E receives the RREQ packet, it validates the packet similarly like B and D. E will decrypt the destination message, i.e. V_E, E understands that it is a destination node for the RREQ. E will get the session key K_{AE}, the validation nonce N_V, and the validation key K_V. Next, E will assemble an RREP message for the A's RREQ.

2) *Anonymous Route Reply:*

Destination Node: When D sends RREQ to E, E will assemble an RREP packet and send it back to D. The RREP packet format is as follows:

E:- (RREP, N_{rt}, (K_V, Onion (D)) K_{DE})

Where, RREP defines the packet type identifier, N_{rt} is the pseudonym of the route generated by E. The shared key, K_{DE} encrypt the K_V and Onion (D), which are taken from original RREQ. This RREP is projected for D.

Intermediate Node: It is assumed that D already has an established neighbour relationship with node B, C and E. Following entries will be updated in the neighbourhood table of D:

Neigh. pseudonym	Session key
N_E	K_{DE}
N_B	K_{BD}
N_C	K_{CD}

When D receives the RREP from E, it will look for the shared keys in its neighbourhood table. It will use the shared key to decrypt (K_V, Onion (D)) K_{DE}). If D successfully decrypts, then D will know that the RREP is valid and sent by E, and N_B is the next hop for RREP. D will verify the linkage between RREP with the already stored RREQ. D tries to decrypt V_{AE} stored in the routing table and then compares if it matches. If it finds the matches of V_{AE}, it will update the routing table as the following:

Req. pseudonym	Dest. Pseudonym	Ver. Msg.	Next hop	Status
N_{sq}	N/A	V_{AE}	N_E	Active

Since the N_V, is not issued by D, it is not the source of the RREQ. D will assemble another RREP packet and forwards to the previous node B in the following format:

D:- (RREP; N_{rt}; (K_V; Onion (B)) $K_{BD)}$

Where, N_{rt} and K_V are taken from the received RREP. Onion (B) is obtained by decrypting Onion (D). The key K_{BD} is obtained from the neighbourhood table of D.

The RREP packet will travel from E to A according to the onion. After updating the routing table, D will also update its forwarding table. E assigns different pseudonyms for different paths. The forwarding table will be updated as follows:

Route pseudonym	pre_hop_pseudonym	nex_hop_pseudonym
$N_{rt.1}$	$N_{B.1}$	$N_{E.1}$
$N_{rt.2}$	$N_{C.1}$	$N_{E.2}$

Source Node: Source node validates the RREP similar as that of intermediate nodes. A is the original source of the RREQ if the decrypted core of the onion N_A is equal to the nonce issued by A. The updated routing table of A will be as follows:

Req pseudonym	Dest. pseudonym	Ver. Msg.	Next hop	Status
N_{sq}	N_E	V_{AE}	N_B	Active

3) Anonymous Data Transmission:

A can transmit data to E along the established route. The packet format of the data is as follows:

A:- E : (DATA, N_{rt}, (P_{data})K_{AE})

Where, DATA is the identifier of the packet type; N_{rt} is the route pseudonym, P_{data} is the payload and K_{AE} is session key used to encrypt payload.

All nodes look into the forwarding tables, after receiving data packets. If the N_{rt} matches with the entry of forwarding table then the node will forward the packet to the next anonymous node else the packet will be dropped.

In ad hoc networks the shape of routing paths may change considerably because of mobility of nodes while the connectivity is undamaged. Most of the previously proposed on-demand routing schemes do not initiate a new path discovery process until there is a link failure. The Path Aware algorithm monitors the route and tries to shorten it if shortcut path is available. Consider source node Src_q and a destination node $Dest_q$. When node a receives a packet it first checks for the available shortest path. The algorithm is as follows.

Algorithm : Path Aware Algorithm

Step 1: When node a receives or overhears a packet P, IF the node a is the final destination address, consume the packet. GOTO END;

Step 2: (Assume P belongs to <Src_q, $Dest_q$> flow). Compare <Src_q, $Dest_q$> to all the valid entries in the hop comparison array;

Step 3: IF there is no match with the entries, store <Src_q, $Dest_q$, HC_q, $Neigh_q$> in the hop comparison array;

Step 4: IF the packet is destined to a as the next-hop node,process the packet for forwarding further.

Step 5: (Assume that it matched with an entry <Src_q, $Dest_q$, HC_p, $Neigh_p$>) IF ($HC_q - HC_p > 2$), a short-cut is found, node a does the following:

Step 5.1: Send a message to $Neigh_p$ to update the routing table such that the next hop address for destination node $Dest_q$ is modified to the address of node a;

Step 5.2: Modify its routing table by making the next-hop address for destination $Dest_q$ as $Neigh_q$;

Step 5.3: Modify its hop comparison array, delete the entry corresponding to <Src_q, $Dest_q$>;

Step 6: Return the delay efficient path.

Step 7: Stop.

If it is the final destination packet will be consumed. The current hop count and stored hop count field is compared, if there is difference of two, then shortcut path exists.

III. SIMULATION

Proposed DEAASR protocol is implemented in NS-2 by extending AODV module. Performances of DEAASR are compared against AASR and AODV.

1) Performance Metrics

- Throughput – Throughput is the percentage number of packets successfully reaching the destination over communication channel. It is measured in terms of bits per second.
- Packet Loss – It is the difference between number of packets sent or transmitted and number of packets received. Packet loss is proportional to packet drop. Lower value of packet loss means better the performance.
- Average End to End Delay – It is the average time taken by a data packet to arrive in the destination. Lower end to end delay means better performance of the protocol.

2) Results

Here 2 scenarios of simulation results are considered.

1. Scenario 1: Varying mobility speed

When mobility speed increases, the throughput varies. As compared to AODV and AASR, DEAASR achieves highest throughput. DEAASR achieves less packet loss ratio under different number of mobile scenarios as compared to AODV and AASR. Due to additional security processing time in RREQ flooding, AASR has longer delay than AODV. Since DEAASR uses path aware routing so its delay is lower than AASR.

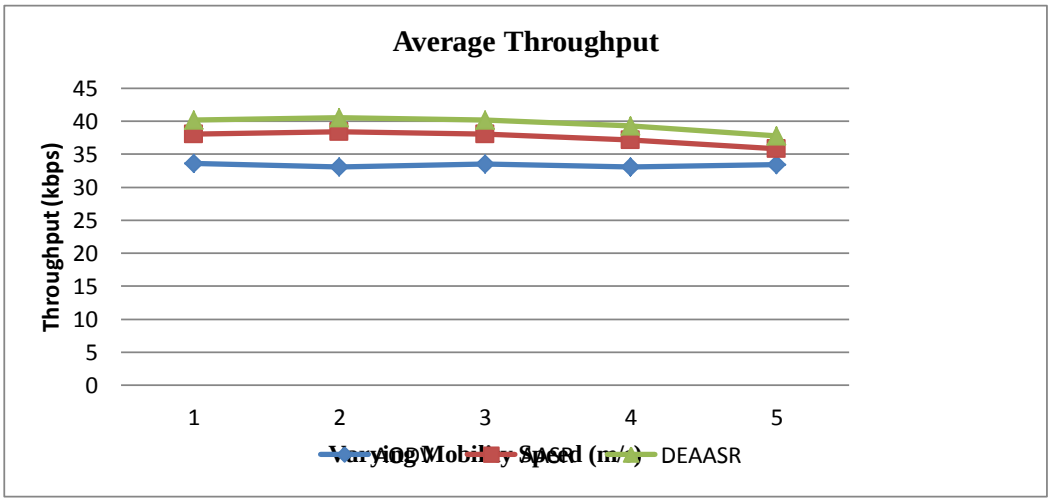

Fig 2: Throughput comparison under different mobility scenarios

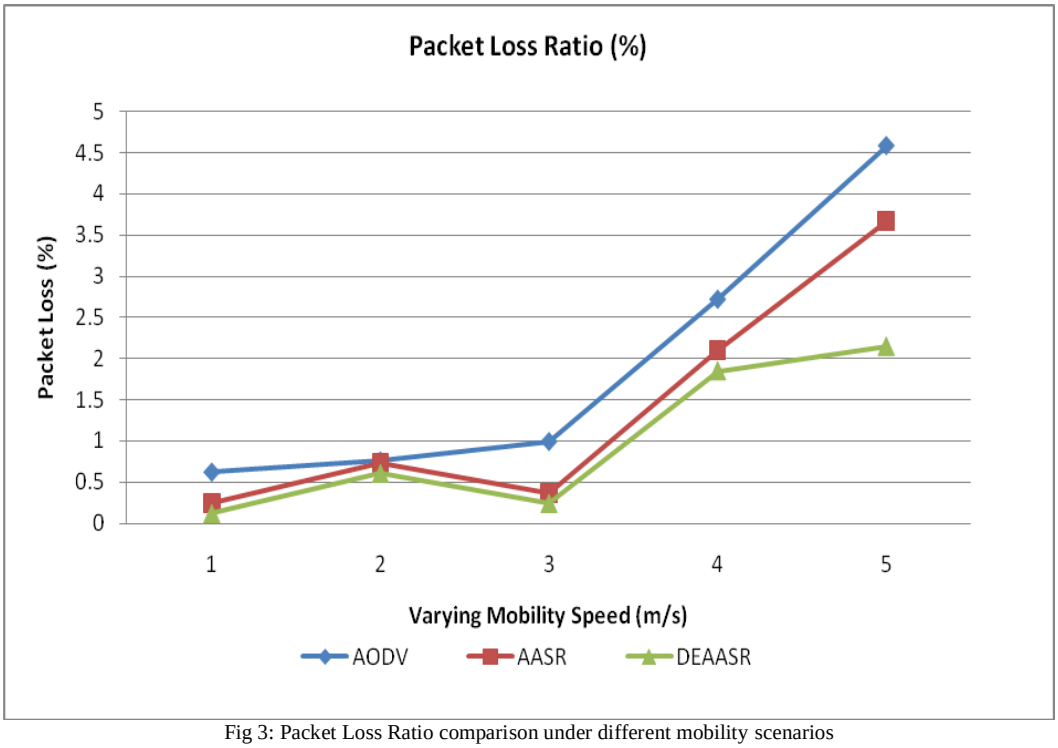

Fig 3: Packet Loss Ratio comparison under different mobility scenarios

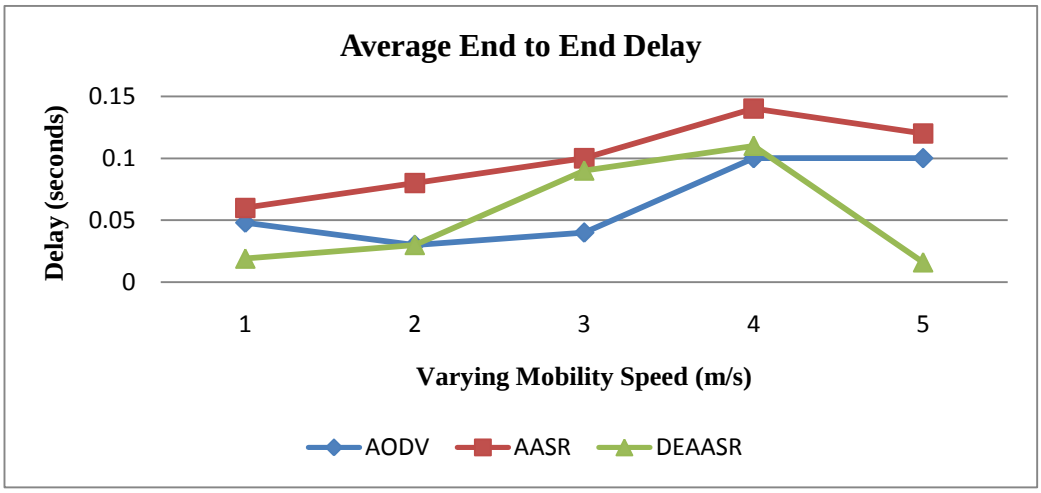

Fig 4: End to End Delay comparison under different mobility scenarios

2. Scenario 2: Varying number of malicious nodes

When number of malicious nodes increases, the average throughput of three protocols decreases. Since DEAASR has the ability to detect the packet dropping attack, it is better than AASR and AODV. DEAASR achieves less packet loss ratio as compared to AASR and AODV. AASR spends time in the security processing in the route discovery; their delays are higher than AODV. Since DEAASR uses path aware routing technique so its delay is lower than AASR.

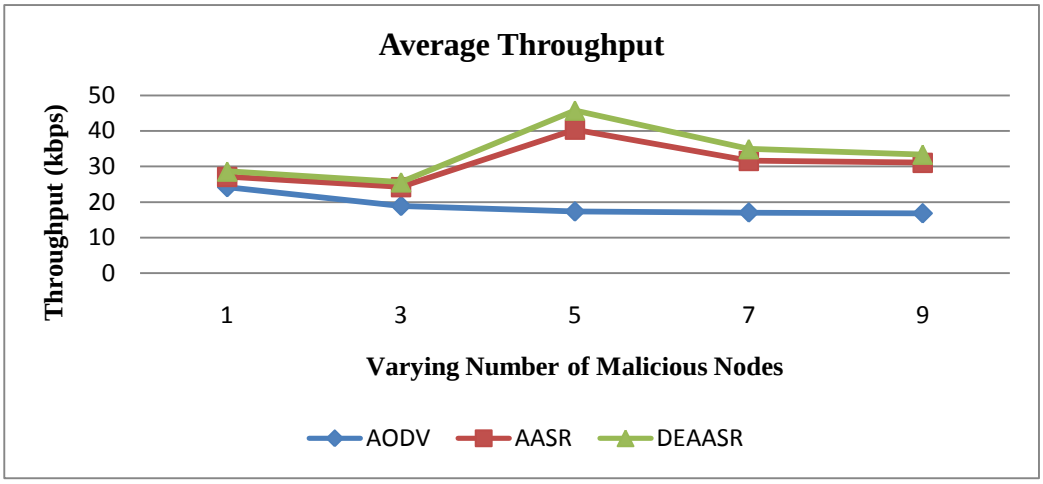

Fig 5: Throughput comparison under different number of malicious nodes

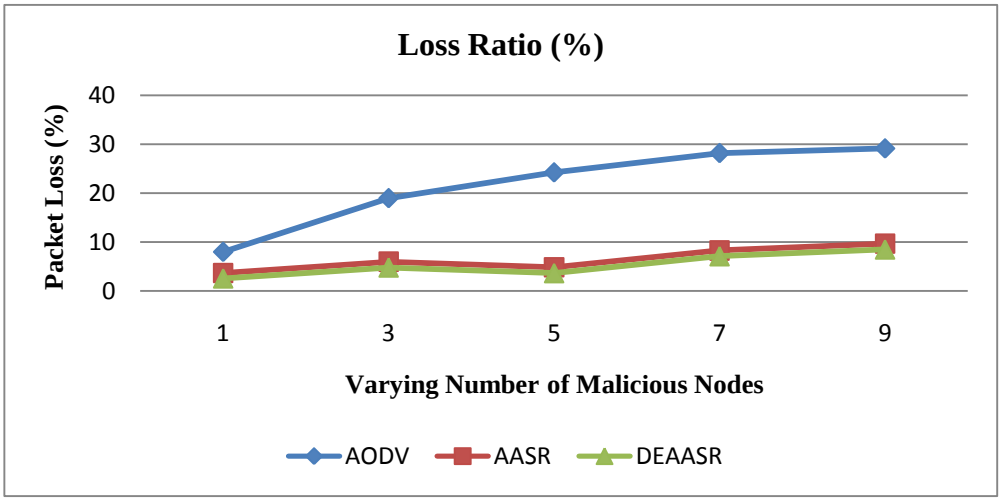

Fig 6: Packet Loss Ratio comparison under different number of malicious nodes

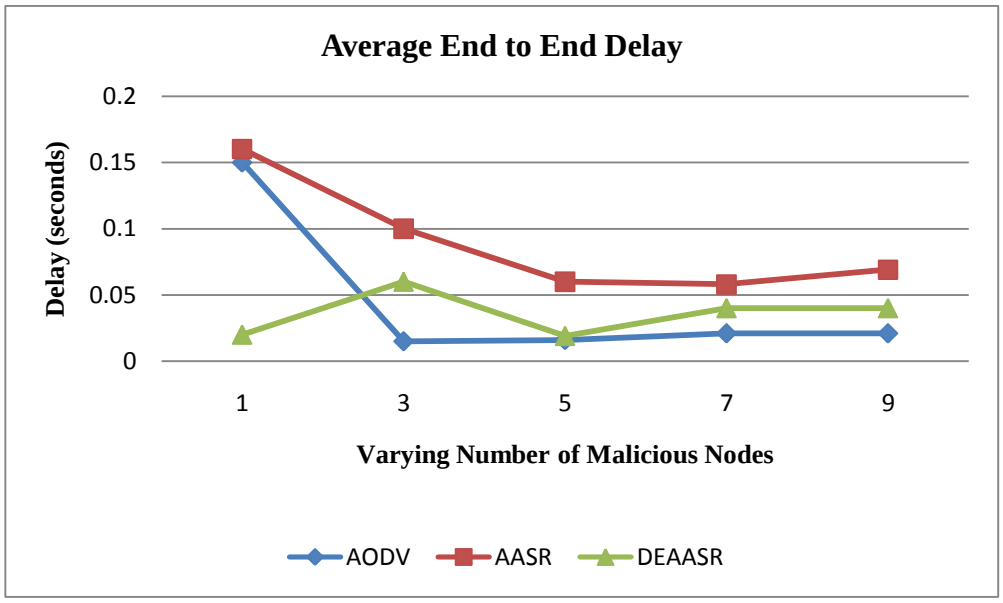

Fig 7: End to End Delay comparison under different number of malicious nodes

IV. CONCLUSION

The DEAASR model is designed to provide anonymity. The group signature scheme prevents the active attacker without introducing the node identity. The onion routing scheme prevents the intermediate nodes from deducing the actual destination. DEAASR is compared with AODV and AASR, DEAASR provides higher throughput, reduced packet delay and lower packet loss ratio.

REFERENCES

[1] Wei Liu and Ming Yu, *"AASR: Authenticated Anonymous Secure Routing for MANETs in Adversarial Environments"* IEEE Transactions on VehicularTechnology,Volume:63, No:9,November 2014.

[2] S. William and W. Stallings, *Cryptography and Network Security, 4th Edition*. Pearson Education India, 2006.

[3] M. G. Reed, P. F. Syverson, and D. M. Goldschlag, *"Anonymous Connections and Onion Routing," IEEE Journal on Selcted Area in Comm.*, vol. 16, no. 4, pp. 482–494, May 1998.

[4] D. Boneh, X. Boyen, and H. Shacham, *"Short group signatures,"* in *Proc. Int. Cryptology Conf. (CRYPTO'04)*, Aug. 2004.

[5] A. Boukerche, K. El-Khatib, L. Xu, and L. Korba, *"SDAR: a Secure Distributed Anonymous Routing Protocol for Wireless and Mobile Ad hoc Networks,"* in Proc. IEEE Int'l Conf. Local Computer Networks (LCN'04), Nov. 2004, pp. 618–624.

[6] J. Kong, X. Hong, and M. Gerla, *"ANODR: An identity-free and on-demand routing scheme against anonymity threats in mobile ad hoc networks,"* IEEE Trans. on Mobile Computing, vol. 6, no. 8, pp. 888–902, Aug. 2007.

[7] R. Song, L. Korba, and G. Yee, *"AnonDSR: efficient anonymous dynamic source routing for mobile ad hoc networks,"* in Proc. ACM Workshop Security of Ad Hoc and Sensor Networks (SASN'05), Nov. 2005.

[8] Y. Zhang, W. Liu, W. Lou, and Y. G. Fang, *"MASK: Anonymous On-Demand Routing in Mobile Ad hoc Networks,"* IEEE Trans. on Wireless Comms., vol. 5, no. 9, pp. 2376–2386, Sept. 2006.

[9] L. Yang, M. Jakobsson, and S. Wetzel, *"Discount anonymous on demand routing for mobile ad hoc networks"*, in *Proc. Int. Conf. SECURECOMM*, pp. 1–10, Aug. 2006.

[10] J. Paik, B. Kim, and D. Lee, *"A3RP: Anonymous and Authenticated Ad hoc Routing protocol,"* in Proc. International Conf. on Information Security and Assurance (ISA'08), Apr. 2008.

[11] C. Perkins, E. Belding-Royer, S. Das, *et al.*, *"RFC 3561 - Ad hoc On-Demand Distance Vector (AODV) Routing," Internet RFCs*, 2003.

