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*—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 enables 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

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. As an 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].

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, in an ad hoc network jointly formed by a group of medical officials and FBI agents, 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. 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 should be able to initiate a multicast packet targeting at any node group in the network. 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.

Using a different public/private key pair for each group will solve this problem. This method, however, may introduce several problems. First, even with Elliptic Curve Cryptography, 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.

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 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 presented. The approach enables efficient secret updates during group changes. The overhead and safety of the proposed approach are also investigated.

The remainder of the paper is organized as follows. Section II reviews previous research efforts. Section III presents the formation of information sharing structures. We focus on structure updates caused by node movements and dissemination of multicast data. Section IV introduces the key management and update method. Section V analyzes the security features of the proposed approach. Section VI concludes the paper.

II. Related Work

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, link changes, and limited resources. LKHW [2] extends the application of Logical Key Hierarchy to sensor networks and enforces both backward and forward secrecy. In [3], a node joins a multicast group by attaching to the closest member so that a physical security tree structure is constructed. To reduce the maintenance overhead, stateless multicast protocols [4], [5] and overlay multicast protocols [6], [7] have been developed.

Several special features of wireless networks raise new challenges to secure multicast. For example, in a mobile network, the key distribution structure may change over time because of node movement. Both CKDS [8] and GKMPAN [9] avoid the adoption of LKH. In [10], a subset-cover framework is proposed to achieve the goal. The approaches in [11], [12] take a tree-based structure to distribute keys and achieve resistance to packet loss by appending additional information to subsequent messages. A generic approach to key management in access hierarchies is proposed in [13].

III. Formation of Information Sharing Structures

The proposed research focuses on three problems in constructing inter- and intra-group multicast structures: (1) localized updates to multicast structures; (2) locating data injection points for different groups; (3) efficient data dissemination through multiple sources. 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, and both intra and inter group multicast traffic must be protected.

A. Localized Updates to Multicast Structures

Node movements in a mobile ad hoc network will lead to changes of multicast structures. If all such changes are handled by a few special nodes, they will soon become overwhelmed. To avoid such scenarios, we propose a distributed approach. During the joining event, a node locates the ‘closest’ entity 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 the next section. The time-to-live (TTL) value of the request can be used to control the propagation area. 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 examines whether or not it is a member of the target group. If not, it reduces the TTL and rebroadcasts the request. If the node is a member and has a connection to the multicast structure of the group, it will verify the eligibility of the requester and unicast a reply back to it. The reply contains information of the multicast structure and proof of the knowledge of the group key. When the requester receives this reply, it is attached to the multicast structure of the group.

(3) **Mutual Authentication and Multicast Path Optimization**: During a joining event, the requester and the replier must verify the eligibility of each other. They can achieve this goal through the encryption of a pair of freshly generated nonces so that the procedure will be robust against resend attacks. Under many conditions, a newly joined node will enable optimization of the multicast structure by reducing the length of data propagation paths. The nodes may conduct such optimization based on the information collected through multiple joining replies.

Steps of Leaving Events

(1) **Updating Data Dissemination Paths**: When a connection in the multicast structure breaks, the two end nodes will detect this change and update their routing tables. All paths using 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 above, a large amount of communication overhead will be generated. To reduce control traffic in the network, the nodes may adopt a localized repair approach.

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 send 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: To enable a node to locate members of different groups through which it can inject traffic into, 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. When a nearby node u in G_2 receives the request, it sends back a reply. 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.

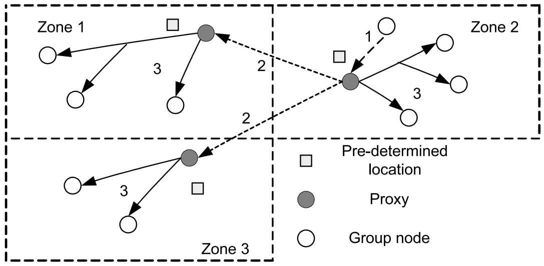

Fig. 1. Multicast packets dissemination procedures. (1) send the packet to a proxy; (2) dissemination among proxies; and (3) multicast to group members.

Data Dissemination Procedures: Since a multicast packet may originate from any node in the network, a tree-based structure will not serve as an efficient distribution procedure. We propose to integrate location based routing [14], [15] with multi-source multicast for MANETs [16], [17] to solve this problem. Below we use a location-aware environment as an example to illustrate the data dissemination procedure.

We divide the network area into zones and select a predetermined 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 and receives data from it. Sending data occurs in three steps, as illustrated in Fig. 1. First, a source unicasts the packet to its proxy through location based routing. Next, the proxy relays this packet 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 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.

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 to compromise the key management mechanism. A mobile node can switch its group dynamically. 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 generation of group managers will be discussed in Section V. We use $E_k(msg)$ and $D_k(msg)$ to represent the encryption and decryption of msg with a symmetric key k. 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 $group\ manager$. 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 shares a pairwise key $K_{u,GM}$ with the group manager. 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 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 $h_{2,1}(v)$ and recover the information. 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 $h_{3,2}(u)$.

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. 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 $h(x)$. Table I summarizes the traffic encryption keys (TEK) held by node u and their usage.

TABLE I
TEK keys held by node u and their usage.

TEKs	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. Another approach based on our stateless key management method for inter-group communication [18] will be discussed in Section V.

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. 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, and they can be used for key updates when group changes happen. 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:

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

We use each of the sibling keys and the current group key K_{G2} to doubly 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 for connection.

(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:

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

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 G_1 and G_3, the polynomials $h_{21}(x)$ and $h_{23}(x)$ 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:

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

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.

(2) Node v will initiate a localized broadcast and locate node w in group G_2. It will then get $h'_{21}(v)$ by sending:

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

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

(5) Preventing u from sending fake information to G_1 and G_3. 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 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.

V. Evaluating Proposed Approach

A. Overhead

In this section, we investigate the storage, computation, and communication overhead of the proposed mechanism. In the proposed approach, both the required storage space and consumed bandwidth during secret updates for the TEKs and KEKs are $O(\log(n) + dt)$. We find that the distribution and storage of the t-degree polynomials explain a majority of the overhead. However, this cost 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.

Different methods can be used to reduce communication overhead. For example, we can apply one way hash functions to calculate new personal key shares when a joining event happens. In this way we can reduce about half of the key management traffic. As another example, multiple messages described in section IV.C can be merged into one packet to reduce control traffic. We can conduct periodic secret updates to avoid the KEK distribution after every group change.

Second, compared to group changes in wireless networks, encryption and decryption of multicast data packets happen much more frequently. Through using symmetric encryption, we can reduce data processing time at wireless nodes, improve system efficiency, and increase the network lifetime under the same traffic scenarios.

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.

B. Security and Robustness

Generating Group Managers. Group managers play an important role in the proposed mechanism. 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 [19]. 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.

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. We have developed a method to defend against impersonation attacks conducted by nodes in another group [18]. The basic idea is to let the sender of a packet prove its knowledge of the polynomials belonging to that group. The verification parties are dynamically determined based on the data packet contents to avoid collusive attacks.

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 [9], [20], [21] have been proposed in previous research.

We plan to integrate our stateless key update scheme for inter-group communication [18] 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 for MANETs.

Acknowledgment

The research is supported by the National Science Foundation (NSF) Grant DUE-0754592 and CNS-0721666.

References

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

[2] 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.

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

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

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

[6] 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.

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

[8] M. Moharrum, R. Mukkamala, and M. Eltoweissy, "Ckds: an efficient combinatorial key distribution scheme for wireless ad-hoc networks," in *IEEE ICPCC*, 2004, pp. 631–636.

[9] 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 International Conference on Mobile and Ubiquitous Systems: Networking and Services*, 2004, pp. 42–51.

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

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

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

[13] M. Atallah, K. Frikken, and M. Blanton, "Dynamic and efficient key management for access hierarchies," in *ACM CCS*, 2005, pp. 190–202.

[14] 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.

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

[16] 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.

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

[18] 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.

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

[20] 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.

[21] 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.