Secure Group-based Information Sharing in Mobile Ad Hoc Networks

Weichao Wang
Department of Software and Information Systems
University of North Carolina at Charlotte, USA
Email: weichaowang@uncc.edu

Yu Wang
Department of Computer Science
University of North Carolina at Charlotte, USA
Email: yu.wang@uncc.edu

***Abstract*—Secure multicast has become an important component of many applications in wireless networks. In this paper, we investigate secure intra and inter group information sharing in a network consisting of multiple node groups. We develop a mechanism for the establishment and maintenance of multicast structures, which enable flexible topology changes and efficient information distribution. We develop a key distribution and update method for secure information sharing in the same group and among different groups. It adopts polynomials to support the distribution of personal key shares and employs LKH (Logical Key Hierarchy) to achieve efficient key refreshment. We also investigate the overhead and safety of the proposed mechanism and demonstrate its advantages over previous approaches.**

I. Introduction

With the proliferation of mobile ad hoc networks, more and more organizations start to notice the advantages brought by information sharing among multiple such networks with overlapping physical coverage. For example, Phoenix Joint Terrorism Task Forces have initiated a program to integrate the communication and planning capabilities of fire, police, and emergency medical officials [1]. Enabling information sharing among ad hoc networks established by multiple agencies will drastically reduce the deployment and maintenance cost of each institute, and improve the accuracy and efficiency of collaborative efforts such as joint intrusion detection.

Before these benefits can be fully utilized, special care must be taken to preserve confidentiality since both information sharing and isolation must be enforced. For example, during an emergency event, a group of medical officials and FBI agents jointly form an ad hoc network. A medical official usually has lower security clearance than a FBI agent. When a physician observes a suspicious event and sends a message to report it to all FBI agents, none of other physicians should get access to this highly sensitive information. At the same time, when physicians exchange information about a patient on the network, it should not be accessed by FBI agents to preserve the patient's privacy. Therefore, *in an integrated ad hoc network, group-based data access must be controlled through security mechanisms*.

Enforcing security in these environments puts new challenges to researchers. First, a mobile node (e.g. a physician) should be able to initiate a multicast packet targeting at any node group (FBI agents or physicians) in the network. It is different from traditional multicast applications since both intra-group and inter-group communication must be protected. Second, membership changes among groups will bring new difficulties to access management. Finally, special properties of mobile ad hoc networks, such as network topology changes, must be properly handled.

A straight forward approach is to deploy a public-private key pair for each group [2]. For example, a physician may send a multicast message to all FBI agents by encrypting it with the public key of FBI. This method, however, may introduce several problems that will impact the network performance. First, even with the newly developed asymmetric encryption algorithms such as Elliptic Curve Cryptography [3], symmetric encryption still has its unique advantages in power consumption and computation overhead. Second, unless an authentication method is adopted, public key encryption will not provide any information about identity of the sender. Finally, the maintenance overhead for public-private secrets is usually heavy during group changes [4].

In this paper, we propose an approach to secure intra- and inter-group information sharing in an integrated ad hoc network containing multiple groups of wireless nodes. We first develop a method to construct and maintain the information sharing structures that can adapt to network topology changes caused by node movements. The approach enables individual nodes to efficiently inject and disseminate multicast traffic from various locations in the network. A key distribution and update method for securing multicast traffic among different groups is then presented. The approach enables efficient secret updates during group changes. The overhead and safety of the proposed approach are also investigated.

The contributions of the paper are as follows. First, the proposed mechanism for information sharing is different from multicast tree establishment and maintenance. It involves information transmission among multiple groups and supports information injection by individual nodes. Second, symmetric encryption is adopted for information protection, which avoids heavy computation and reduces information processing overhead. Third, analyses are conducted to demonstrate the improvements in efficiency and safety over previous approaches.

The remainder of the paper is organized as follows. Section II reviews previous research efforts. Section III presents the

The research is supported in part by the US National Science Foundation (NSF) under Grant No. DUE-0633143 and CNS-0721666.

formation of information sharing structures. We are especially interested in structure updates caused by node movements and efficient dissemination of multicast data. In Section IV, we introduce the key management and update method. In Section V, we analyze the security features of the proposed approach. Section VI concludes the paper.

II. Related Work

Group communication has become an important component of many applications in mobile wireless networks. Below we summarize previous research efforts in two fields.

Multicast Structure Formation: Various approaches have been proposed to improve the efficiency and security of group communication in wireless networks. They target at special features such as node mobility and frequent link changes. The limited resources on computation capability, energy, and available bandwidth are also considered. LKHW [5] extends the application of Logical Key Hierarchy (LKH) to sensor networks and enforces both backward and forward secrecy. In [6], a node joins a multicast group by attaching to the closest member so that a physical security tree structure is constructed. The joining and leaving operations are managed by the upstream node in the tree. [7] and [8] consider the location information and different models of signal attenuation when constructing the multicast hierarchy so that a better energy efficiency can be achieved. To reduce the maintenance overhead, stateless multicast protocols [9], [10] and overlay multicast protocols [11], [12] have been developed.

Key Management for Group Communication: While key management for multicast has been well studied for wired networks, several special features of wireless networks raise new challenges. For example, in a mobile network, the key distribution structure may change over time because of node movement. Both CKDS [13] and GKMPAN [14] avoid the adoption of LKH. CKDS uses a matrix-like key distribution structure in which the unknown secrets to the revoked nodes can be used to distribute new keys. GKMPAN depends on TESLA [15] for the authentication of multicast packets and group key updates. It assumes high node mobility and provides the desirable stateless property. In [16], a subset-cover framework is proposed to achieve the goal. The approaches in [17], [18] take a tree-based structure to distribute keys and achieve resistance to packet loss by appending additional information to subsequent messages.

III. Formation of Information Sharing Structures

The proposed research focuses on three problems in constructing inter- and intra-group multicast structures: (1) since node movement may lead to frequent topology changes, updates to multicast structures must be handled in a distributed manner; (2) since both intra and inter-group multicast packets need to be distributed through the constructed structure, a node must be able to locate not only members of its own group, but also those of other groups; (3) since multicast traffic may be initiated by any node, the dissemination method is different from traditional multicast applications that involve a single sender and multiple receivers. Therefore, an efficient data distribution mechanism must be designed. Below we discuss solutions to these problems respectively. Without losing generality, we assume that there are three node groups G_1, G_2, and G_3 in the network and both intra and inter group multicast traffic must be protected.

A. Localized Updates to Multicast Structures

The network topology of a mobile ad hoc network is continuously changing because of node movements, which also leads to changes of the multicast structure. If all such changes are handled by one or a few special nodes in the group, they will soon become overwhelmed. At the same time, one of these special nodes may be compromised or become disconnected from the network, leading to the single point of failure. To address these problems, we propose a distributed approach. During the joining event, a node initiates a localized broadcast to locate the 'closest' node that is already in the multicast structure of the target group. Here the 'closeness' may represent distance in hops or other measures such as power consumption or node workload. Through attaching to this intermediate node, the node becomes a new member of the multicast structure. When a node becomes detached from the multicast structure, it will notify the downstream members to locate a new attaching point. Its upstream node will detect the link change and stop sending traffic along the path. Below we describe the details of the joining and leaving events.

Steps of Joining Events

(1) **Localized Broadcast of Joining Request**: During a joining event, a node initiates a localized broadcast to locate a member already in the target group. The node needs to prove its eligibility by demonstrating its knowledge of the group key, which will be discussed in detail in the next section. Similar to AODV [19], the time-to-live (TTL) value of the request will start from a small value and keep increasing until an attaching point is found. Each request will be uniquely identified by the sender's ID and a sequence number so that every receiver will forward the request at most once.

(2) **Handling Reply Packets**: When a node receives a joining request, it first examines whether or not it is a member of the target group. If not, it reduces the TTL, attaches its ID, and rebroadcasts the request. If the node is a member and has a connection to the multicast structure of the group, it first verifies the eligibility of the requester. If the verification succeeds, the node then unicasts a reply back to the requester. The reply contains its node ID, information of the multicast structure, and proof of the knowledge of the group key. When the original requester receives this reply, it is attached to the multicast structure of the group.

(3) **Mutual Authentication**: During a joining event, the requester and the replier must verify the eligibility of each other. In this paper, the goal is achieved through verification of the knowledge of the group encryption key. The requester and the verifier can authenticate each other through the encryption of a pair of freshly generated nonces so that the procedure will be robust against resend attacks.

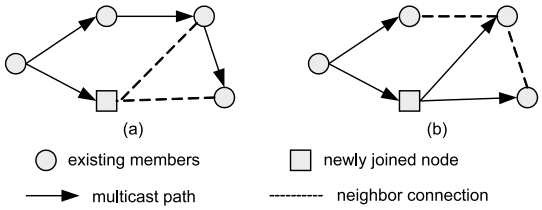

Fig. 1. Multicast paths (a) before and (b) after introduction of a new member.

(4) **Multicast Structure Optimization**: Under many conditions, a newly joined node will enable optimization of the multicast structure. For example, Fig. 1 illustrates how a new member can change the overall length of multicast paths. The nodes may conduct optimization based on the information collected through multiple joining replies.

Steps of Leaving Events

When the connection between two members of a group breaks, suitable changes must be conducted to preserve data dissemination paths.

(1) **Updating Data Dissemination Paths**: When a connection in the multicast structure breaks, the two end nodes detect this change and update their routing tables. All paths depending on this connection will be aborted, and the neighboring nodes will be notified. To avoid formation of loops, methods such as split horizon and reverse poisoning can be adopted.

(2) **Reconnecting to Multicast Groups**: When a connection in the multicast structure breaks, one or multiple nodes will become disconnected and they must locate a new attaching point. Although every node may adopt the localized broadcast method described in Section III.A to accomplish this task, a large amount of communication overhead will be generated. To reduce control traffic in the network, the nodes may adopt a method similar to the local repair approach of AODV.

B. Dissemination of Inter and Intra Group Multicast Packets

The major differences between the investigated application scenarios and traditional one-sender-multiple-receiver multicast model are as follows: (1) there are multiple node groups coexisting in the network and both intra and inter group multicast traffic must be protected; and (2) instead of a unique source, every node can initiate a multicast packet targeting at members of any group in the network. To suit these special properties, the following methods will be adopted.

Locating Traffic Injection Points: In addition to joining the multicast structure of its own group, a node must also locate members of other groups through which it can inject traffic into. To reduce control traffic for maintaining such connections, we propose to use an on-demand method. When node v in group G_1 wants to send a packet to all members of group G_2, it first initiates a localized broadcast to find a member in G_2. The request contains the identity of the node, its current group, and the target group. When a nearby node u in G_2 receives the request, it sends back a reply through unicast. As we will demonstrate in Section IV, the two nodes will use inter-group encryption keys to verify the identities of each other. If the verification succeeds, v sends the packet to u, who will further distribute the message through the multicast structure of G_2.

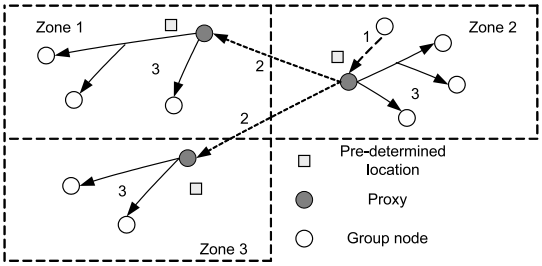

Fig. 2. Multicast packets dissemination procedures. (1) send the packet to a proxy; (2) dissemination among proxies; and (3) multicast to group members.

Data Dissemination Procedures: Since a multicast packet may originate from any node in the network, a tree-based structure will not enable an efficient distribution procedure. At the same time, if the root node is disconnected from the network, the distribution procedure will be impacted. Therefore, we propose to integrate efficient location based routing [20], [21] with multi-source multicast for MANETs [22]–[24] to solve this problem. Below we use a location-aware environment as an example to illustrate the data dissemination procedure.

In a location aware environment, we divide the network area into zones and select a pre-determined central position for each zone. For every node group, a member close to the central position of a zone will serve as the proxy of this group in this zone and organize group members in this zone to form a traditional single source multicast structure. The proxies form a high level overlay. Every node locates the closest proxy of its group, joins its multicast structure, and receives data from it. Sending data occurs in three steps, as illustrated in Fig. 2. First, a source unicasts the packet to its proxy through location based routing. Next, the proxy relays this packet by unicasting it to all other proxies. Finally, each proxy delivers the packet to group members in its multicast structure.

Linking roots of multicast structures to pre-determined positions in the network area can reduce overhead of a node to locate a proxy. To preserve stability of multicast structures, a distance threshold d_t will be adopted: only when the distance between the pre-determined central position and the current proxy becomes larger than d_t, a new proxy will be generated. This method can be easily applied to inter-group multicast traffic: the source will first unicast the data to a node in the target group, then the same procedure can be followed.

When the mobile nodes are not aware of their locations, we may choose a group of nodes to serve as proxies. The choosing criteria may include remaining energy, processing capability, or trust level. During the data dissemination procedure, the source will first send the packet to its proxy through the upstream path, then the same operations can be adopted.

IV. Key Management Mechanisms

A. Notation

We assume that every node is uniquely identified by a node ID u, where $u \in \{1 \cdots n\}$ and n is the total number of

nodes. The nodes are divided into d different groups, which are represented by G_1 to G_d, respectively. All operations described in the protocol will take place in a finite field F_q, where q is a prime number with a large enough value.

We assume that in a group G_i, at most t mobile nodes can collude together and attempt to compromise the key management mechanism. Since a mobile node can switch its group dynamically and rejoin the current group later, the group membership changes are not monotonic. When a group change happens, secrets must be updated to preserve forward and backward secrecy. We assume that there is a $group\ manager$ in charge of generating and distributing new keys. The role of group manager can be jointly played by multiple nodes in a distributed manner, which will be discussed in Section V.

We use $E_k(msg)$ and $D_k(msg)$ to represent the encryption and decryption of the message msg with a symmetric key k, respectively. We use $h(x)$ to represent a t-degree polynomial in $F_q[x]$, and $h(u)$ is the value of the function at point u.

We assume that a packet has the format ($sender$, $receiver$, $objective$, $data\ contents$, $integrity\ protection$). If a packet has a group name as the $receiver$, it is a multicast message that targets at all current members of the group.

B. Secure Group Communication

During the network initiation procedure, every node will receive a set of secret keys from the $group\ manager$ through a secure channel such as the physical contact before deployment. These keys can be divided into two groups: traffic encryption keys (TEK) to protect multicast packets, and key encryption keys (KEK) to support secret refreshment. Without losing generality, we assume that the nodes are divided into three groups G_1, G_2, and G_3. Below we use a node u in group G_2 as an example to illustrate the secret keys that it holds.

We assume that node u can communicate with the $group\ manager$ securely. This can be achieved through a pairwise key $K_{u,GM}$ shared between the two entities. As a member of G_2, u will get a copy of the symmetric group key K_2 which is used to encrypt and decrypt the multicast traffic within the group. Here the index '2' represents the group number.

We use t-degree polynomials $h(x)$ to determine the personal key shares and protect inter-group multicast traffic. As a member of G_2, u must be able to recover multicast packets sent by the nodes in G_1 and G_3. Therefore, it will be aware of two such functions, $h_{2,1}(x)$ and $h_{2,3}(x)$. Here the first and second indexes represent the destination and source groups of the multicast packets, respectively. For example, $h_{2,1}(x)$ is the polynomial to determine the personal key shares of the members in G_1 to send multicast packets to G_2. A node v in G_1 will get its personal key share $h_{2,1}(v)$ from the $group\ manager$. When it wants to send a multicast packet msg to the members in G_2, it will send out $(v, G_2, E_{h_{2,1}(v)}(msg, H(msg)))$. Since every node in G_2 knows $h_{2,1}(x)$, it can calculate the personal key share $h_{2,1}(v)$ by applying v to the polynomial and recover the information. Similarly, u is aware of the polynomial $h_{2,3}(x)$ so that it can decrypt multicast messages from the members in G_3. To enable node u to send multicast packets to the members in G_1 and G_3, it will get two personal key shares $h_{1,2}(u)$ and $h_{3,2}(u)$ from the $group\ manager$.

Two advantages have been brought by the personal key shares determined by polynomials. First, for two different nodes v and w in G_1, they will have different personal keys $h_{2,1}(v)$ and $h_{2,1}(w)$ to encrypt multicast packets to G_2. Therefore, information isolation has been achieved, and only the sender and members in the target group can recover the packet. Second, it becomes more difficult for an attacker to impersonate another node in the same group unless it can collect $t+1$ personal keys and reconstruct the polynomial $h(x)$. Secret separation among the nodes in the same group is especially valuable to wireless networks that consist of mobile nodes coming from different organizations. Under these conditions, the members in one group usually have weaker trust on the members in another group. Therefore, the mobile nodes want to confirm the identity of the source when an inter-group packet is received. Mechanisms to prevent inter-group impersonation will be discussed in later sections.

Table I summarizes the traffic encryption keys (TEK) held by node u and their usage. The key encryption keys and the refreshment operations will be discussed in the next part.

TABLE I
TEK KEYS HELD BY NODE u AND THEIR USAGE.

TEK Keys	Domain	Usage
K_2	F_q	group key for members of G_2
$h_{2,1}(x)$	t-degree polynomial in $F_q[x]$	polynomial to determine the keys for decrypting the multicast traffic from a node in G_1
$h_{2,3}(x)$	t-degree polynomial in $F_q[x]$	polynomial to determine the keys for decrypting the multicast traffic from a node in G_3
$h_{1,2}(u)$	F_q	personal key share to encrypt multicast traffic sent to the members of G_1
$h_{3,2}(u)$	F_q	personal key share to encrypt multicast traffic sent to the members of G_3

C. Key Updates and Revocation

When a group change happens, secrets must be updated to preserve forward and backward secrecy. Below we describe the approach based on LKH [25], [26]. Another approach based on our stateless key management method for inter-group communication [27] will be discussed in Section V.

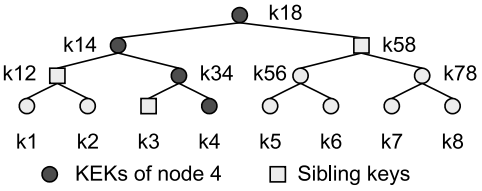

Fig. 3. Example of Logical Key Hierarchy.

In our key update mechanism, the wireless nodes in the same group will form a binary tree based on their node ID. Every mobile device is a leaf node in the tree and will get a copy of the key encryption keys (KEK) corresponding to each node in the path from the leaf node to the tree root. For

example, Fig. 3 illustrates a LKH containing eight nodes, and node four will get keys k_4, k_{34}, k_{14}, and k_{18}. At the same time, the KEKs corresponding to the sibling nodes of the path will form a special group: every other node in the tree will have at least one key from the group. For example, the sibling keys in Fig. 3 are k_3, k_{12}, and k_{58}. Every node except node four will have at least one key from the sibling keys, and they can be used for key updates when node four leaves the group. Below we use node u in G_2 as an example to illustrate the key update operations during a leaving event. The joining operations are very similar. We assume that the KEKs of node u are $s_i, i = 1 \cdots (\lceil \log_2 n \rceil + 1)$, where s_1 represents the leaf node key, and $\lceil \log_2 n \rceil + 1$ is the height of the tree. The sibling keys are $\overline{s_i}, i = 1 \cdots (\lceil \log_2 n \rceil)$.

When node u leaves group G_2, the following updates are required.

(1) Establishing new group key K'_{G2}. Node u should not get access to multicast traffic in G_2 after leaving the group. We can use sibling keys in the LKH to distribute the new group secret. The group manager (GM) will send out:

$$(GM,\ G_2,\ group\ key\ update\ for\ G_2,$$
$$E_{\overline{s_1}}E_{K_{G2}}(K'_{G2}),\ E_{\overline{s_2}}E_{K_{G2}}(K'_{G2}), \cdots,$$
$$E_{\overline{s_{\lceil \log_2 n \rceil}}}E_{K_{G2}}(K'_{G2}), GM's\ digital\ signature)$$

We use each of the sibling keys and the current group key K_{G2} to double encrypt the new group key K'_{G2} and distribute it to the group members. The remaining nodes in G_2 can recover the new secret and use it as a secure channel to communicate with each other.

(2) Establishing new LKH. The KEKs known to node u must be updated. The new secrets $s'_i, i = 1 \cdots (\lceil \log_2 n \rceil + 1)$ can be distributed to the remaining nodes in G_2 through double encryption. For example, the GM will send out:

$$(GM,\ G_2,\ KEK\ update\ for\ G_2,$$
$$E_{K'_{G2}}E_{s_1}(s'_1),\ E_{K'_{G2}}E_{s_2}(s'_2), \cdots,$$
$$E_{K'_{G2}}E_{s_{\lceil \log_2 n \rceil + 1}}(s'_{\lceil \log_2 n \rceil + 1}), GM's\ digital\ signature)$$

Only remaining nodes in G_2 that hold the old key s_i will be able to recover the new secret s'_i since they also know K'_{G2}.

(3) Establishing new polynomials $h'_{21}(x)$ and $h'_{23}(x)$. To prevent node u from getting access to multicast traffic from the members of G_1 and G_3, the polynomials $h_{21}(x)$ and $h_{23}(x)$ that determine their personal key shares must be replaced by new functions $h'_{21}(x)$ and $h'_{23}(x)$. In this part we describe how the new functions can be distributed to the nodes in G_2. The update operations for the nodes in G_1 and G_3 will be presented in the next part. The group manager will broadcast:

$$(GM,\ G_2,\ Polynomial\ update\ for\ G_2,$$
$$E_{K'_{G2}}(GM, G2, hash(h_{21}(x), h_{23}(x)), h'_{21}(x), h'_{23}(x)),$$
$$GM's\ digital\ signature)$$

Since only the remaining nodes in G_2 know the new group key K'_{G2}, they can decrypt the packet and get the new polynomials.

(4) Nodes in G_1 and G_3 getting new personal key shares. The members of G_1 and G_3 can acquire their new personal key shares in a distributed manner from the nodes in G_2 nearby. Below we use a node v in G_1 and w in G_2 as an example to illustrate how the personal key share can be updated.

(1) The group manager will broadcast an authenticated message and notify all nodes in G_1 and G_3 to acquire the new personal key shares. The ID of the expelled node will also be identified in the packet so that it will be avoided during the key refreshment procedures.

(2) After verifying the packet from the group manager, v will initiate a localized broadcast within a few hops and locate a node w in group G_2. It will then get $h'_{21}(v)$ by sending:

$$v \rightarrow w : (v, w, request\ for\ h'_{21}(v),$$
$$E_{h_{21}(v)}E_{h_{12}(w)}(v, w, R)\ )$$
$$w \rightarrow v : (w, v, reply\ for\ h'_{21}(v),$$
$$E_{h_{21}(v)}E_{h_{12}(w)}(w, v, h'_{21}(v), Hash(R, h'_{21}(v)))\ )$$

The random number R is used to guarantee the freshness of the reply. As a secure channel, v can get its new key share from w by using the dual encryption method $E_{h_{21}(v)}E_{h_{12}(w)}(\cdot)$. This procedure can be conducted through a multi-hop path.

(5) Preventing u from sending fake information to G_1 and G_3. As an expelled node, u still has the personal key shares $h_{12}(u)$ and $h_{32}(u)$, and it can use these keys to send false information to the members of G_1 and G_3. To prevent such scenarios from happening, the nodes in G_1 and G_3 will maintain a list of the expelled nodes until the new polynomials $h'_{12}(x)$ and $h'_{32}(x)$ are established. Since u will not get the updated personal key shares, it will not be able to generate false information to mislead the wireless nodes in the network.

V. Evaluating Proposed Approach

A. Overhead

We investigate the storage, computation, and communication overhead of the proposed mechanism and compare it to that of the public-private key approach discussed in Section I. In the proposed mechanism, both the required storage space and consumed bandwidth during secret updates for the TEKs and KEKs are $O(\log(n) + dt)$, while those of the public-private key method are only $O(\log(n) + d)$. Studying the results, we find that the distribution and storage of the t-degree polynomials explain a majority of the increased overhead. However, this increase can be justified as follows.

First, the costs of storage media for mobile devices keep decreasing. For example, with less than $20, a user can add 1G Byte storage space to her/his PDA. If we assume that there are 10 node groups in the network, the degree of polynomials is 80, and every key is 64 bits long, every node will need less than 20K Byte space for key storage. Therefore, the increased storage space will not impact users' costs to a large extent.

Second, compared to group changes in wireless networks, encryption and decryption of multicast data packets happen much more frequently. By replacing exponential computation

with a symmetric encryption procedure, we reduce data processing time at wireless nodes and improve system efficiency.

The proposed approach will also drastically increase the network lifetime under the same traffic scenarios. For example, in [28], investigators have measured power consumption of security operations in a PDA. They find that verifying a digital signature of ECDSA needs about 196 mJ, while the encryption of 1K Byte with DES will consume only 2.22 mJ. Therefore, the proposed approach will consume only about 1.1% energy to encrypt/decrypt a multicast data packet when it is compared to the public-private key method. This will not only prolong network lifetime, but also improve network robustness against power exhaustion attacks.

Third, the adoption of polynomials enables the distribution of personal key shares. Only the sender and members of the target group can read the information. It becomes more difficult for an attacker to impersonate another node even when additional authentication methods are not applied. The analysis in Section IV.C has also shown that by integrating the personal key shares of two nodes belonging to different groups, we can establish a secure communication channel between them.

B. Security and Robustness

Generating Group Managers. Group managers play an important role in the proposed mechanism. Below we describe the generation procedures under two conditions. If a predistributed infrastructure exists in the wireless network, the manager generation procedure can take advantage of those special nodes. For example, in a Cellular-Ad hoc integrated system, the base stations can maintain the member list of every group and generate new keys during group changes.

In a self-organized environment, a more complicated manager election or generation procedure must be adopted. One possible solution is a variation of the secure leader election algorithms for ad hoc networks [29]. The mobile nodes use a preference function that integrates multiple decision factors to represent the desirability of a candidate. The node that receives the most "votes" will become the manager.

For the simplicity of presentation, we have assumed a single group manager for each group. To improve robustness of the proposed mechanism and avoid single point of failure, distributed key management can be adopted. Multiple managers may perform equally or form a hierarchy to control the key distribution and update procedures. When a joining or leaving event happens, they can generate new keys in a collaborative manner [30] to prevent the security defections in one manager from degrading the safety of the whole system. Another advantage is that a wireless node has a higher probability to communicate with a manager locally, which will reduce the communication overhead caused by control traffic.

Defending Against Collusive Attacks. Wireless nodes in the network may collude to get illegal access to multicast traffic. The proposed mechanism is robust against collusive attacks from the malicious nodes in the same group. Mechanisms to defend against inter-group collusive attacks will be investigated in future work.

Malicious nodes in the same group can benefit from collusion by reconstructing polynomials of other groups. They can calculate personal key shares of other members and get illegal access to multicast traffic that is not destined to them. Since a t-degree polynomial is robust against the collusion of up to t compromised members, we can adjust the choice of this parameter based on the security requirements to balance the safety of the mechanism and the storage, computation, and communication overhead.

C. Future Extensions

Integrating Stateless Property. The movement of wireless nodes may lead to topology changes and network partitions in the system. Mobile nodes may miss some of the key update messages due to the error-prone transmission medium or unavailable paths. Therefore, the stateless property is highly desirable in wireless networks, which allows a mobile node to recover the current group key without requesting it from the manager. Several protocols that support this property [14], [31], [32] have been proposed in previous research.

We plan to integrate our stateless key update scheme for inter-group communication [27] with the proposed mechanism to improve its performance in highly mobile environments. The t-degree polynomials will be protected by masking functions and the wireless nodes with suitable pre-distributed information will be able to recover the lost secrets without interacting with managers.

VI. Conclusion

Secure multicast has become an important component of many applications in wireless networks. In this paper, we investigate secure information sharing in a network consisting of multiple node groups. We develop a mechanism for the establishment and maintenance of intra and inter group multicast structures. It enables flexible changes of multicast structures and efficient distribution of information. We develop a key distribution and update method for secure information sharing in the same group and among different groups. It adopts polynomials to support the distribution of personal key shares and employs LKH to achieve efficient key refreshment. The additional storage and communication overhead caused by the proposed mechanism has been properly justified. We also study the safety of the proposed mechanisms.

We plan to integrate the stateless property into the proposed mechanism. Additional research is also required to study the impacts of group changes and traffic patterns on its performance. The results will lead to a more robust and efficient information sharing mechanism among multiple groups in wireless networks.

References

[1] R. P. Churay, "Terrorism preparedness," Testimony before the House Committee on Government Reform, March 2002.

[2] S. Yi, P. Naldurg, and R. Kravets, "Security-aware ad hoc routing for wireless networks," in *Proceedings of ACM MOBIHOC*, 2001.

[3] L. Ertaul and N. Chavan, "Elliptic curve cryptography based threshold cryptography (ecc-tc) implementation for manets," *Int. Jour. of Computer Science and Network Security*, vol. 7, no. 4, pp. 48–61, 2007.

[4] C. Wolf, "Efficient public key generation for HFE and variations," in *Cryptographic Algorithms and Their Uses*, 2004, pp. 78–93.

[5] R. Pietro, L. Mancini, Y. Law, S. Etalle, and P. Havinga, "LKHW: A directed diffusion-based secure multicast scheme for wireless sensor networks," in *IEEE Int. Conf. on Parallel Processing Workshops*, 2003.

[6] T. Kaya, G. Lin, G. Noubir, and A. Yilmaz, "Secure multicast groups on ad hoc networks," in *Proc. of ACM workshop on security of ad hoc and sensor networks*, 2003, pp. 94–102.

[7] L. Lazos and R. Poovendran, "Energy-aware secure multicast communication in ad-hoc networks using geographic location information," in *IEEE Int. Conf. on Acoustics Speech and Signal Processing*, 2003.

[8] L. Lazos and R. Poovendran, "Location-aware secure wireless multicast in ad-hoc networks under heterogeneous pathloss," University of Washington," Technical Report UWEETR-2003-0012, 2003.

[9] L. Ji and M. Corson, "Differential destination multicast - a MANET multicast routing protocol for small groups," in *IEEE INFOCOM*, 2001.

[10] L. Ji and M. Corson, "Explicit multicasting for mobile ad hoc networks," *Mobile Networks and Applications*, vol. 8, no. 5, pp. 535–549, 2003.

[11] K. Chen and K. Nahrstedt, "Effective location-guided tree construction algorithms for small group multicast in MANET," in *Proc. of IEEE INFOCOM*, 2002, pp. 1180–1189.

[12] C. Gui and P. Mohapatra, "Efficient overlay multicast for mobile ad hoc networks," in *Proc. of IEEE WCNC*, 2003.

[13] M. Moharrum, R. Mukkamala, and M. Eltoweissy, "CKDS: an efficient combinatorial key distribution scheme for wireless ad-hoc networks," in *Proc. of IEEE IPCCC*, 2004.

[14] S. Zhu, S. Setia, S. Xu, and S. Jajodia, "GKMPAN: An efficient group rekeying scheme for secure multicast in ad-hoc networks," in *Proc. of Int'l Conf. on Mobile and Ubiquitous Systems*, 2004.

[15] A. Perrig, R. Canetti, D. Song, and D. Tygar, "Efficient and secure source authentication for multicast," in *Proc. of Network and Distributed System Security Symposium (NDSS)*, 2001.

[16] D. Naor, M. Naor, and J. Lotspiech, "Revocation and tracing schemes for stateless receivers," in *CRYPTO'01, LNCS 2139*, 2001, pp. 41–62.

[17] A. Perrig, D. Song, and J. Tygar, "ELK, a new protocol for efficient large-group key distribution," in *IEEE Symp. on Security and Privacy*, 2001.

[18] C. Wong and S. Lam, "Keystone: A group key management service," in *Proceedings of International Conference on Telecommunication*, 2000.

[19] C. E. Perkins, E. M. Belding-Royer, and S. Das, "Ad hoc on demand distance vector (aodv) routing," IETF RFC 3561, 2003.

[20] Y.-B. Ko and N. H. Vaidya, "Location-aided routing (lar) in mobile ad hoc networks," *Wireless Networks*, vol. 6, no. 4, pp. 307–321, 2000.

[21] B. Karp and H. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in *Proc. of ACM MobiCom*, 2000.

[22] Y.-Y. Su, S.-F. Hwang, and C.-R. Dow, "An efficient multi-source multicast routing protocol in mobile ad hoc networks," in *International Conference on Parallel and Distributed Systems*, 2005, pp. 8–14.

[23] D. Zappala and A. Fabbri, "Using ssm proxies to provide efficient multiple-source multicast delivery," in *IEEE Globecom*, 2001.

[24] M. Ripeanu, A. Iamnitchi, I. Foster, and A. Rogers, "In search of simplicity: A self-organizing group communication overlay," in *Int. Conf. on Self-Adaptive and Self-Organizing Systems*, 2007, pp. 371–374.

[25] C. Wong, M. Gouda, and S. Lam, "Secure group communications using key graphs," *IEEE/ACM Tran. Net.*, vol. 8, no. 1, pp. 16–30, 2000.

[26] D. Wallner, E. Harder, and R. Agee, "Key management for multicast: Issues and architectures," IETF RFC 2627, 1999.

[27] W. Wang and T. Stransky, "Stateless key distribution for secure intra and inter-group multicast in mobile wireless networks," *Elsevier Computer Networks*, vol. 51, no. 15, pp. 4303–4321, 2007.

[28] N. Potlapally, S. Ravi, A. Raghunathan, and N. Jha, "Analyzing the energy consumption of security protocols," in *Proc. of International Symposium on Low Power Electronics and Design*, 2003.

[29] S. Vasudevan, B. DeCleene, N. Immerman, J. Kurose, and D. Towsley, "Secure leader election algorithms for wireless ad hoc networks," in *IEEE DARPA Information Survivability Conf. and Exposition*, 2003.

[30] C. Tang, A. T. Chronopoulos, and C. S. Raghavendra, "Soft-timeout distributed key generation for digital signature based on elliptic curve d-log for low-power devices," in *Int. Conf. on Security and Privacy for Emerging Areas in Communications Networks*, 2005, pp. 353–364.

[31] D. Liu, P. Ning, and K. Sun, "Efficient self-healing group key distribution with revocation capability," in *Proc. of ACM conference on Computer and communications security*, 2003, pp. 231–240.

[32] J. Staddon, S.Miner, M. Franklin, D. Balfanz, M. Malkin, and D. Dean, "Self-healing key distribution with revocation," in *Proc. of IEEE Symposium on Security and Privacy*, 2002.