A Two Tiered Data Origin Authentication Scheme for Adhoc Network

Tibin Thomas[1] Karthik M[2] Leenu Rebecca Matheew[3] Jyothish K John[4]
[1, 2, 3] M. Tech [4]Assistant Professor
[1,2,3,4] Department of Computer Science and Engineering
[1, 2, 3, 4]Federal Institute of Science and Technology (FISAT) Angamaly, India

Abstract—Ad-hoc networks are becoming an effective tool for many mission critical applications such as troop coordination in a combat field, situational awareness, etc. Therefore, authenticating the source and ensuring the integrity of the message traffic become a fundamental requirement for the operation and management of the network. Since we are using small system in the adhoc network application, memory usage is also an important issue. Here we are presenting a system which reduces the memory attacks and control the memory usage of the devices in the adhoc network. By clustering the adhoc network we also achieve high scalability. One way hash function and MACs are used for the authentication purpose. The simulation demonstrates the advantage of this system to existing system in terms of throughput, memory, delay, etc.

Key words: Digital Signature, Message Authentication Code

I. INTRODUCTION

An adhoc network is a collection of autonomous nodes with dynamically changing infrastructure. By direct or by multi-hop communication nodes in the adhoc network can efficiently communicate. The nodes in the adhoc network have limitations in their on board energy and in their communication and computation power. So we required network management solutions which are suitable for these limitations. Since the adhoc network is covering very large areas, the network solutions must be scalable.

A. Security required in adhoc network

Assuring a certain level of security is a strong requirement for a large deployment of the communication model [1]. For example in a combat mission, each troop may require to report the status to other troops. Such transfer among the nodes has to be delivered in a secure and trusted manner. So each adhoc network should provide the following the following features. (a) Confidentiality, prevent the third parties from reading the data, (b) Message source authentication, assuring that message send from the legitimate user, and (c) Message integrity, this prevent third parties to alter the transmitted data.

Some application use data origin authentication without non-repudiation to avoid the complex computation, but others use authentication with non-Repudiation.

B. Tiered System

For improving the scalability of authentication, the adhoc network may be divided into many tiers. This help to add any number of nodes to the network without increasing the delay of authentication. For example for two tiered systems, nodes are clustered into many groups. Nodes in each cluster use intra cluster authentication to send data between nodes in the same cluster. For sending data from a node in one cluster to another, two tiered system use inter cluster authentication.

C. Contribution

This paper proposes a two tiered authentication scheme for traffic flow in adhoc network. The scheme uses network clustering in order to cut overhead and ensure scalability. Traffic within the same cluster employs one- way hash chains to authenticate the message source. It also prevents the memory DoS attack of intra cluster authentication. For that in intra cluster authentication part, sender sends the MAC and key index before the message and the corresponding key.

D. Organization of the Paper

The remaining of the paper is organized as follows. Section 2 describes the related works. The survey describes much type of authentication schemes, which can be used for making tiered authentication systems. In section 3 we describe an existing two tiered scheme, TAM and its main problem. The proposed system is mentioned in section 4 and implementation details of the existing and proposed schemes are described in section 5. In section 6 we compare the existing and proposed scheme based on parameter like delay, throughput etc.

II. SURVEY

We classify the source authentication into following two categories: (1) authentication with non-repudiation and (2) without non-repudiation. The category two can again classified as follows (1) secret information asymmetry, (2) time asymmetry, and (3) hybrid asymmetry. The asymmetry property denotes that a receiver can verify the message origin using the MAC in a packet without knowing how to generate the MAC. A full survey of the authentication scheme is discussed in the paper [11].

Latif-Aslan-Ramly Multi-cast Authentication Protocol is an authentication protocol described in [3] uses both public key signature and UMAC function. It is an authentication protocol with non-repudiation. It uses eraser code function to resist the packet loss and a counter value to resist reply attack. The use symmetric encryption system like AES provides confidentiality.

In the data origin authentication scheme without asymmetry have three major approaches to introduce asymmetry in authentication data.

1) Secret Information Asymmetry: Here each sender has a set of secret keys. Each receiver has a share of these keys. In this strategy for creating an authentication information requires the knowledge of all keys. So the receivers cannot forge authentication information.

2) Time asymmetry: In this scheme the time asymmetry is achieved by changing the shared key periodically
3) Hybrid asymmetry: This is the combination of both time and secret information asymmetry.

The Canetti et al. protocol is a secret information asymmetry protocol that [1] assure the authentication by appending a MACs to the message. Sender calculates the MACs using k different keys. Each receiver holds a share of secret keys among k keys and verifies the authenticity of the received massage using that shared keys.

PAPER	APPROACH	LATENCY	TOLERATE PACKET LOSS	SYNCHRONIZATION REQUIRED	ADVANTAGES
Efficient authentication and signing of multicast streams over lossy channels	EMSS: attaching hash of previous packet, and signing.	At the receiver side	The authentication probability of a packet is at least 90 percent	Yes	With nonrepudiation.
Multicast security: a taxonomy and efficient constructions	MAC based secret information asymmetric scheme.	At the source side	Yes	No	Dynamic addition of sources. Saved the time needed to generate the signature.
The BiBa one-time signature and broadcast authentication protocol	Utilizes birthday paradox scheme. Generate one way SEAL chain for broadcast authentication protocol	At the source side	Yes	Yes	Smaller signature, so verification is fast. With non-repudiation
Multi-receiver/Multisender Network Security: Efficient Authenticated Multicast/Feedback	Polynomial based secret information asymmetry	At the source side	Yes	No	Each packet is small in size.
A Graph-based new amortization scheme for multicast streams authentication	Amortize a single signature over a group of packet. Multiple Connected Chain Model scheme is used.	Not at source or receiver side	Yes	No	Higher authentication probability.
New Real Time Multicast Authentication Protocol	Latif-Aslan-Ramly Multicast Authentication Protocol using public key signature & UMAC	At source and receiver side	No	No	Resist pollution attack Resist replay attack .
MABS: Multicast Authentication Based on Batch Signature	Use batch signature scheme with packet filtering.	Not at source or receiver side	Yes	Yes	No correlation among packet. Verification possible even if some packet attacked.
Digital Signatures for Flows and Multicasts	Tree chaining technique. Feige-Fiat-Shamir digital signature scheme was extended.	At source and receiver side	Yes	No	Improve signing and verification rates. Packets in a flow are individually verifiable
The Elliptic Curve Digital Signature Algorithm (ECDSA)	Mathematical form of DSA. Uses asymmetric key pair	At source side	No	No	Strength per key bit is high. Secure and faster dissemination of information.
A hybrid scheme for multicast authentication over lossy networks	Combines Gennaro & Rahatgi and Gollee & Moderdugo scheme	Not at source or receiver side	Yes	Yes	Authentication can be performed at real time Resist the packet loss & joinable at real time
TAM: A Tiered Authentication of Multicast Protocol for Ad-Hoc Networks	Network is clustered for authentication, Both time and secret information asymmetry is used.	Not at source or receiver side	Yes	Yes	Less Communication overhead Scalable

Table. 1: Comparison of Different Authentication schemes

The TESLA (Timed Efficient Stream Loss-tolerant Authentication) protocol proposed by Perrig et al. [2] uses one way key chain to create the MAC for a message. It is a time asymmetry protocol. The senders first generate one way key chain to use as the MAC keys. From that a secret MAC key used to generate the MAC for a particular message in a time interval. Then the message with the corresponding MAC is send to the receiver. The key that is used to authenticate the message is kept secret for a time interval, and discloses the key to the receiver after the interval. This prevents the attacker to receive the key before the message. Upon receiving the key the receiver can verify the authenticity of the previously received packet.

Althouse et al. proposed a hybrid two tiered scheme TAM [4], which exploits network clustering to reduce overhead and increase scalability. In this method, the entire adhoc network is divided as clusters, where the inter cluster authentication is done using time asymmetric approach while the intra cluster authentication is done using TESLA protocol (discussed in the previous section). Cluster head is created for each cluster for communicating in inter cluster multicasting. Less communication overhead and scalability are the main advantages. The main disadvantage of this scheme is memory DoS attack. Delayed authentication and time synchronization are the other challenges of this method.

III. EXISTING SYSTEM

Here we are considering an existing two tiered authentication scheme, TAM. TAM first partitions the entire network and then authenticates the traffic using time asymmetry in intra cluster authentication and secret information asymmetry in inter cluster authentication.

A. Intra Cluster Authentication

Since grouping the nodes into different clusters create a tight bound for both end to end delays for the delivering of the packet, we can use time asymmetry protocol based authentication. In TAM the intra cluster authentication done using TESLA protocol (described in the survey section).

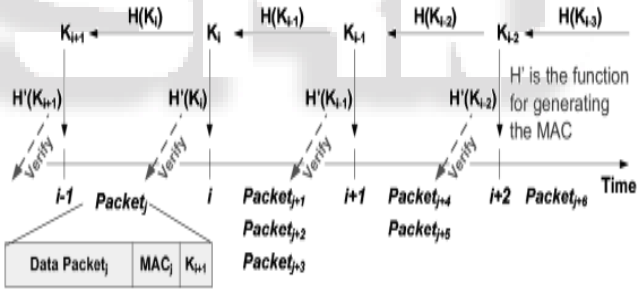

Fig. 1: A source used a key Ki during period j and reveals it in period j+1. Thus, a packet in period j will have a MAC based on Ki and will also include Ki+1 for authenticating the packet received in period j-1.

Intra cluster authentication is done as explained below. A source node generates a one way hash chain using SHA, MD5 etc... Then the source node share the last generated key Kl in the one way hash chain to the receiver, only after revealing the key used in generating the MAC, the receiver can authenticate the message. In order to verify the received key, receiver will recursively apply the hash function until reaching the key Kl. If the received key is outdated, then The receiver will ignore the MAC and the message.

B. Inter Cluster Authentication

Time asymmetric authentication requires clock synchronization. So it is not suited for large network. So TAM uses secret information asymmetry based authentication. Secret information asymmetry protocols can be used for both unicast and multicast.

C. Problem Definition

The existing two tiered authentication system use TESLA for intra cluster authentication. One of the main problems TESLA is memory DoS attack. The attacker may send data plus mac packet to the sender with invalid key. This cause memory waste at the receiver side and it leads to out of memory exception. Below we discuss a two tiered authentication system which uses TESLA++ as the authentication technique in intra cluster authentication.

IV. PROPOSED SYSTEM

A. Architectural Model

Adhoc network is an autonomous system that can be dynamically created without any predefined infrastructure. The system model considered in this paper groups nodes into clusters. The keys can be input either with the data or can be previously distributed to each cluster nodes.

Improving the scalability is the reason for clustering the network. Each cluster is controlled by the cluster head. The nodes in the cluster is reachable to cluster head, either directly or multi-hop path.

An attacker is considered in the system which tries to capture or compromise a node. If a node becomes a compromise node, then it can be used for attacking. When a node is captured, it can be used to creates attacks memory DoS attack, in others system.

B. The main advantages of this system are:-

1) It has a small MAC overhead.
2) Since the receiver can refer back to Kl, any missing of packet would not prevent successive packets from authentication.

C. Intra Cluster Authentication

Clustering of nodes enables a bound to delay of traffic and thus it enable to use time asymmetric authentication in intra cluster traffic. Intra cluster is based in this scheme is based on TESLA++ [13].

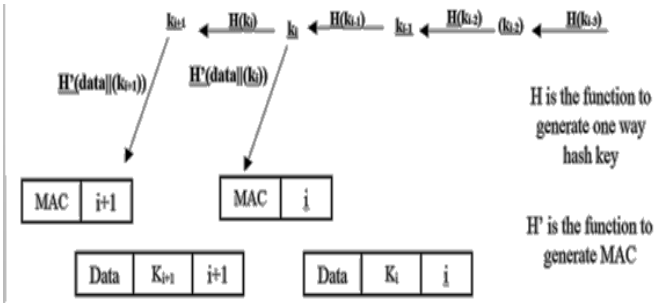

Fig. 2: The source first sends a MAC and key index i+1 and in the next packet it sends the data and corresponding key with key index.

A source node generates a chain of one-time-use keys using the hash function and shares only that last generated key; Kl sender sends the MAC and key index of the 1st data packet. On receiving the MAC packet, receiver checks the validity of the key index. The receiver receives the MAC packet only if the key index is greater than already accepted key index. After receiver receives the MAC packet, sender sends the corresponding data packet. On receiving the data packet with key, to verify the authentication key, the receiver recursively applies the cryptographic hash function until reaching Kl In reality; the receiver can stop when reaching a key that has been used before. The message will be ignored if the MAC is based on the expired key.

TESLA++ reduces the memory requirement for the receiver by reducing the size of the received MAC. On receiving the MAC and index at the receiver, TESLA++ reduces the size of the MAC by using any hash algorithm. For authentication of the data packet, receiver again creates the reduced MAC, and checks for a match in stored MAC list. TESLA++ can reduce the memory DoS attack.

D. Inter cluster authentication

For inter-cluster traffic, here we applies a strategy based on secret information asymmetry and engages the cluster heads in the authentication process. The inter cluster authentication can be used for the multicast traffic too. Here we create this secret information asymmetry protocol in that manner, but in simulation part we used single sender-receiver traffic. Secret information asymmetry works for the multicast in the following manner. Suppose the sources that belong to Cluster i will send the multicast packets to the heads of all clusters that have designated receivers.

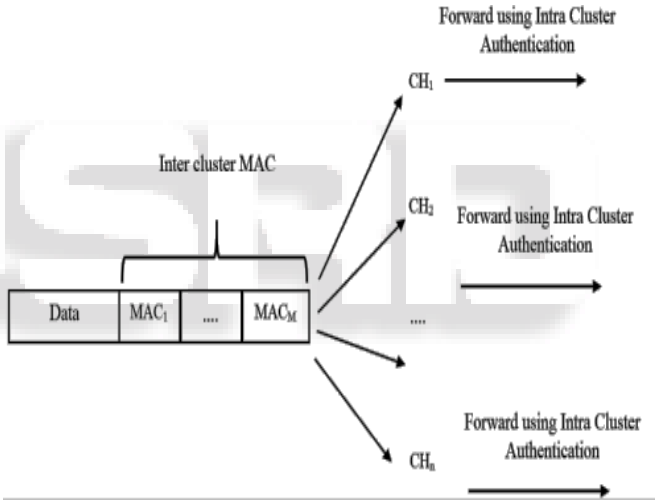

Fig. 3: Source sends M number of MACs to different cluster heads

The process is as follows. The source cluster head will generate a pool of N keys. Each of the clusters in the network will be assigned a share M of keys. This key share is first sends to all designated cluster head. The source will then create MAC with N keys and append N MACs to the data. On receiving the data with appended MACs, each receiver verifies M MACs using their share. If M MACs in the packet matches then the message can be accepted.

Agents for TESLA, TESLA++, and secret information asymmetry for inter cluster authentication and agents for generating attacks.

V. IMPLEMENTATION DETAILS

We implemented the system in ns-2.35. We added three new agents into ns-2.35, one for simulating TESLA, 2nd for simulating secret information asymmetry protocol, and last one for simulating TESLA++. Network with varying number of system is used.

A. Pictorial Representation of TESLA and TESLA++

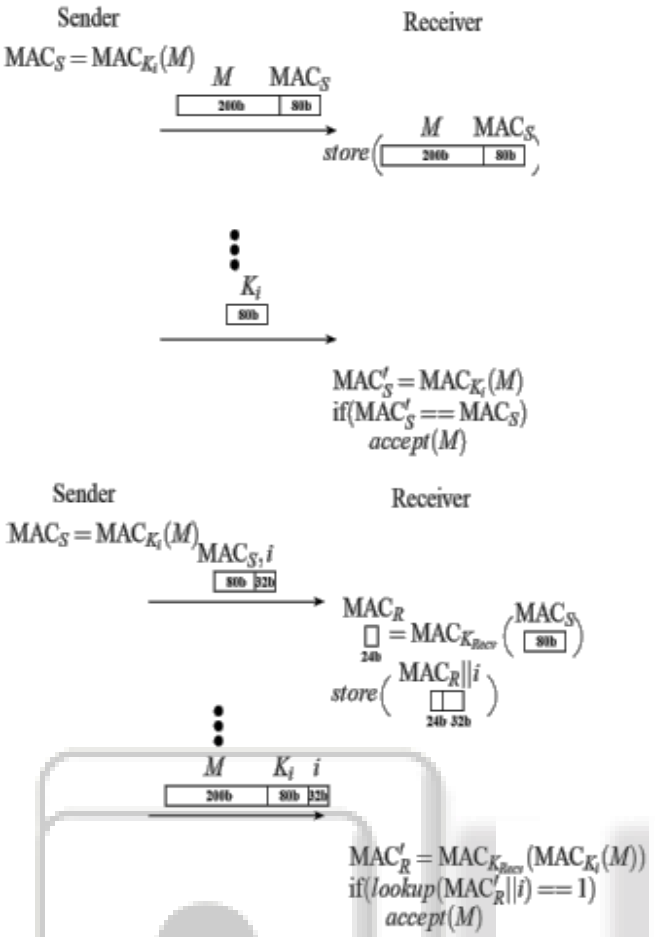

Fig. 4: Data flow of TESLA and TESLA++

B. Agent for TESLA

To implement TESLA we created the new agent below algorithm shows the working of TESLA agent.

Algorithm

1) Sender first sends the last key in the one way hash chain
2) Sender sends the data, with its MAC and the key used to generate the MAC of previous key.
3) On receiving the data, MAC and the key receiver checks the validity of the received key, by hashing the key. If the hash value of received key is equal to last accepted key, the received valid accept that key. Otherwise queue the data.
4) Use the accepted received key for authenticating the previous received data.
5) If the checking fails, that data can be rejected. Otherwise accept the data.
6) Frequently the data from the queue is checked for authentication.

One of the main disadvantages of TESLA is memory DoS attack. In the above class we can see a TESLA packet contain 128 byte data, 32 byte MAC, and 4 byte key. So if a packet loss we required to we need to queue the successive packet value. This queuing helps the attacker to create Memory DoS attack.

In this paper we are comparing the two tiered authentication schemes, with one uses TESLA for inter cluster authentication and another uses TESLA++ for intra cluster authentication. For that we need to create

C. Agent for TESLA++

To implement TESLA++, we created the new agent. TESLA++ works in the following manner

Algorithm

Step. 1 : Sender first sends the last key in the one way hash chain.

Step. 2 : Then then sender sends the first MAC packet (MACS =MACKi (M)), which is created using the current key from the one way hash chain and the key index i.

Step. 3 : On receiving the key index and MAC, the receiver first checks the validity of the key. If the index i, i.e. key Ki is expired, then reject it.

Step. 4 : Otherwise re-MAC the received MAC using the secret data known only to the receiver (MACR=MACKRecv(MACs)) and store the shortened MAC along with key index.

Step. 5 : After that the sender will broadcast any messages and the key used to calculate the messages MACs.

Step. 6 : To verify the message receiver first verifies the validity of the key Ki.

Step. 7 : The receiver then re-calculates the reduced MAC of the received message and checks it with the MAC and index stored in memory.

Step. 8 : If any stored MAC/key index pair matches, then receiver consider the message as authentic.

Step. 9 : If none of the stored pairs match the newly calculated value, the receiver considers the message unauthentic and discards the message.

Receiver store all MAC and key index pair in the memory. The receiver will free up the memory, when a stored MAC successfully authenticates a message. If memory space becomes insufficient, all shortened MACs with key indices that are older than the last authentic message received from that sender will be removed.

D. Agent for Secret Information asymmetry

Algorithm

1) The sender fixes number of share of keys for each receiver.
2) Then find (number of share X no of receiver) of keys
3) Send share to each receiver.
4) Then find (number of share X no of receiver) of MACs.
5) Then send the message with all appended MACs.
6) At the receiver side, the receiver verifies the message using their key share.
7) If minimum number of MACs satisfied then message can be accepted. Otherwise rejected.

E. Agents for Simulate the Attack

In TESLA, the main attack possible by an attacker is memory DoS attack, i.e. sending too many data packet, without sending the corresponding key in the successive packet. On receiving the data packet the receiver queue it in the memory and wait for the key. If too many packets received without key in the successive packet, it wastes the memory and at last it end up with memory out of condition. This scenario can be generated by sending only data packet

to the receiver, with invalid keys. In the actual case the key must be from the one way key chain, but here we select some random value as the key.

TESLA++ reduced the above mentioned problem by, removing the old stored MACs. TESLA++ also stores, the MAC with reduced size.

VI. PERFORMACE EVALUATION

The proposed scheme has been successfully simulated and tested using NS2 simulation platform.

A. Simulation Process

The network is created using different number of nodes at different times. We tested all type of possible sending and receiving scenarios like send from one node to another node in another cluster, send from one node to cluster head of another cluster or one cluster head to anode in another cluster etc. Here we use SHA512 for creating the MAC, which creates a 64 byte character data. At the receiver side for reducing the size of the MAC, TESLA++ uses polynomial hashing of the 64byte MAC which generates 4byte unsigned integer value. You can use any algorithm for creating the MAC (like SHA256, MD5 etc.) and also any algorithm that produce a smaller size MAC be used for creating the reduced MAC in case of TESLA++.In the following, paragraphs we us the parameters like throughput, end to end delay and packet delivery ratio with number of nodes in the network for comparing. Two tiered authentication network one using TESLA for intra cluster and another using TESLA++.

1) Throughput

Throughput means the max number of data send in a unit time. It can be found using the equation given below. Here we use different scenarios by varying the number of nodes in the network. Below diagrams shows the throughput of network containing 20 nodes and using TESLA and TESLA++. Table given in last of this section gives average throughput for networks with different number of nodes.

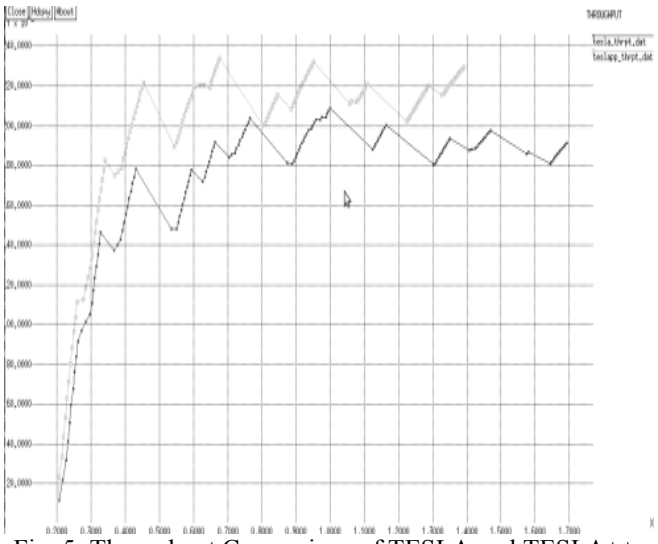

Fig. 5: Throughput Comparison of TESLA and TESLA++

We also did a comparison of Throughput on the basis of number of nodes in the network. From the graph below we can see that throughput is almost increases as the number of nodes increases, for both TESLA and TESLA++. We can also see that throughput for TESLA++ is greater than TESLA.

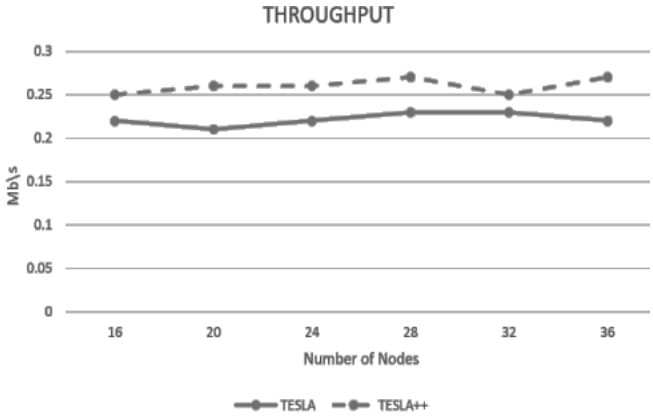

Fig. 6: Throughput Comparison of TESLA and TESLA++ for varying number of nodes.

2) End To End Delay

End to end delay is the time taken to receive the send data packet. The below graph shows the average end to end delay of the transmission as the number of nodes increases.

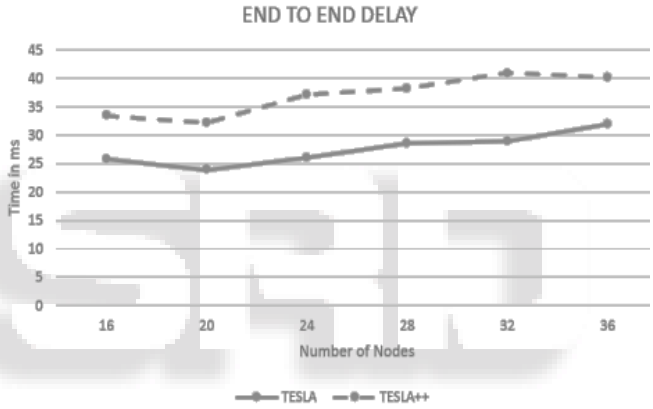

Fig. 7: End to End delay Comparison of TESLA and TESLA++

3) Packet Delivery Ratio

It gives the percentage of sent packet received at the destination side

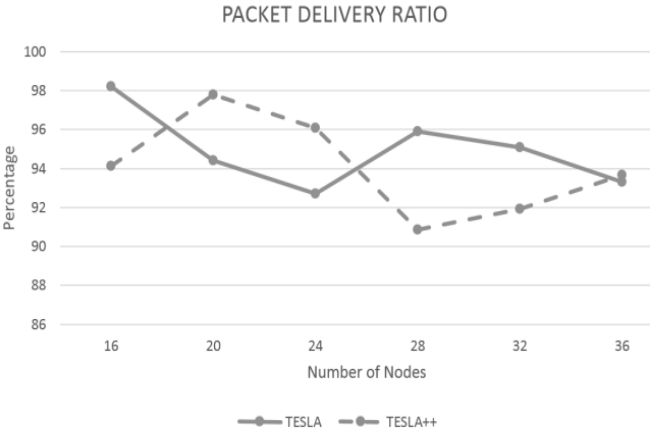

Fig. 8: Packet delivery ratio comparison for TESLA and TESLA++

The below graph shows the packet delivery ratio of the transmission as the number of nodes increases.

VII. CONCLUSION

Now days the adhoc networks are widely used in different applications, including security-sensitive application. Securing the traffic in adhoc network is very important, particularly authenticating the source and the message. We also required controlling the memory usage. In this paper we represented a two tiered authentication scheme, which have a good scalability due to the clustering mechanism and by also use counter measures to reduce the wastage of memory. Here we show only a scheme that can reduce the memory DoS attack. So as our first future plan includes the complete removal of memory dos attack, second is to create authentication for multicast traffic, and third is to dynamically creating clusters and finding cluster head.

REFERENCES

[1] P. Judge and M. Ammar, Security Issues and Solutions in Multicast Content Distribution: A Survey, IEEE Network, Jan./Feb. 2003, pp. 3036.
[2] Y. Challal, H. Bettahar, and A. Bouabdallah, A taxonomy of multicast data origin authentication, issues and solutions, IEEE Commun. Surveys & Tutorials, vol. 6, no. 3, pp. 3457, 2004
[3] Y. Zhou, X. Zhu, and Y. Fang MABS: Multicast Authentication Based on Batch Signature IEEE Trans. On Mobile Computing, vol. 9, no. 7, pp. 982-993 July 2010.
[4] M. Younis, O. Farrag and B. Althouse TAM: A Tiered Authentication of Multicast Protocol for Ad-Hoc Networks IEEE Trans. Network and a Service Management, vol. 9, no. 1, March 2012
[5] J. Menezes, P. C. van Oorschot, and S. A. Vanstone, Handbook of Applied Cryptography, CRC Press, 1996.
[6] R. Shirey, Internet Security Glossary, May 2000, RFC 2828.
[7] Kaufman, R. Perlman, and M. Speciner, Network Security: Private Communication in a Public World, Prentice Hall series in Computer Networking and Distributed Systems Ed., 2002.
[8] R. Canetti et al., Multicast Security: A Taxonomy and Efficient Constructions, INFOCOM, 1999.
[9] Perrig et al., Efficient and Secure Source Authentication for Multicast, 8th Annual Internet Society Symp. Network and Distributed System Security, 2001.
[10] A. Perrig, The BiBa one-time signature and broadcast authentication protocol, in Proc. 2001 ACM Conf. Computer Commun. Security.
[11] Tibin Thomas et al , Survey of Source Authentication Schemes for Multicast transfer in Adhoc Network, in IJSRD, vol. 1, no. 4,2013

