Hindawi Publishing Corporation
The Scientific World Journal
Volume 2015, Article ID 801632, 10 pages
http://dx.doi.org/10.1155/2015/801632

Research Article

Mobility Based Key Management Technique for Multicast Security in Mobile Ad Hoc Networks

B. Madhusudhanan,[1] S. Chitra,[2] and C. Rajan[3]

[1]*Department of Computer Science, Er.Perumal Manimekalai College of Engineering, Hosur 635117, India*
[2]*Er.Perumal Manimekalai College of Engineering, Hosur 635117, India*
[3]*Department of Information Technology, KSR College of Technology, Tiruchengode 637211, India*

Correspondence should be addressed to C. Rajan; crajanksr@gmail.com

Received 12 September 2014; Accepted 1 January 2015

Academic Editor: Hai Jiang

In MANET multicasting, forward and backward secrecy result in increased packet drop rate owing to mobility. Frequent rekeying causes large message overhead which increases energy consumption and end-to-end delay. Particularly, the prevailing group key management techniques cause frequent mobility and disconnections. So there is a need to design a multicast key management technique to overcome these problems. In this paper, we propose the mobility based key management technique for multicast security in MANET. Initially, the nodes are categorized according to their stability index which is estimated based on the link availability and mobility. A multicast tree is constructed such that for every weak node, there is a strong parent node. A session key-based encryption technique is utilized to transmit a multicast data. The rekeying process is performed periodically by the initiator node. The rekeying interval is fixed depending on the node category so that this technique greatly minimizes the rekeying overhead. By simulation results, we show that our proposed approach reduces the packet drop rate and improves the data confidentiality.

1. Introduction

A set of wireless communication nodes performing self-configuration in a dynamic mode for formation of network excluding fixed infrastructure or centralized supervision is termed as mobile ad hoc network (MANET) [1]. It defines the set of wireless heterogeneous mobile nodes that performs communication with each other over multihop paths devoid of fixed infrastructure [2]. The key aim of MANETs is to extend the mobility criteria in autonomous, mobile, and wireless domain. The nodes in MANET perform as both hosts as well as routers for sending the packet to each other [3]. During ad hoc routing, every node in the network is permitted to discover its multihop path via the network to any other node [1]. The application of the MANET includes military battlefields, emergency search, and rescue locations, and so forth which requires quick deployment and active reconfiguration. Here the members make use of mobile devices for sharing the information [1].

The process of broadcasting the packets to a group of zero or more hosts recognized by a single destination address is termed as multicasting [1]. This implies that message is transmitted from one sender to several receivers or from multiple senders to multiple receivers. The merit of multicast technique is that it offers service to multiple users exclusive of network and resources overloading in the server [4]. The multicast technique is utilized by the application such as routing, neighbor discovery, key distribution and topology control. This technique is also used in identical data transmission from a single sender to several receivers that minimizes the network traffic and energy consumption [5].

The multicasting approach can enhance the efficiency of the wireless links for transmitting the multiple copies of messages in order to utilize the inbuilt broadcast nature of wireless transmission. Thus, multicast takes a major responsibility in MANET. The major aim of multicast routing protocol is to reduce the control overhead and processing overhead, enhancing the potentiality of multicast routing protocol,

upholding the dynamic topology and avoids network loops and so on.

Security in Multicasting in MANET. The basic features of security in MANET are as follows: confidentiality guarantees that the network information cannot be revealed to the illegal unit. Integrity is essential to maintain the data to be transmitted among nodes without any change or degradation. Availability means that the services are demanded are available in timely manner without any potential issues in the system. The lack of authentication can cause the attacker masquerade any node and rules over the whole network. Nonrepudiation guarantees that the message forwarded cannot be refused by the message instigator [3].

Key Management. The methods of making, distributing, and updating the keys for a secure group communication application are termed as key management [6]. Encryption and reencryption are completed with the assistance of Traffic Encryption Keys (TEKs) and Key Encryption Keys (KEKs). In a secure multicast communication, each member possesses a key to encode and decrypt the multicast data. The method of updating and distributing the keys to the group members corresponds to rekeying operation. When each membership changes the rekey process is performed. However, throughout continual membership modulation, key management needs several exchanges per unit time for upholding forward and backward secrecies [7]. The secure multicasting is categorized into two types such as centralized and distributed scheme. The Group Controller (GC) performs group key management and only small loads are applied on the users of the group in case of centralized scheme. For distributed scheme, the key management is performed by each user to reinforce the load on the user [4].

2. Related Work

Chang and Kuo [8] have proposed a two-step secure authentication approach for multicast MANETs. A Markov chain trust model determines the Trust Value (TV) and the node with the highest TV is selected as CA server. The security analysis guarantees that this approach achieves a secure reliable authentication in multicast MANETs. Numerical results show that the analytical TV is very close to that of simulation under various situations. The speed of convergence of the analytical TV shows that the analyzed result is independent of initial values and trust classes. Huang and Medhi [9] have projected a secure group key management scheme for hierarchical mobile ad hoc networks to enhance each scalability and survivability of group key management for large-scale wireless ad hoc networks. A multilevel security model and a decentralized group key management infrastructure to come back through such a multi-level security model are projected. This approach reduces the key management overhead and improves resilience to any single point failure problem.

Bouassida and Bouali [10] have introduced an evaluation method for group key management protocols (GKMP). They have compared four main existing group key protocols, namely, scalable and efficient group rekeying protocol (GKMPAN) for ad hoc networks, Distributed Multicast Group Security Architecture (DMGSA), BALADE, and Hierarchical group key management protocol (Hi-GDH). In the above approaches, GKMPAN is an example for centralized approach. DMGSA approach belongs to distributed type key management scheme. BALADE protocol and Hi-GDH stand for decentralized approach. They have discussed the need for performance evaluation of GKMP's in the context of MANET's. Lin et al. [11] have proposed a new group key management protocol to reduce the communication and computation overhead of group key rekeying caused by membership changes. The protocol can handle synchronous and asynchronous rekeying operations, and a new k-node insertion algorithm is designed to further optimize the key tree in batch update operations. With strong encryption function and key derivation function, this protocol is provably secure. Simulation result shows that, compared to LKH, OFT, and ELK, SKD requires the least communication bandwidth and computation power, and it is efficient with binary key trees and asynchronous rekeying.

3. Proposed Work

The proposed technique uses Link Quality (LQ) and Reputation of nodes to identify them as strong or weak nodes. The multicast tree constructed with secure communication is based on the classified nodes and described in the subsections in detail.

3.1. Estimating Received Signal Strength. Here the proposed work makes use of the Friis free space propagation model to measure the received signal strength value. The received signal strength (RSS) is computed using the following formula [12]:

$$\mathrm{RSS} = \alpha * \theta * S_{\mathrm{tx}}, \tag{1}$$

where α is a constant that relies on the wavelength and the antennas. θ is the channel gain. S_{tx} is the signal power of the transmitter.

RSS can be expressed in terms of the dB and dBm (dB milliWatts) as follows:

$$\mathrm{RSS}\,[\mathrm{dBm}] = 10\log_{10}\alpha + \theta\,[\mathrm{dB}] + S_{\mathrm{tx}}\,[\mathrm{dBm}]\,. \tag{2}$$

3.1.1. Link Quality. Link Quality (LQ) is estimated by ratio of the number of bits in error to the number of bits received (bit error rate) [13]:

$$\mathrm{LQ} = \frac{b_{\mathrm{rx}}}{b_{\mathrm{error}}}. \tag{3}$$

This value gets updated for every packet received at a node over a certain period. It depends on parameters such as the interference effect of the wireless channel, additive white Gaussian noise, and signal transmission range.

3.1.2. Stability Index. Stability index (SI_{ij}) is computed for a link to a neighbor based on the received signal strength, mobility, and link quality (using Sections 3.1.1, 3.1.2, and 3.1.3) [13]. SI_{ij} of a link between node i and node j is defined as follows:

$$SI_{ij} = \frac{RSS}{LQ}. \quad (4)$$

3.1.3. Estimation of Reputation of Nodes. Consider nodes i and j.

The recent satisfaction index (P_{ij}) for node i about node j is computed as follows:

$$P_{ij} = f(i,j) - e(i,j), \quad (5)$$

where $f(i,j)$ is the percentage of packets originated from i that were forwarded by node j over the total number of packets offered to node j.

$e(i,j)$ is the percentage of packets that were expired over the total number of packets offered to node j.

Thus, P_{ij} can be considered as the direct reputation of node j:

$$Rep_{ij} = Rep_{ij\text{-pr}} * W_{hist} + P_{ij} * (1 - W_{hist}), \quad (6)$$

where $Rep_{ij\text{-prev}}$ is the reputation value that node i had for node j before incorporating the most recent satisfaction index.

W_{hist} is a constant that reflects the level of confidence that node i has in the past observed reputation for its neighbor j.

The reputation index REP_{ij} is normalized using the following equation:

$$REP_{ij} = \frac{REP_{ij}}{\max_t (REP_{ij})}. \quad (7)$$

$\max_t$ is the function that reports the maximum observation of REP_{ij} over time [14].

3.2. Classifying the Nodes. The nodes are categorized into two types, namely, strong and weak nodes. The steps involved in selecting the nodes are as follows.

(1) Each node deployed in the network periodically exchanges a HELLO packet with its neighbor nodes.

(2) By exchanging the hello packets, every node measures the RSS, link quality and mobility $M_j(i)$ of its neighbor nodes (explained in Sections 3.1.1 and 3.1.2).

(3) Based on the measurement of RSS, link quality, and $M_j(i)$, each node computes the stability index (SI) of its neighbor nodes (explained in Section 3.1.3) and the values are stored in the neighbor table (NT).

(4) The SI of each neighbor N_i is checked such that

Let SI_{th} be the predefined threshold value of Stability Index

If $SI_i < SI_{th}$

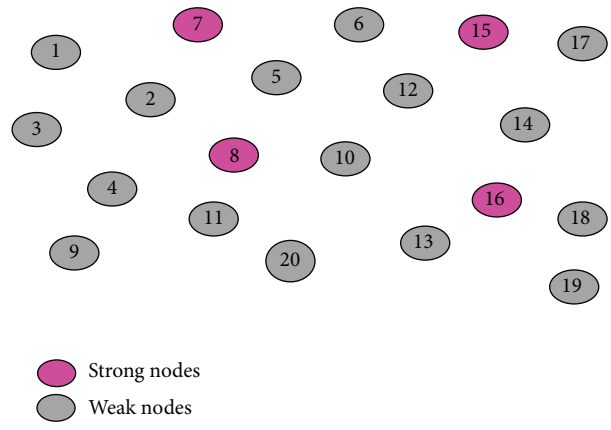

Figure 1: Selection of strong and weak nodes.

Then The nodes are marked as weak nodes (N_{wi}) and stored in NT

Else The nodes are marked as strong nodes (N_{si}) and stored in NT

End if

For example, consider the network in Figure 1. The nodes 7, 8, 15, and 16 are marked as strong nodes as their stability index is greater than the threshold value. Remaining nodes are marked as weak nodes as their stability index is less than the threshold value.

3.3. Multicast Tree Construction. The multicast tree construction phase involves two phases.

Phase 1. Each N_{wi} sends a child request message (CREQ) to each predetermined strong neighbor (N_{sj}) stored in NT:

$$N_{wi} \xrightarrow{\text{CREQ}} N_{sj}. \quad (8)$$

Upon receiving the CREQ message, N_{sj} sends a child reply message (CREP) to N_{wi}:

$$N_{wi} \xleftarrow{\text{CREP}} N_{sj}. \quad (9)$$

Every N_{wi} upon receiving CREP joins with N_{sj} as child nodes and respective N_{sj} becomes the parent node. Thus, for every weak node, there is at least a strong parent. N_{sj} then stores its child nodes information in a table.

For example, consider the network in Figure 2. The weak nodes 2 and 5 get attached with the strong node 7. Thus, nodes 2 and 5 become the child nodes for the strong parent node 7. In the similar manner, other strong nodes 8, 15, and 16 chooses their child nodes.

Phase 2. A multicast tree can be constructed and maintained using the periodic "JOIN_TREE" messages.

Each strong node N_{sj} periodically sends a "JOIN_TREE" message to the multicast source S:

$$N_{sj} \xrightarrow{\text{JOIN\_TREE}} S. \quad (10)$$

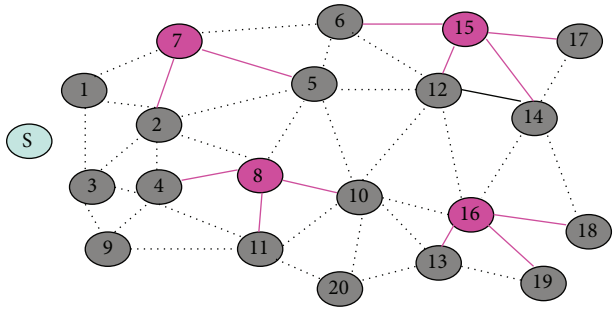

Figure 2: Phase 1: selection of child nodes.

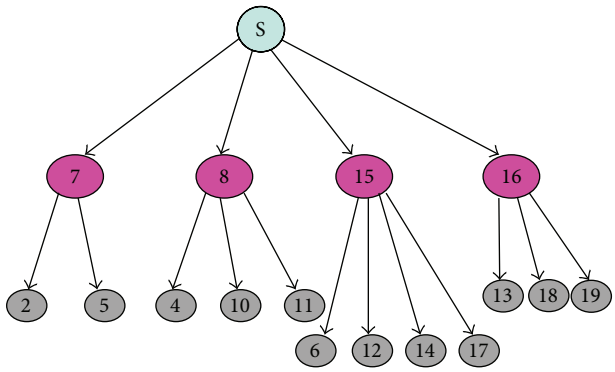

Figure 3: Multicast tree.

S constructs a multicast tree consisting of the paths that "JOIN_TREE" pass through. There is only one path from the S to each N_{sj} of the multicast group.

Figure 3 shows an example of a multicast tree constructed on a MANET. The parent nodes 7, 8, 15, and 16 sends JOIN_TREE message to S. S constructs a multicast tree consisting of the paths traversed by "JOIN_TREE" message.

3.3.1. Secure Multicast Communication. When any node N_i wants to transmit multicast data to destination D in a secured manner, it performs the following steps.

(1) Initially, N_i bounds the multicast data with hash message authentication code (Q) for ensuring the data integrity which is represented as Q(data).

(2) N_i and D cooperatively compute the session key K_{iD} and N_i utilizes K_{iD} to encrypt Q[data]. This encrypted data is represented as $K_{iD}[Q(\text{data})]$. Here, the session key is generated using Elliptic Curve Diffie-Hellman Key Management Agreement protocol (ECDH) [15].

(3) Every member node holds a group key GK_i. N_i again encrypts $K_{iD}[Q(\text{data})]$ with GK_i and it is represented as $GK_i\{K_{iD}[Q(\text{data})]\}$. GK_i is the multicast group key, where, $i = 1, 2, \ldots, n$.

(4) When any node along the path N_i-D receives the $GK_i\{K_i[Q(\text{data})]\}$, it decrypts the data using GK_i and encrypts it with GK_i again and forwards the encrypted data.

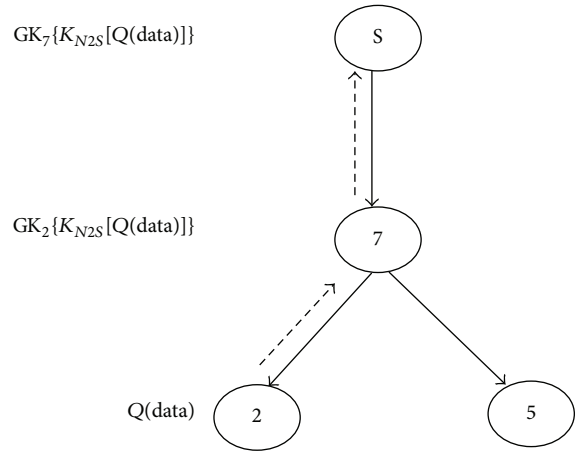

Figure 4: Secure data transmission.

(5) When D receives the encrypted data, it decrypts the data using its respective GK_i and session key K_{iD} and verifies the integrity of Q(data).

For example consider the network in Figure 4.

The node N_2 wants to transmit the data packet to S. The data to be transmitted will be in the form: Q(data).

Initially, N_2 and S cooperatively compute the session key K_{N2S} and N_2 encrypts Q(data) with K_{N2S} which is represented as $K_{N2S}[Q(\text{data})]$. N_2 again encrypts $K_{N2S}[Q(\text{data})]$ with group key GK_2 which is given as $GK_2\{K_{N2S}[Q(\text{data})]\}$. This encrypted data is forwarded to N_7.

N_7 decrypts the data using the GK_2 and encrypts again with GK_7 and forwards it to S which will be in following form

$$GK_7\{K_{N2S}[Q(\text{data})]\}. \tag{11}$$

When S is receiving the encrypted data, it decrypts the information victimization GK_7 and session key K_{N2S} and verifies the integrity of Q(data). If any changes happen throughout the transmissions, the receiving node detects the modifications in real time by validating the Q. The secured transmission of information between a node and therefore the supply is illustrated in Figure 4.

3.4. Detection of Attacker Nodes. When the data is not delivered at a reliable rate and optimum path quality, it is predicted that attack is detected. The attack detection technique depends on the capacity of I to detect the difference among the predicted PDR (PrP) and recognized PDR (ReP). The estimation of PrP and ReP is as follows.

PrP can be estimated from the Success Probability Product metric (SPP) at the concerned route.

SPP for a path of n links among S and D is given by

$$\text{SPP}_{S\to D} = \prod_{i=1}^{i} \text{SPP}_i, \tag{12}$$

where the metric for each link i on the path is $\text{SPP}_i = \text{Pr}_{\text{succ}}$.

ReP of a route is determined by testing the continuity of the sequence number in received data packets. That is by

dividing the number of received packets by the number of packets sent by the source over an interval of time.

ReP in terms of performance of packet delivery is given by the following equation:

$$\text{ReP} = \frac{P_r}{P_s}, \tag{13}$$

where P_r is the average number of packets received by all receivers and P_s is the number of packets sent by the source.

Even if the attacker nodes drop all data packets, initiator nodes have the capacity to determine the ReP with the inclusion of the backup data packet authenticated by the source:

If $|\text{PrP} - \text{ReP}| > \eta$

Then

The malicious behavior is detected by I since the particular route does not deliver the data at consistent level with optimal path quality.

End if

3.4.1. Isolation of Attacker Nodes. The steps involved in the isolation of attacker nodes are as follows.

Step 1. While detecting the malicious behavior, it temporarily recriminates the suspicious node by flooding a failure notice in the network that includes ID of recriminated and recriminator nodes and the period of recrimination.

Step 2. Until the recrimination is valid, metrics broadcasted by the recriminated node will not be taken into account and will be discarded during routing process.

Step 3. In case of transient network variations, the temporary recrimination scheme is taken into consideration.

Step 4. In temporary recrimination strategy, initially the time period of recrimination is computed in relative to the observed difference among PrP and ReP. This is performed with the intention that the recriminations caused by increase in metric values as well as malicious data dropping rate retains for longer duration than the recriminations caused by the transient network variations.

Step 5. In order to avoid the recrimination caused by attackers, a node is not permitted to announce a new recrimination prior to the expiry of the already announced recrimination.

Step 6. If the best metric is broadcasted by a recriminated node.

Then, the initiator node activates the recriminated node in addition to the best nonrecriminated node.

Step 6 reveals that the valid paths can still be utilized in spite of false recrimination of the strong nodes.

3.5. Rekeying Technique. Among the chosen N_{sj}, some nodes have to be designated as initiators, which initiates the re-process. In this section, suppose that initiators are selected by centralized node considering reputation index (RI) of nodes. The initiators are selected based on the RI of nodes (explained in Section 3.1.3).

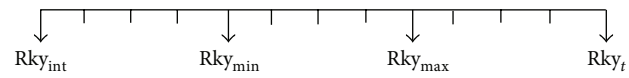

Figure 5: Rekeying time interval.

The direct reputation of node N_{sj} is given as

$$\text{Rep}_{ws} = \text{Rep}_{ws\text{-pr}} * z + P_{ws} * (1 - z), \tag{14}$$

where $\text{Rep}_{ws\text{-pr}}$ is the reputation value of N_{sj} contained in N_{wi} prior to the addition of recent satisfaction index. z is the constant that replicates the level of confidence possessed by N_{wi} for its N_{sj}. P_{ws} is the recent satisfaction index for N_{wi} about N_{sj}.

Thus, N_{sj} with high Rep_{ws} values are selected as initiators. The selected initiator starts the rekeying process periodically using the rekeying interval Rky_{int}. Rky_{int} is the fixed parameter and rekeying procedure is demonstrated as follows.

Let Rky_{int} be the initial time.

Let Rky_{max} represent the maximum thresholds for rekeying interval.

Let Rky_{min} indicate the minimum thresholds for rekeying interval.

Let Rky_t represent the stop time.

According to the rekeying interval, rekeying process is performed using the following cases. Figure 5 shows the rekeying time interval.

Case 1.

If $\text{Rky}_{\text{int}} > \text{Rky}_{\text{min}}$

then,

the rekeying is performed for requested weak nodes from NT by the initiator.

End if

Case 2.

If $\text{Rky}_{\text{int}} > \text{Rky}_{\text{max}}$

then,

the rekeying is performed for requested strong nodes from NT by the initiator.

End if

Case 3.

If $\text{Rky}_{\text{int}} = \text{Rky}_t$

Then

Rekeying is stopped and the timer is refreshed to start the new session.

End if

The rekeying is performed in the weak node within minimum rekeying interval since they possess minimum stability index which causes them to frequently join or leave the network. In the strong nodes, rekeying is performed at the maximum rekeying interval since they have maximum stability index and their possibility to join or leave the network is less. This periodic rekeying reduces the repeated rekeying process that further reduces the overhead. In rekeying technique, the multicast group key (GK_i) is rekeyed considering the three cases given above. The rekeying algorithm functions as follows [16].

According to the cases given above rekeying process is triggered. Initially, node N_i performs the ECDH key management agreement from leaf node to the source of multicast tree to obtain subgroup key cooperatively as

$$K_{N_i} + K_{N_{i+1}} + \cdots + K_{N_{n-1}} P. \quad (15)$$

Here, K_{N_i} is the leaf node, $K_{N_{n-1}}$ is the source, and P is the key generator in Diffe-Hellman. Finally, the generated subgroup chain reaches the source and it computes the new group key for the group. Once, the new group key is generated by the source, it unicasts it to the members securely.

Considering the tree structure given in Figure 4, node N_2 and N_5 are leaf nodes, N_7 is the parent node of nodes 2 and 5, and S is the multicast source. Assume N_2 invokes the rekeying process, and then the sequential process of rekeying is given below.

Step 1. N_2 generates subgroup key as $K_{N_2} + K_{N_5} P$ and transmits to N_7.

Step 2. Node N_7 computes the subgroup key as $K_{N_2} + K_{N_5} + K_{N_7} P$ and forwards to the source.

Step 3. Finally, the source computes cooperative subgroup key as $K_{N_2} + K_{N_5} + K_{N_7} + K_S P$ and then generates new group key as K_i' the source then unicasts the new group key securely to its member nodes.

4. Simulation Results

The proposed technique was simulated under different scenarios using varying number of receivers and varying the mobility of the nodes.

4.1. Simulation Model and Parameters. To analyze the performance of the proposed work NS2 [17] was used. In our simulation, the channel capacity of mobile hosts is set to the same value: 2 Mbps. We use the distributed coordination function (DCF) of IEEE 802.11 for wireless LANs as the MAC layer protocol. For multicasting, we used Multicast AODV (MAODV) [16] routing protocol. Simulations were carried out in 1500 meter × 1500 meter region for 50 seconds simulation time. We assume each node moves independently with the same average speed. All nodes have the same transmission range of 250 meters. In our simulation, the speed varied from 5 to 25 m/s and performance measured. The simulated traffic is Constant Bit Rate (CBR). In this simulation, we consider both the node capture and insider attacks. In node capture attack, a malicious attacker steals the credentials and secret keys from the legitimate nodes. An insider attacker is a malicious authenticated group member which may intimate false trust relations and injects false trust reporting. It may also inject packets *n* the network to disturb communications and consume the network resources. Our simulation settings and parameters are summarized in Table 1.

Table 1: Simulation parameters.

Number of receiver nodes	10, 20, 30, 40, 50
Area size	1500 × 1500
Mac	802.11
Radio range	250 m
Simulation time	50 sec
Traffic source	CBR
Rate	250 Kb
Mobility model	Random way point
Speed	5, 10, 15, 20 and 25.

4.2. Performance Metrics. We compare our Mobility Based Key Management Technique (MBKM) with the traditional GKMPAN [10] and efficient clustering scheme for group key management (ECGK) [18]. We evaluate mainly the performance according to the following metrics.

Average Packet Delivery Ratio. It is the ratio of the number of packets received successfully and the total number of packets sent.

Overhead. It is the control overhead (in terms of packets) occurred in keying and rekeying operations.

Packet Drop. It is the average number of packets dropped at each receiver.

Detection Accuracy. It is the ratio of number of attacks detected to the number of attacks performed.

Resilience. It is the ratio of fraction of data compromised to the fraction of nodes compromised.

4.2.1. Based on Receivers. In our first experiment, we vary the number of receivers per group as 10, 20, 30, 40, and 50 with speed 5 m/s.

(i) Comparison with GKMPAN. The proposed MBKM technique is compared with GKMPAN and the above performance metrics are evaluated by varying the group size.

Figures 6 and 8 present the packet delivery ratio and packet drop of both techniques, respectively, when the group size is increased from 10 to 50. From the figure, we can see that MBKM has 89% less packet drop than the existing GKMPAN techniques, since it assures high reliability using the strong nodes. Because of this reduced packet drop, the delivery ratio of the proposed MBKM is 23.57% higher than the GKMPAN technique. Figure 7 presents the control

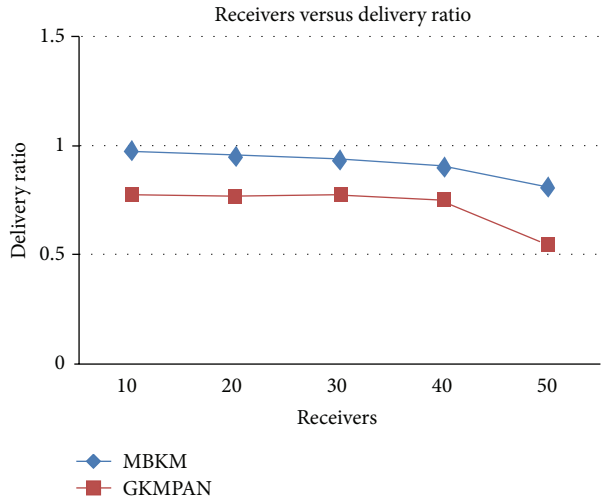

Figure 6: Comparison of delivery ratio with GKMPAN for varying receivers.

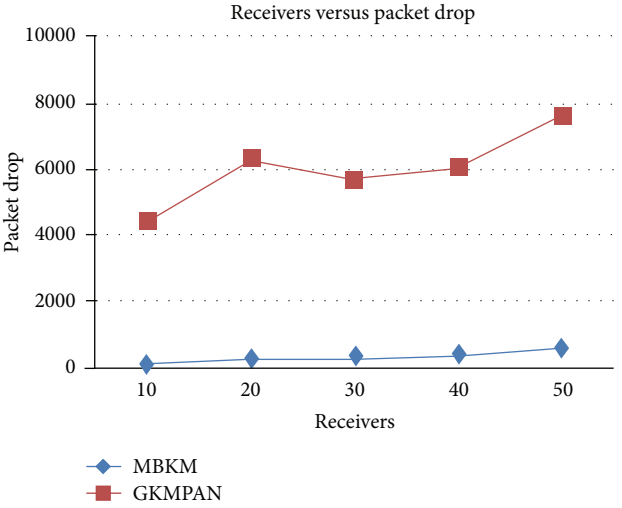

Figure 8: Comparison of packet drop with GKMPAN for varying receivers.

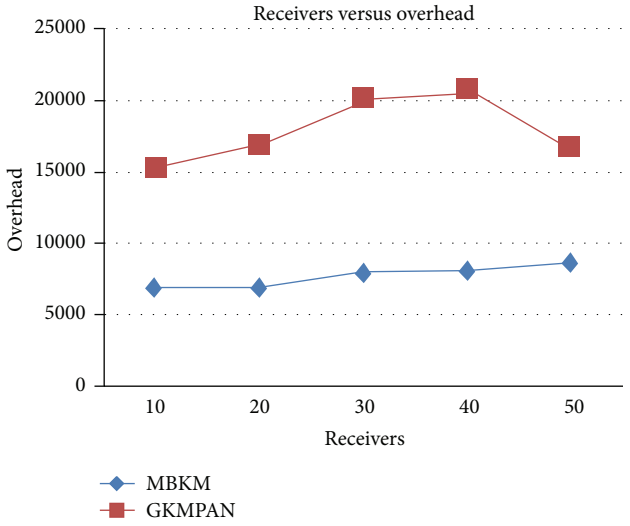

Figure 7: Comparison of overhead with GKMPAN for varying receivers.

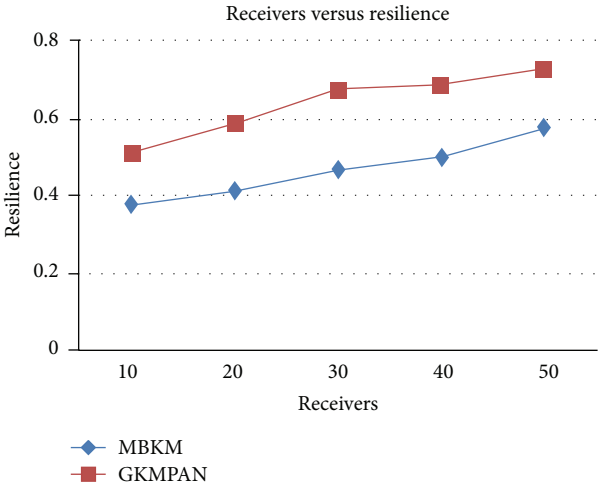

Figure 9: Comparison of resilience with GKMPAN for varying receivers.

overhead that occurred for both the techniques when the group size is increased. It can be seen that MBKM has 79.01% lesser overhead than the existing GKMPAN scheme, since it does not use the traditional multicast tree structure which involves large number of nodes. Figure 9 presents the results for resilience for both the techniques when the group size is increased. It can be seen that MBKM has 30.96% lesser resilience than GKMPAN, since it has efficient rekeying technique.

(ii) Comparison with ECGK. The proposed MBKM technique is compared with ECGK and the above performance metrics are evaluated by varying the group size. Figures 10 and 12 presents the packet delivery ratio and packet drop of both techniques, respectively, when the group size is increased from 10 to 50. From the figure, we can see that MBKM has 35.02% less packet drop than ECGK technique, since it assures high reliability using the strong nodes. Because of this reduced packet drop, the delivery ratio of the proposed MBKM is 1.82% higher than the ECGK technique.

Figure 11 shows the control overhead occurred for both the techniques when the group size is increased. It can be seen that MBKM has 15.32% lesser overhead than ECGK technique, since it does not use the traditional multicast tree structure which involves large number of nodes. Figure 13 presents the results for resilience for both the techniques when the group size is increased. It can be seen that MBKM has 16.51% lesser resilience than GKMPAN, since it has efficient rekeying technique.

4.2.2. Simulation Based on Node Speed. In our second experiment we vary the speed of the mobile node as 5, 10, 15, 20, and 25 m/s for 10 receivers. Figures 14 and 16 present the

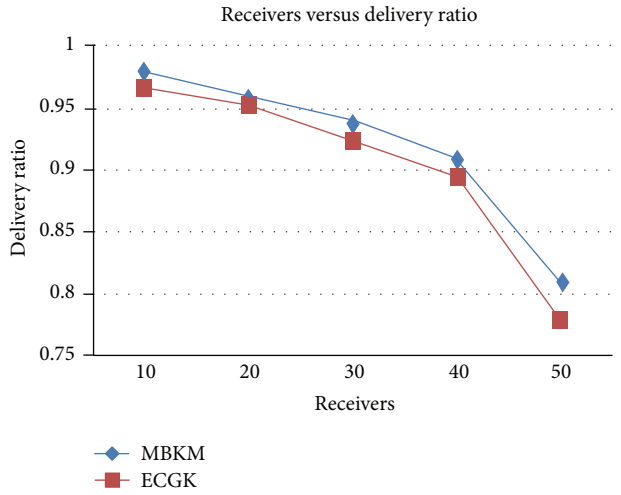

Figure 10: Comparison of delivery ratio with ECGK for varying receivers.

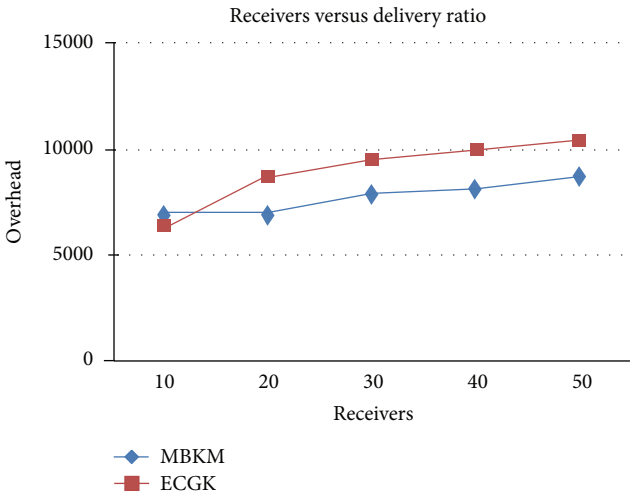

Figure 11: Comparison of overhead with ECGK for varying receivers.

packet delivery ratio and packet drop of both techniques, respectively, when the speed of the node is increased from 5 to 25 m/s. From Figure 11, we can see that the packet drop increases as the speed increases, due to disconnections and route breakages. But MBKM has 84% less packet drop than the existing GKMPAN techniques, since it uses stable and energy efficient nodes for routing. Because of this reduced packet drop, the delivery ratio of the proposed MBKM is 29% higher than the GKMPAN technique. Figure 15 presents the control overhead occurred for both the techniques when the group is increased. It can be seen that MBKM has 56% lesser overhead than the existing GKMPAN scheme, since it does not use the traditional multicast tree structure which involves large number of nodes.

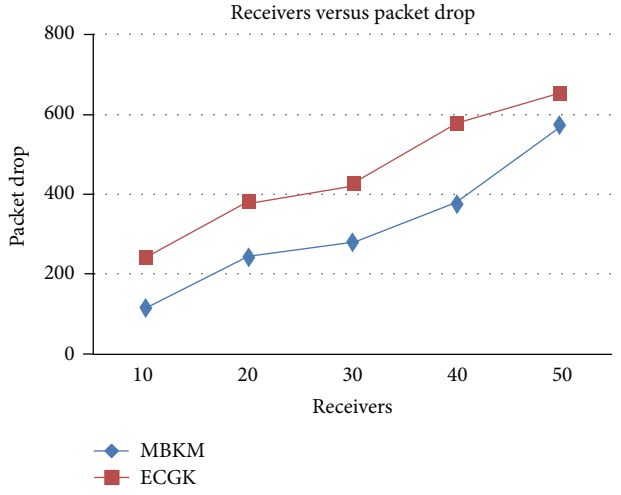

Figure 12: Comparison of packet drop with ECGK for varying receivers.

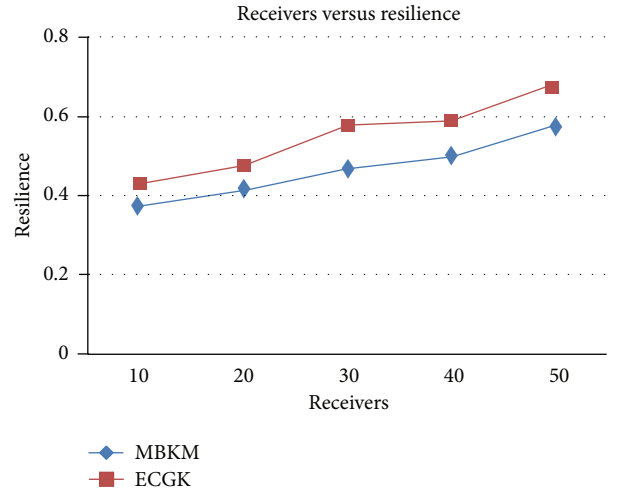

Figure 13: Comparison of resilience with ECGK for varying receivers.

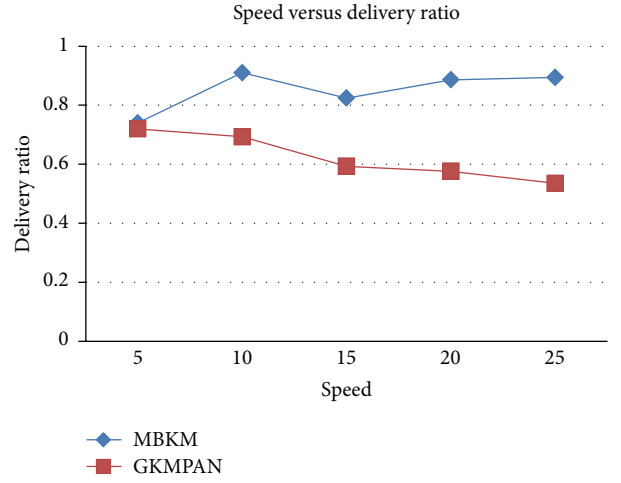

Figure 14: Speed versus delivery ratio.

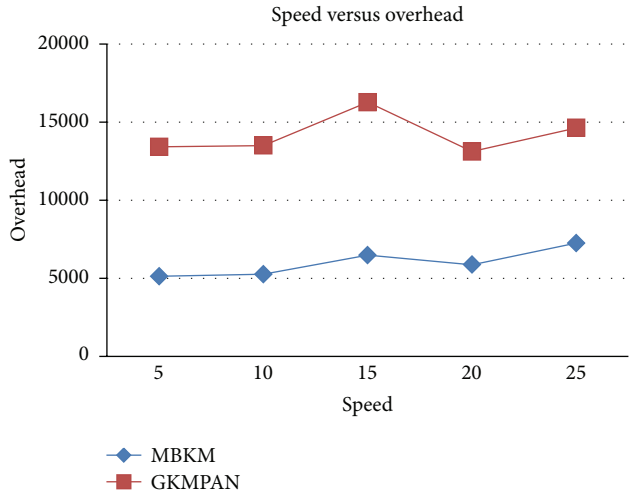

Figure 15: Speed versus overhead.

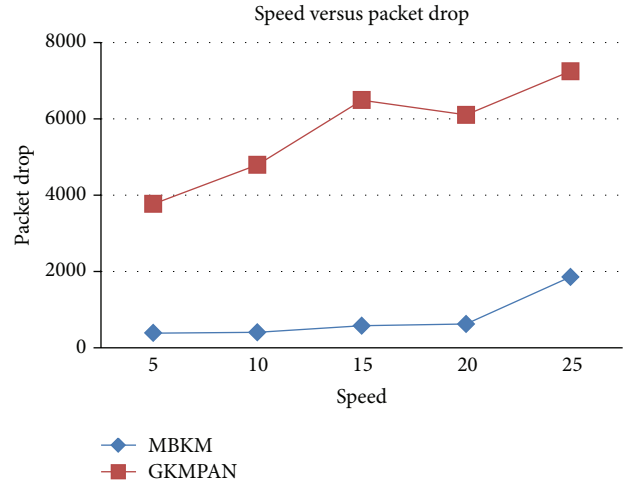

Figure 16: Speed versus drop.

5. Conclusion

In this work, mobility based key management technique is used for multicast security in MANET. Initially the nodes are categorized into strong and weak nodes according to their stability index. The stability index is estimated based on the link availability and mobility. A multicast tree is constructed such that for every weak node, there is a strong parent node. When any node desires to transmit a multicast data to destination, a session key based encryption technique is utilized. The rekeying process is performed periodically by the initiator node which is chosen among the strong nodes based on the reputation index. The rekeying interval is fixed depending on the node category. For the weak nodes, the initiators perform rekeying within minimum rekeying interval as they possess minimum stability index. Whereas, for the strong nodes, the initiators perform rekeying at the maximum rekeying interval since their stability index is more and the possibility of their position change due to mobility is less. This technique minimizes the repeated rekeying process that further minimizes the overhead. By simulation results proposed approach reduces the packet drop rate and improves the data confidentiality.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] L. Junhai, X. Liu, and Y. Danxia, "Research on multicast routing protocols for mobile Ad-hoc networks," *Computer Networks*, vol. 52, no. 5, pp. 988–997, 2008.

[2] M. Striki and J. S. Baras, "Key distribution protocols for secure multicast communication survivable in MANETs," in *Proceedings of the IEEE Military Communications Conference (MILCOM '03)*, Boston, Mass, USA, October 2003.

[3] C. Rajan and N. S. Shanthi, "Misbehaving attack mitigation technique for multicast security in mobile ad hoc networks (MANET)," *Journal of Theoretical and Applied Information Technology*, vol. 48, no. 3, pp. 1349–1357, 2013.

[4] R. Srinivasan, V. Vaidehi, R. Rajaraman, S. Kanagaraj, R. Chidambaram Kalimuthu, and R. Dharmaraj, "Secure group key management scheme for multicast networks," *International Journal of Network Security*, vol. 10, no. 3, pp. 205–209, 2010.

[5] L. Lazos and R. Poovendran, "Power proximity based key management for secure multicast in Ad hoc networks," *Wireless Networks*, vol. 13, no. 1, pp. 127–148, 2007.

[6] D. S. Devi and G. Padmavathi, "A reliable secure multicast key distribution scheme for mobile Adhoc networks," *World Academy of Science, Engineering and Technology*, vol. 56, pp. 321–326, 2009.

[7] S. Devaraju and G. Padmavathi, "Dynamic clustering for QoS based secure multicast key distribution in mobile Ad hoc networks," *International Journal of Computer Science Issues*, vol. 7, no. 5, pp. 30–37, 2010.

[8] B.-J. Chang and S.-L. Kuo, "Markov chain trust model for trust-value analysis and key management in distributed multicast MANETs," *IEEE Transactions on Vehicular Technology*, vol. 58, no. 4, pp. 1846–1863, 2009.

[9] D. Huang and D. Medhi, "A secure group key management scheme for hierarchical mobile Ad hoc networks," *Ad Hoc Networks*, vol. 6, no. 4, pp. 560–577, 2008.

[10] M. S. Bouassida and M. Bouali, "On the performance of group key management protocols in MANETs," in *Proceedings of the Joint Conference on Security in Network Architectures and Information Systems (SAR-SSI '07)*, pp. 275–286, Annecy, France, June 2007.

[11] J.-C. Lin, K.-H. Huang, F. Lai, and H.-C. Lee, "Secure and efficient group key management with shared key derivation," *Computer Standards & Interfaces*, vol. 31, no. 1, pp. 192–208, 2009.

[12] V. Sridhara and S. Bohacek, "Realistic propagation simulation of urban mesh networks," *Computer Networks*, vol. 51, no. 12, pp. 3392–3412, 2007.

[13] R. Biradar, S. Manvi, and M. Reddy, "Mesh based multicast routing in MANET: stable link based approach," *International Journal of Computer and Electrical Engineering*, vol. 2, no. 2, pp. 371–380, 2010.

[14] S. R. Zakhary and M. Radenkovic, "Reputation-based security protocol for MANETs in highly mobile disconnection-prone environments," in *Proceedings of the 7th International Conference on Wireless On-Demand Network Systems and Services (WONS '10)*, pp. 161–167, February 2010.

[15] *Elliptic Curve Cryptography, Version 2.0, Technical Guideline*, Bundesamt für Sicherheit in der Informationstechnik, 2012.

[16] H.-Y. Lin and T.-C. Chiang, "Efficient key agreements in dynamic multicast height balanced tree for secure multicast communications in Ad Hoc networks," *EURASIP Journal on Wireless Communications and Networking*, vol. 2011, Article ID 382701, 15 pages, 2011.

[17] Network Simulator, http://www.isi.edu/nsnam/ns/.

[18] K. Drira, H. Seba, and H. Kheddouci, "ECGK: an efficient clustering scheme for group key management in MANETs," *Computer Communications*, vol. 33, no. 9, pp. 1094–1107, 2010.