Journal of Theoretical and Applied Information Technology
10th October 2014. Vol. 68 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

CLUSTER HEAD BASED GROUP KEY MANAGEMENT FOR MALICIOUS WIRELESS NETWORKS USING TRUST METRICS

[1]V.BHUVANESWARI, [2] Dr. M.CHANDRASEKARAN

[1]Government Polytechnic College, Department of Computer Science and Engineering, Dharmapuri, INDIA

[2]Government College of Engineering, Department of Electronics and Communication Engineering, Bargur, INDIA

E-mail: [1]vbhuvaneswari66@yahoo.in

ABSTRACT

The process of transferring messages from a member to another member securely within a network is known as secure group communication. Key management is an important primitive to ensure this, as it provides a secure method for cryptographic keys creation, distribution and management. Group key establishment/management methods are key management's two sides. Group members use group key (GK) for encryption/decryption of messages in group communication. Communication needs quality and security for better performance and for acceptance of users and client companies. Diffie-Hellman (DH) was the first published public key algorithm that is used for secure key exchange mechanism. The purpose of algorithm is used to enable users to security exchange a key that can be used for subsequent encryption. Earlier schemes used only one group controller and were thus affected by single point failure (1-affects-n problem). To prevent this, a new technique where a control group generates the group key is introduced based on the nodes capability within two hops. In this scheme, direct trust and indirect trust is computed to identify Cluster Heads (CH) and the concept of auxiliary cluster head is introduced for effective key management.

Keywords: *Mobile ad hoc networks (MANETs), Dynamic Source Routing (DSR), Malicious Nodes, Clustering, Key Management.*

1. INTRODUCTION

Mobile Ad hoc NETwork (MANET) [1] consists of mobile nodes that communicate with each other without a predefined infrastructure or a central authority. Some characteristics of MANET include cooperation, dynamism of topology, lack of fixed infrastructure and resource constraints. Securitizing the routing process is a challenging task due to open exposure of wireless channels and nodes to attackers, lack of central agency/infrastructure, dynamic topology etc.[2].

Threats range from passive eavesdropping to active interference. Free roaming nodes become security issues having static configurations ensure that it is not enough in dynamically changing topologies. MANETs decentralized decision making is dependent on nodes cooperation. Cooperative algorithms can be broken when malicious nodes refuse to cooperate with them [3].

Data is transmitted to all n group members in group communication applications through minimum resources. Efficiency is achieved through bandwidth saving as data packets are transmitted only once between nodes. As against unicast based group communications where senders transmit n copies of the same packet [4]. Secure communication should ensure that access to transmitted data is limited to authorised group members alone. Encryption through selective key distribution ensures limiting group information access. An encryption algorithm is used to convert plain text (unhidden) into a cipher text (hidden) in order to secure against data thieves. Only knowledge of right keys will enable recovery of the original message from ciphered text [5] thereby securing multicast sessions. Chosen key encryptions protect messages which in group communications are known as the group key. With the known of the group key alone will recover the original message. The cryptographic keys used to

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

encrypt Group Key (GK), are called as Key Encryption Key (KEK). So the key management problem can be considered as the secure and efficient distribution of KEKs and GK to only valid members.

Authentication, access control, integrity verification and confidentiality are basic security mechanisms. A secure group communication session prevents non-group members from reading data exchanged within a secure group [6]. But this is possible only when group members establish and maintain a common key. Called group key or traffic encryption key (TEK) can both encrypt and decrypt message exchange in the group [6, 7]. Group orientations are used in Audio / Video conferencing, Online chatting, Military applications, Scientific discussion, Pay-per-view and other related applications.

A centralized controller generates and distributes the group key to all members in a centralized approach. This method's advantage is that it is efficient and facing the 1-affects-n problem is its biggest disadvantage. Decentralisation splits the group into many sub groups managed by various sub group controllers and they in turn are managed by the group controller. Scalability is decentralisation's advantage but it comes with a costly computation overhead. Group key is generated from group member's uniform contributions in a contributory approach and it has improved fault tolerance even though it has issues with power consumption and costly computation overheads.

The Traffic Encryption Key (TEK) [5, 9]: Group communication confidentiality ensures that non-group members cannot access data exchanged in a secure group communication session but this needs establishing and maintaining a common key – called group key or traffic encryption key - between members. This key encrypts/decrypts group message exchanges.

Key Encryption Key (KEK) [6]: The Key Encryption Key (KEK) is derived directly from the AK and it is 128 bits long. The KEK is not used for encrypting traffic data; so SS require the TEK from BS. TEK is generated as a random number generating in the BS using the TEK encryption algorithm where KEK is used as the encryption key. TEK is then used for encrypting the data traffic.

Previous schemes used only one group controller and they are affected by single point failure or 1-affects-n problem. A new technique is introduced to prevent single point failure wherein the concept of auxiliary CH is introduced to ensure redundancy. To avoid malicious nodes and to ensure secure key management, cluster heads exchange keys based on trust. The paper is organized as follows: Section 2 reviews some of the related works available in the literature, section 3 details the materials and methods used in this investigation, section 4 gives the results and section concludes the paper.

2. LITERATURE REVIEW

Cryptography plays an integral role in secure communication and is usually the strongest link in the chain of security. Multilanguage cryptography, an advancement of classical cryptography, may evolve as a choice of classical cryptography lovers seeking a better security. Srivastava, et al., [13] proposed an algorithm in Multilanguage approach, which generated different ciphertexts at different time for the same plaintext over a range of languages supported by Unicode. It has a better frequency distribution of characters in the cipher text than previous work on this approach. Bouassida, et al., [14] showed the specific challenges towards key management protocols for securing multicast communications in ad hoc networks, and provides taxonomy of these protocols in MANETs. A new approach, called BALADE, was also presented. It was based on a sequential multi-sources model, and taken into account both localization and mobility of nodes, while optimizing energy and bandwidth consumptions.

Diffie-hellman was the first published public key algorithm that was used for secure key exchange mechanism. The purposed of algorithm was used to enable users to security exchange a key that could be used for subsequent encryption. This cryptographic problem ensure A (resp. B) that no other participants aside from B (resp. A) Can learn any information about the agreed value and often also ensure A and B that their respective partner has actually computed this value. But this algorithm was no longer strong, since the key can be easily identified by discrete logarithmic approach. Hence, in order to strengthen this algorithm, Thanuja, et al., [15] was generated the private keys by own mathematical equations. The user alone was going choose keys from many available factors for that mathematical equation that was already defined. Hence, it was difficult for the intruder to identify the correct factor for the equation that we framed.

In praxis Diffie-Hellman key agreement was very often used as part of security protocols or

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

security standards to secure data over public and communication systems, thus the security of the Diffie-Hellman was critical because any weaknesses can lead such systems to become vulnerable against attacks. Kakish [16] was introduced a security improvement that makes the Diffie-Hellman key agreement and encryption scheme more secure against attacks, such as the known plaintext attacks, it suggests the use of randomized parameter in both schemes, this will allows to produce a new shared secret key each time a communication session is build and to generate different encryption messages for all kinds of messages even for same message, thus making the Diffie-Hellman more secure compared with the basic version of the Diffie-Hellman.

Wu, et al., [17] introduced a MANET setting adapted, simple group key management scheme in which a multicast tree is formed in MANETs for efficiency. To achieve fault tolerance, two multicast trees are constructed and maintained parallelly. When one tree links is broken, it is substituted by the other. One tree is named blue and the other red. Group members act as group coordinators in rotation to compute/distribute intermediate keying materials to members through active tree links. This work is undertaken in rounds with the coordinator being selected in a distributed way. The latter is also responsible to maintain multicast group connections. Group coordinators compute/distribute intermediate keying materials through the underlying tree links to all members.

A scheme to provide security to dynamic multicast VoIP systems efficiently was suggested by Srinivasan, et al., [18]. Security is provided by media packets encryption from one user to others through a shared key called session encryption key. Group key management is the most time consuming process in a dynamic multicast VoIP environment. Whenever group membership changes, the key has to be updated and sent to all group members. Hence system performance can be improved by lowering update messages required for an updated key which in turn makes the scheme more efficient. Advantages of logical-key tree structure and Chinese remainder theorem are combined by the proposed secure key management scheme to achieve effectiveness. The proposed scheme's efficiency is compared with existing schemes. Comparisons reveal that it outperforms current schemes regarding key update messages reduction.

An authenticated key transfer protocol based on secret sharing scheme that KGC can broadcast group key information to all group members was proposed by Harn, et al., [19] Here group key recovery is only through authorized group members. Information is theoretically secure due to the confidentiality of this transformation. Group key transportation authentication is provided.

Lim, et al., [20] suggested two group key management schemes for hierarchical self-organizing wireless sensor network architecture designed so that the forwarding node has more computational and communication burden with a similar load being kept very low with other sensor nodes. This also ensures multilevel security to sensor groups at various levels. Sensor network implements these encryption primitives efficiently without sacrificing strength.

A cluster-based group key management scheme for wireless sensor networks aimed at reducing communication overhead and sensor nodes storage cost was proposed by Zhang, et al., [21]. The procedure includes group key generation through cluster head collaboration with cluster nodes. Cluster heads are responsible to reconstruct and delivery group key. Performance evaluations reveal that the scheme has good security while simultaneously reducing communication overhead when compared to existing schemes like large scale WSN.

Gharout, et al., [22] presented a mobility support solution for group key management. The proposed scheme focuses on a secure, dynamic and scalable key management. Highly scalable to dynamic groups it treats node mobility with a null re-keying cost keeping perfect backward/forward secrecy. Simulation studies prove that it has better performance when compared with other protocols and also reduces overall overhead and re-keying messages number. It also has no security failures.

Drira, et al., [23] proposed a group key management framework based on a trust oriented clustering scheme. It was demonstrated that trust is a relevant clustering criterion for group key management in MANETs. Trust information enforce authentication and is disseminated by the mobility of nodes. Furthermore, it helps to evict malicious nodes from the multicast session even if they are authorized members of the group. Simulation results show that our solution is efficient and typically adapted to mobility of nodes.

3. METHODOLOGY

The Dynamic Source Routing protocol (DSR) is a reactive protocol [24]. This generates less overhead and provides more reliable routing than

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

proactive routing, but at the cost of finding the optimal route. Mobile hosts do not utilize periodic messages, with a consequently energetic advantage in battery consumption. DSR updates automatically only when it needs to react to changes in the routes currently in use. This protocol is simple and efficient. The protocol is composed of the two main mechanisms of "Route Discovery" and "Route Maintenance", which work together to allow nodes to discover and maintain routes to arbitrary destinations in the ad hoc network. Other advantages of the DSR protocol include easily guaranteed loop-free routing and very rapid recovery when routes in the network change. The DSR protocol is designed mainly for mobile ad hoc networks of up to about two hundred nodes, and is designed to work well with even very high rates of mobility.

Ad-hoc On-demand Distance Vector (AODV) is another on-demand routing protocol, which has characteristics very similar to that of DSR.AODV also discovers routes on an as needed basis via a similar route discovery process. However AODV differs from DSR in its route maintenance mechanism, it uses routing tables, one entry per destination .AODV relies on routing table entries to propagate an RREP back to the source and, subsequently, to route data packets to the destination. AODV uses sequence number maintained at each destination to determine freshness of routing information and to prevent routing loops. All routing packets carry these sequence numbers.

AODV maintains timer-based states in each node, regarding utilization of individual routing table entries. A routing table entry is expired if not used recently. RERR packets in AODV are intended to inform all sources using a link when a failure occurs. Route error propagation in AODV can be visualized conceptually as a tree whose root is the node at the point of failure and all sources using the failed link as the leaves [25].AODV uses expanded ring search to control the RREQ floods in the route discovery process. Expanded ring search is used initially to discover routes to unknown destination. In expanded ring search increasing larger neighborhoods are searched to find the destination. The Time-To-Live (TTL) field in the IP header of the RREQ packets controls the search. If a route to a previously known destination is needed, the prior hop-wise distance is used to optimize the search. This enables computing the TTL value dynamically.

Diffie-Hellman (DH) algorithm is an amazing and ubiquitous algorithm found in many secure connectivity protocols on the Internet. DH is a method for securely exchanging a shared secret between two parties, in real-time, over an untrusted network [26]. A shared secret is important between two parties who may not have ever communicated previously, so that they can encrypt their communications. As such, it is used by several protocols, including Secure Sockets Layer (SSL), Secure Shell (SSH), and Internet Protocol Security (IPSec). These protocols will be discussed in terms of the technical use of the DH algorithm and the status of the protocol standards established or still being defined. The mathematics behind this algorithm is conceptually simple enough that a high school student should be able to understand it. The fundamental math includes the algebra of exponents and modulus arithmetic.

3.1 Proposed Method

Trust calculation can be performed by each node for surrounding nodes and the calculated values are stored locally for later use being regularly updated based on new interactions. This algorithm holds the concept of risk value associated with each node in the network and can be derived from trust value needed to become a cluster head.

Random chosen of a node to become the cluster head checks the required trust present in the network. The algorithm helps in comparing the node's trust value by combining direct and indirect trusts in order to achieve whole trust. A required trust value ($T_{threshhold}$) is associated with each job to be node processed till all the cluster heads are selected finally. Trust (T) is then tested against trust sources along with direct trust value (D_t), indirect trust value (I_t), and total trust value (T_t). If the total trust value is higher than or equal to the required trust value then the node can be selected as the CH provided none of the two hop nodes have higher Trust value than the current node. The second highest trust value within the two hop node becomes an auxiliary node.

The cluster head is elected if a node (X) is randomly to become a cluster head, then the latter checks whether it had any earlier experience with its neighbourhood nodes and if so, the direct trust value (D_t) is as given in equation

$$D_t = \frac{\mp \sum_{i=1}^{n} T_{y_i}(x)}{n} \quad (1)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

where, $T_{y_i}(x)$ is the sum of the trust value with its two hop neighbors

If (D_t) is higher/ equal to (T_{max}), the associated risk is lower than risk threshold, then node (X) becomes cluster head provided there are no two hop neighbors who have higher T value compared to the current node (X) or node (X) checks if there are any recommendations about itself from other nodes. If that is the case then indirect trust value (It) is given in equation

$$I_t = \frac{\sum_{y=1}^{m} T_y(x)}{m} \quad (2)$$

where $T_y(x)$ trust value of node X based on recommendations from its two hop neighbors

If (I_t) is greater/equal to (T_{max}) then associated risk is lower than risk threshold and so node(X) becomes a cluster head that provided there are no neighbour nodes with higher T values. If neighbour nodes are already a cluster head and with lower T value then the CH can be replaced with node(X) and the other node becomes the auxiliary cluster head when required. If node (X) value T is lower than T_{max} then total trust value (T_t) is computed as

$$T_t = D_t * W_A + I_t * W_B \quad (3)$$

where W_A and W_B are weights assigned.

If (T_t) is greater than or equal to ($T_{threshod}$) then the process can be continued as mentioned above.

In case if all CH is not discovered then the $T_{threshold}$ is decreased.

Once CH is selected, the CH communicates with neighbour CH about route discovery and provides trust value certificates to its node members for each successful delivery it is involved with. The trust value certificates can be used by the nodes when it moves to adjacent clusters and this count is used to compute indirect trust. The indirect trust uses the communication data rate that can be denoted as R_c is the rate of successful communication with evaluated nodes. The value is between 0 and 1. The initial value is set as 1. The data delivery rate is denoted by R_d is the rate of successful packet delivery by the evaluated node. The indirect trust is the weighted sum of Trust value certificate and communication data rate.

The CH and the auxiliary node together are termed as the "control set" that creates the TEK agreement by using A-GDH2 from the cliques protocol. A-GDH.2protocol lets a group of users to agree on a contributively generated key [27]. It is based on Diffie-Hellman (DH) [28] key agreement method is responsible for key authentication. The A-GDH.2 protocol is shown in Figure 1.

Let $M=\{M_1,..,M_n\}$ be set of users wanting to share key S_n

$A-GDH.2$ executes n rounds

Initialize:

Let p be a prime and q a prime divisor of p-1

Let G be unique cyclic subgroup of $\mathbf{Z}_p^*$ of order q

Let α be a generator of G

Round i $(0<i<n)$

1. M_i selects $r_i \in R\mathbf{Z}_p^*$

2. $M_i \rightarrow M_{i+1}: \left\{ \alpha^{\frac{r_1 \ldots r_i}{r_j}} \middle| j \in [1,i] \right\}, \alpha^{r_1 .. r_i}$

Round n

1. M_n selects $r_n \in R\mathbf{Z}_p^*$

2. $M_n \rightarrow ALL\ M_i: \left\{ \alpha^{\frac{r_1 \ldots r_i}{r_j}.K_{in}} \middle| i \in [1,n] \right\}.$

Upon receipt of the above, every M_i computes:

$$\alpha^{\left(\frac{r_1 \ldots r_i}{r_j}.K_{in}\right)K_{in}^{-1}.r_i} = \alpha^{r_1 .. r_n} = S_n$$

Figure 1 *A-GDH.2 Protocol*

Users and the control set member use a centralized approach. Control set members form a key agreement zone and contributes to TEK computation in a cluster. Cluster members receive TEK from its controller and all clusters maintain TEK which encrypts/decrypts data. This ensures that 1-affects-n scalability is enhanced as rekeying after joining/leaving affects only affected cluster members. A node selects a best trust value route to distribute TEK in the cluster during data transmission as a key encryption key (KEK) is required. The control member and users under it form a group to generate KEK which transmits TEK to users.

4. RESULTS AND DISCUSSION

Experiments were conducted with 50 to 250 mobile nodes, spread over an area of 2 km by 2 km. The nodes communicate over UPD/IP network. The data rate is uniformly maintained at 11 Mbps for all

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

nodes. The transmission power of 0.005 watts and reception power threshold set at -95dBm is maintained. The malicious nodes are designed to randomly drop packets irrespective of the source or destination address. Experiments were conducted to simulate the 15% of the nodes being malicious. The following routing was used to evaluate the proposed method. The simulation results are compared with the trust model of Drira et al [23].

- Dynamic Source Routing protocol (DSR)
- Ad-hoc On-demand Distance Vector (AODV)
- Proposed DSR with GDH
- Proposed DSR with DH Key Management

Several performance metrics are used to compare the proposed DSR protocol with the existing one. The following metrics were considered for the comparison were

Packet Delivery Ratio: it is the ratio of the number of packets received and the number of packets sent.

Average End to End delay: it gives the mean time (in seconds) taken by the packets to reach their respective destinations.

Table 2 to 4 tabulates the simulation results for route discovery time, end to end delay and packet delivery ratio respectively. Figure 2-4 shows the same.

Table 1 Route discovery time

No. of Nodes	DSR	AODV	Proposed DSR with GDH	Proposed DSR with DH Key Management	Trust model proposed by Drira et al.,
50	0.692	0.741	0.714	0.694	0.736
100	0.806	0.855	0.864	0.84	0.891
150	0.962	1.04	1.08	1.05	1.114
200	1.12	1.202	1.35	1.326	1.392
250	1.46	1.564	1.58	1.547	1.629

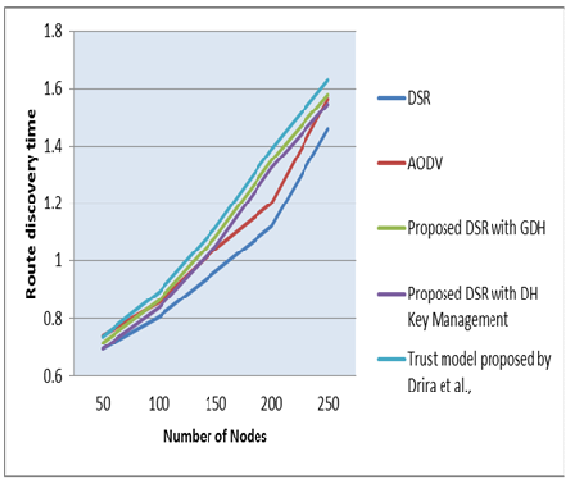

Figure 4: Average recall

It is observed from figure 2 that the route discovery time for the proposed DSR is slightly more than the DSR. As the increase is negligible when compared to DSR, the delay is overlooked. When compared to Drira et al trust model, the proposed DSR with DH key achieves 5.7% less route discovery time for a 100 node network and 4.7% less for 250 node network.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 2 End to End Delay

No. of Nodes	DSR	AODV	Proposed DSR with GDH	Proposed DSR with DH Key Management	Trust model proposed by Drira et al.,
50	0.007	0.007	0.00658	0.00632	0.00698
100	0.0094	0.01	0.00782	0.00747	0.0083
150	0.0104	0.011	0.00874	0.00836	0.00928
200	0.0138	0.014	0.00924	0.00891	0.00981
250	0.0165	0.017	0.0114	0.0103	0.0121

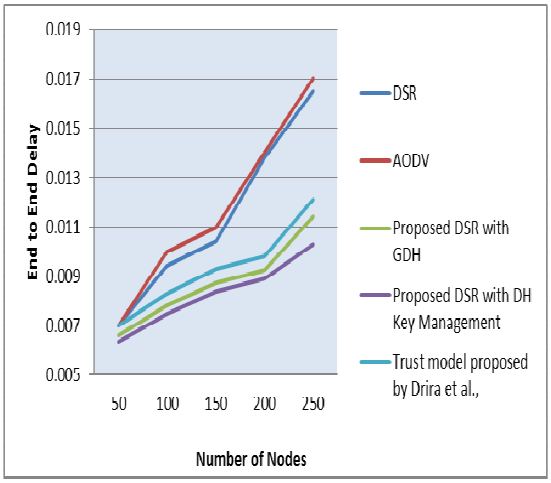

Figure 3: End to End Delay

The end to end delay in the proposed DSR is considerably less and it is observed that with the increase in number of nodes, the delay in DSR increases drastically. The proposed DSR with DH key achieves 9.7% decrease in end to end delay for a 50 node network and 37.6% for a 250 node network when compared to DSR. Though the Drira et al achieves better end to end delay performance when compared to DSR, the proposed DSR with DH key achieves 14.9% less end to end delay for a 250 node network than the trust model of Drira et al.

Table 4: Packet Delivery Ratio

No. of Nodes	DSR	AODV	Proposed DSR with GDH	Proposed DSR with DH Key Management	Trust model proposed by Drira et al.,
50	0.942	0.896	0.961	0.945	0.951
100	0.928	0.883	0.947	0.931	0.937
150	0.904	0.86	0.924	0.908	0.914
200	0.874	0.832	0.892	0.877	0.883
250	0.843	0.802	0.878	0.863	0.869

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

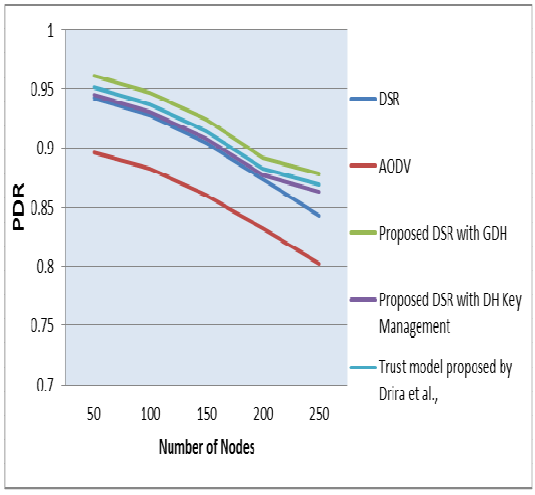

Figure 4: Packet Delivery Ratio

The PDR improves in the range of 2.02% to 4.15 % with the use of proposed DSR with GDH when compared to DSR. Similarly, the proposed method achieves 7.21% to 9.48% more PDR when compared with AODV. The proposed DSR with GDH achieves an average of 1.05% more PDR than Drira trust model. It is observed from the tables and figures that the proposed DSR performance better than DSR in the presence of malicious nodes and has better performance in larger network.

5. CONCLUSION

MANETs vulnerable to malicious node attacks are also liable to have packets dropped in such attacks. Key management is crucial for MANET security. This paper investigates network performance degradation due to such attacks when trust is used. Trust based clusters are formed based with routing considering intermediate nodes trust values. A control group generating the group key is proposed as a new technique in group key management. This includes construction of a tree with total users N being divided into many clusters. Secure key management is performed by malicious nodes being avoided due to cluster heads exchanging keys based on trust. Simulation results demonstrate the effectiveness of the proposed routing. End to end delay is considerably reduced with the proposed method and packet delivery ratio increases. It was also observed that the performance of proposed routing is considerably better in larger networks.

6. REFRENCES

[1] Raj, P. N., & Swadas, P. B. (2009). Dpraodv: A dyanamic learning system against blackhole attack in aodv based manet. arXiv preprint arXiv:0909.2371.

[2] Ramesh, P. S. T. (2013). Secure Routing using Reverse Method and Malicious Node Detection using Global Trust Scheme in MANET.

[3] Singh, S. P., & Maini, R. (2011). Comparison of data encryption algorithms. International Journal of Computer Science and Communication, 2(1), 125-127.

[4] Elminaam, D. S. A., Abdual-Kader, H. M., & Hadhoud, M. M. (2010). Evaluating The Performance of Symmetric Encryption Algorithms. IJ Network Security, 10(3), 216-222.

[5] Yavuz, A. A., AlagOz, F., & Anarim, E. (2010). A new multi-tier adaptive military MANET security protocol using hybrid cryptography and signcryption. Turkish Journal of Electrical Engineering & Computer Sciences, 18(1), 1-21.

[6] Kahya, N., Ghoualmi, N., & Lafourcade, P. (2012). Secure Key Management Protocol in WIMAX. International Journal, 4.

[7] Chiu, Y. P., Huang, C. Y., & Lei, C. L. (2012). SEMPRE: Secure Multicast Architecture Using Proxy Re-Encryption. International Journal Of Innovative Computing Information And Control, 8(7 A), 4719-4748.

[8] Srivastava, A. K., Sharma, S., & Sahu, S. (2012). Msmet: A Modified & Secure Multilanguage Encryption Technique. International Journal on Computer Science and Engineering, 4(3).

[9] Bouassida, M. S., Chrisment, I., & Festor, O. (2008). Group Key Management in MANETs. IJ Network Security, 6(1), 67-79

[10] Chen, Y. R., Tygar, J. D., & Tzeng, W. G. (2011, April). Secure group key management using uni-directional proxy re-encryption schemes. In INFOCOM, 2011 Proceedings IEEE (pp. 1952-1960). IEEE.

[11] Rahman, M., Sampalli, S., & Hussain, S. (2010, December). A robust pair-wise and group key management protocol for wireless sensor network. InGLOBECOM Workshops (GC Wkshps), 2010 IEEE (pp. 1528-1532). IEEE.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

[12] Gomathi, K., & Parvathavarthini, B. (2010, December). An efficient cluster based key management scheme for MANET with authentication. In Trendz in Information Sciences & Computing (TISC), 2010 (pp. 202-205). IEEE.

[13] John, S. P., & Samuel, P. (2010, October). A distributed hierarchical key management scheme for mobile ad hoc networks. In Information Networking and Automation (ICINA), 2010 International Conference on (Vol. 1, pp. V1-308). IEEE.

[14] Niu, Q. (2009, October). A Trust-Based Message Encryption Scheme for Mobile Ad Hoc Networks. In Computer Science and Engineering, 2009. WCSE'09. Second International Workshop on (Vol. 1, pp. 172-176). IEEE.

[15] Thanuja R, Dilip Kumar S A New Approach To Diffie-Hellman Key Exchange Algorithm.

[16] Kakish, M. J. Security Improvments To The Diffie-Hellman Schemes.

[17] Wu, B., Wu, J., & Dong, Y. (2009). An efficient group key management scheme for mobile ad hoc networks. International Journal of Security and Networks, 4(1), 125-134.

[18] Srinivasan, R., Vaidehi, V., Rajaraman, R., Kanagaraj, S., Kalimuthu, R. C., & Dharmaraj, R. (2010). Secure group key management scheme for multicast networks. International Journal of Network Security, 11(1), 33-38.

[19] Harn, L., & Lin, C. (2010). Authenticated group key transfer protocol based on secret sharing. Computers, IEEE Transactions on, 59(6), 842-846.

[20] Lim, S. Y., & Lim, M. H. (2011). Energy-Efficient and Scalable Group Key Management for Hierarchical Sensor Network. Journal of Ubiquitous Systems & Pervasive Networks, 2(1), 39-47.

[21] Zhang, Y., Shen, Y., & Lee, S. (2010, April). A cluster-based group key management scheme for wireless sensor networks. In Web Conference (APWEB), 2010 12th International Asia-Pacific (pp. 386-388). IEEE.

[22] Gharout, S., Bouabdallah, A., Challal, Y., & Achemlal, M. (2012). Adaptive Group Key Management Protocol for Wireless Communications. Journal of Universal Computer Science, 18(6), 874-898.

[23] Drira, K., Seba, H., & Kheddouci, H. (2010). ECGK: An efficient clustering scheme for group key management in MANETs. Computer Communications, 33(9), 1094-1107.

[24] Fotino, M., Gozzi, A., De Rango, F., Marano, S., Cano, J. C., Calafate, C., & Manzoni, P. (2007, July). Evaluating Energy-aware behaviour of proactive and reactive routing protocols for mobile ad hoc networks. In 10th International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS'07) (pp. 16-18).

[25] Tamilarasi, M., Palanivelu, T. G., Rajan, B., & Das, S. K. (2005). Node Optimization in MANETs for Maximum Throughput Using On-Demand Routing Protocols. In Proceedings of the Eleventh National Conference on Communications: NCC-2005, 28-30 January, 2005 (p. 66). Allied Publishers.

[26] Carts, D. A. (2001). A review of the Diffie-Hellman algorithm and its use in secure internet protocols. SANS institute, 1-7.

[27] Pereira, O., & Quisquater, J. J. (2002). Security analysis of the cliques protocols suites: first results. In Trusted Information (pp. 151-166). Springer US.

[28] Whitfield Diffie and Martin Hellman, "New directions in cryptography," IEEE Transactions on Information Theory, vol. IT-22, no. 6, pp. 644{654, Nov. 1976..