Poster: Secure Sharing of Location Information over Ad Hoc Networks: An Application of Secure Multicast

Guevara Noubir
College of Computer Science
Northeastern University
noubir@ccs.neu.edu

Guolong Lin
College of Computer Science
Northeastern University
lingl@ccs.neu.edu

Mikhail Golitsine
College of Computer Science
Northeastern University
golitsin@ccs.neu.edu

Nick Plante
College of Computer Science
Northeastern University
nap@ccs.neu.edu

ABSTRACT

In this paper we address the problem of secure multicast of data streams over a multihop wireless ad hoc network. We propose a key management protocol that aims at solving problems that are specific to ad hoc networks such as mobility, unreliable links, and multihop communication cost. The main idea is to have group members actively participate to the multicast group security therefore reducing the communication and computation load on the source. We also extend the proposed basic secure multicast tree using multi-link capability combined with a *k*-out-of-*N* coding approach. Since the group security is distributed among the group members we propose a service right certificate to verify that a node is authorized to join the group and also corresponding revocation mechanism. We simulated our protocol within the *ns*-2 environment under various mobility, group size, and group dynamic scenarios. Our preliminary simulation results indicate that the communication cost and join delay of the protocol scale well when the group size and nodes mobility increase.

Categories and Subject Descriptors

D.3.3 [**Computer-Communications Networks**]: Network Protocols.

General Terms

Reliability, Security.

Keywords

Secure mutlicast, multihop ad hoc, MANET, tracking.

1. INTRODUCTION

One important research problem for secure dissemination of information over wireless multihop ad hoc networks (MANET) is how to restrict the information access to the group of authorized nodes. The data information has to be encrypted and only authorized users should be able to decrypt it. The security of the group has to be maintained when new members join or leave. The problem can be defined as follows: given one source multicasting a stream of data and multiple receivers that join and leave the multicast session. The goal is to design a low bandwidth/delay protocol that allows authorized nodes and only authorized nodes to access the data stream multicast by the source node. The underlying communication network is a multihop wireless ad hoc network with mobile nodes. This protocol was designed for secure tracking and monitoring of mobile nodes interconnected by a MANET. The testbed that will be demonstrated is composed of a set of iPAQ PDAs and laptops, equipped with wireless interfaces (IEEE802.11) and location acquisition interface (Compact Flash GPS). The nodes are running linux. The poster draft provides more information on the application, testbed, and architecture. In this summary we focus on the secure multicast protocols, we do not address the problem of multicast routing of the data information. The multicast data can take the same path as the security traffic or a different path. In which case the security traffic caries the key to decrypt the data packets. The authorized nodes have a *service access right* certificate that allows them to prove that they are authorized to access the data stream.

Several secure multicast solutions were proposed in the past considering various constraint. Due to space limitations we do not include a list of references to existing protocols but only summarize their limitations. These algorithms are not adequate for multihop wireless ad hoc networks because they do not consider the specific constraints of such networks namely:

Mobility and wireless: routes between nodes change with time and links are unreliable resulting in higher packet. The loss of a packet that contains information to update a group key will prevent the node from updating the group key. This has the same effect as excluding this node from the group. In some existing schemes this would results in requiring the node to send a new join request.

Multihop: the cost of multicasting any packet depends on the number of hops. This is an especially important constraint because of the scarcity of radio resources.

Ad hoc: nodes have limited computation power, on the other hand nodes can play an active role in the group security.

2. Physical security group tree and group discovery

In this section, we describe the basic scheme for secure multicast over MANET. It is based on maintaining a physical security tree of the group members. This uses the idea of the *Iolus* approach to delegate group control. However, in our approach we take into account MANET constraints. Joining members dynamically discover and attach to the "best-closest" tree node. The security multicast tree is used to securely forward the group key to authorized members. Leaving members inform

MobiHoc'03, June 1-3, 2003, Annapolis, Maryland, USA.

downstream nodes that they will be soon leaving the group and request them to attach to another node. Finally, the security multicast tree is optimized in real-time.

Joining steps:

- **Broadcast Group Join Request:** the parameters of this request are *group_id*, *TTL, and SN*. The *TTL* value is dynamically set and, the sequence number *SN* is used to avoid multiple forwarding.
- **Receive Group Join Replies:** nodes that are already within the security multicast tree send replies to the requester. Unless if the number of connected nodes to them exceeds some threshold. The replies contain information about: number of hops to the source, logical path to the source (the sequence of group members that lead to the source will be used in a handover to avoid loops), path quality to the source, and number of nodes already connected to this node. The requesting node will initiate a registration with the sender of the most satisfactory reply (in terms of aggregate path quality).
- **Authentication, registration, and key establishment:** the requester and intermediate node will first *mutually* authenticate each other. The authentication process will lead to the establishment of a shared key. Then, they will both check that they are allowed to access this information. This proof of access right is done using a service-access certificate.
- **Tree optimization:** once registered the newly joining node can send a *path_optimization* message to nodes that are already in the tree but could optimize their path by attaching to the joining node. The joining node knows that from the request reply that they sent to him.
- **Receive encrypted data.** The joining node can start receiving the encrypted data.
- **Receive tree update events.** Three types of tree update events can be received: *Group Join Requests*, *Handover Requests*, *Path Quality Drop*.

Leaving steps:

- **Inform downstream nodes.** The depending downstream nodes should initiate a handover and send a handover complete message once reconnected to the tree.
- **Inform upstream node.** Once all downstream nodes are reconnected or after a timout, the leaving node requests its upstream node to disconnect him from the tree.

3. Multi-link group attachment

The purpose of this protocol is to increase the reliability of a multicast tree. The basic idea is to maintain more than one upstream node. This makes sense if the goal is to increase the reliability of some sensitive traffic. For example if this traffic is lost, then the data traffic will be unrecoverable for a long period of time. In the case of secure multicast if no redundancy coding is used (see Section 3) maintaining more than one link has a high cost. Therefore it will only be considered if the data and security traffic are separated. The multi-link attachment is a variation of the basic physical tree protocol. In this case the joining node authenticates and attaches to *n* upstream nodes.

4. K-out-of-N coding approach for link resiliency

This algorithm addresses two issues: efficient reliability and increased security. The reliability is improved by allowing a node to maintain more than one link but at a lower communication cost. The security is increased by requiring a joining node to authenticate with at least *k* members of the group. We will start by describing our scheme by an example.

Example: $k = 2$, $N = 3$. The simplest scheme can consist of each group member connecting to $N = 3$ other members of the group. Each new member only needs to receive two messages out of three.

Data traffic: the data is multicast directly to the group members or through the security multicast tree.

Security traffic: the first solution is to have the group key changed by the source for each packet. This is a reasonable assumption if the size of the packet is much smaller than the size of the key. The second solution is to change the key by the source at each group membership change. In this case the source has to be informed of membership change. The cost is still reasonable because this information is may be already sent to allow the source to keep track of who is in the group. Also, the membership change only requires a multicast packet and not $O(M)$ packets (where M is the size of the group).

Transmitted key information: the group key K can be partitioned into k (= 2) portions. Here $K = K_0K_1$, and if the size of the key is 128 bits, each portion will have 64 bits. The joining member can request K_1 from his first link, K_2 from his second link and $K_0 \oplus K_1$ from the third link. Another solution is to have each node send to its dependents $K_0 + Id * K_1$ (where the addition and multiplication are computed over a field of size greater than the key space, and Id is the unique identification number of the sending node). With both solutions the receiver can recover the key K using the information sent by two upstream links. In addition a node cannot recover the key if it is *only* connected/authenticated by one group member. The excess computation cost of the second solution can be balanced by the fact that a node always sends the same message. Furthermore, the second solution can be easily extended to larger values of k and N.

General case: each node sends $K_1 + Id * K_2 + \ldots + Id^{k-1} * K_{k-1}$. Any k correctly received messages are used to recover the key K using a Lagrange interpolation. Further, resiliency to malicious nodes that send wrong portions can be achieved by using an error-correction techniques such as the Berlekamp-Welch algorithm.

5. Service right certificates and revocation process

The poster draft provides more information on the *service right certificates* that allow nodes to prove their right to access the group information. It also describes the revocation models and process. Whenever a node is revoked it will be excluded from the group by its upstream node and will not be able to join again at any node. Furthermore, the revocation mechanism only requires group members to store a short list of recently revoked nodes.