

Intrusion Tolerant Key Management for Large Group Multicast

Shivakant Mishra
Department of Computer Science
University of Colorado, Campus Box 0430
Boulder, CO 80309-0430, USA.
Email: mishras@cs.colorado.edu

1 Introduction

A *key management server* manages a set of cryptographic keys in a secure multicast group. It stores these keys, updates them when certain events occur, and distributes them to the group members using a key distribution protocol. The process of updating and distributing cryptographic keys is called a *rekeying* operation. Rekeying is required to ensure that only the *current* group members can send encrypted multicast data, and decrypt the received multicast data. In this project, we focus on large multicast groups with frequent membership changes, i.e., groups consisting of a significantly large number of members (100,000 members or more) with members joining or leaving quite frequently. When a group is large, the cost of key management can become prohibitively expensive.

We have designed and implemented a new key management protocol called Mykil (**M**ulti-**H**ierarchy Based **k**ey **D**istribution Protocol) for managing cryptographic keys in large multicast groups that exhibit frequent membership changes [2, 3]. Important advantages of Mykil include a fast rekeying operation for large group sizes, continuous availability of the key management service in a disconnected network environment, an ability to map the group structure to the underlying network infrastructure, robustness, and support for user mobility and smaller hand-held devices.

Mykil is based on a combination of group-based hierarchy and key-based hierarchy systems for group key management [4]. It uses the idea of group-based hierarchy to divide a multicast group into several smaller subgroups called *areas* with a designated area controller (AC) for each area. There is a separate *area key* for each area. Different areas are linked with one another to form a tree structure, with ACs providing the links—an AC of an area *A* is also a member of another area *B* (area *B* is *A*'s parent in the tree-structure organization). A group member belongs to exactly one area. In addition, Mykil builds a tree-structured hierarchy of cryptographic keys in each area to facilitate key distribution to the area members. An area controller serves

as the root of the tree in its area, and each area member is associated with a different leaf of this tree. To multicast some data, a group member encrypts the data using a randomly generated key. The group members send this encrypted data along with the random key encrypted using the member's area key to its area controller. To forward multicast data to another area, an AC decrypts the random key, and reencrypts it using the other area's area key. An example of propagation of data multicast by a group member m_s is shown in Figure 1.

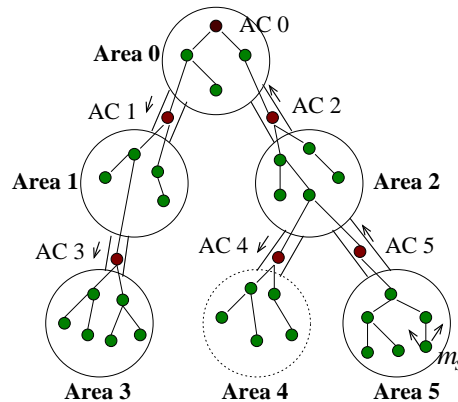


Figure 1. Organization of group members in Mykil.

At present, Mykil is not designed to tolerate intrusions. In fact, we are not aware of any key management system designed for large group multicast that provides intrusion tolerance. The main goal of our project is to investigate intrusion tolerance mechanisms for developing an *intrusion-tolerant key management system* for large group multicast. In Mykil, if an intruder manages to compromise an area controller, it can successfully perform the following four attacks: (1) It can prevent its area members from receiving any multicast data; (2) It can deny its area members an ability to send (multicast) data in the group; (3) It can prevent all areas that happen to lie downstream in the data propagation route from accessing multicast data; and (4) It can

deny membership to legitimate entities, and allow unauthorized entities to join the group. We are currently exploring two approaches to provide intrusion tolerance in Mykil: (1) Build multiple redundant paths to propagate multicast data; and (2) Build intrusion-tolerant area controllers by exploiting threshold cryptography.

2 Redundant Multipath Routing

One of the attacks a compromised area controller can launch is stop forwarding multicast data to its area members. This will not only deny the area members from receiving any multicast data, but also, all areas that lie downstream in the data propagation route will be denied access to multicast data. Our goal is to build redundant multipath routing in Mykil to allow other areas to be able to access the multicast data despite one or more area controllers being compromised.

The basic idea behind redundant multipath routing is to propagate multicast data among area controllers in such a way that each area controller receives it (at least) twice via two independent routes. Independence of routes means that the routes do not share any area controllers, except the source and the destination. With this design, each area controller is guaranteed to receive the multicast data despite compromise of an area controller in the system. Thus, the only members that are denied multicast service due to a compromise of the area controller of an area A are the members of area A .

There are at least three important issues we need to address to deploy redundant multipath routing in Mykil. First, with multicast data being propagated on (at least) two different independent routes, network bandwidth requirements will increase dramatically (two-fold, if enough care is not taken). Second, with area controllers forwarding the same multicast data many times, the load on area controllers can increase significantly. We note that area controllers can get very overloaded, and an important reason for maintaining an auxiliary-key tree structure within each area is to reduce the load on area controllers. A naive implementation of multipath routing can result in worsening this load on area controllers. Finally, intruders can still deny multicast service to other (non-compromised) areas, if they can simultaneously compromise multiple area controllers. To address the three issues, we are relaxing the current rigid, tree-structured interconnection of different areas that Mykil currently employs, and replacing it with a general interconnected network.

3 Intrusion Tolerant Area Controllers

Redundant multipath routing provides intrusion tolerance by minimizing the damage an intruder can cause to

other (non-compromised) areas after compromising the area controller of an area. We are also exploring intrusion tolerance mechanisms to build intrusion tolerant area controllers. In particular, our goal is to minimize the damage that an intruder may cause within an area after compromising that area's area controller.

An area controller performs two important, high level functions for its area members. The first one is encrypting and forwarding multicast data to its area, and the second one is authorizing the newly joining members. To implement these functions, it manages a set of cryptographic keys, including an area key, and maintains an authorization database. By compromising an area controller, an intruder can steal various cryptographic keys, as well as compromise the authorization process of newly joining members. In particular, a compromised area controller can allow unauthorized entities to join the group and prevent authorized entities from joining the group.

We are currently exploring the use of threshold cryptography to make an area controller intrusion tolerant. The basic idea is to build a distributed area controller comprising of a group of n processes, each having access to only a part (share) of the area controller information. When a new entity attempts to join the group, it must be authorized by at least k of these processes, where the value of k is based on failure assumptions. Existing group key management systems such as [5] or Enclaves [1] can not be directly used here because they are either designed for public key cryptosystem [5], or not designed to perform encryption or decryption [1]. Our goal is to investigate an intrusion tolerance mechanism that is based on symmetric-key cryptography, allows partition of cryptographic keys into n shares in such a way that least k ($k \leq n$) shares are needed to construct the key, and facilitates encryption or decryption without reconstructing the key at one location.

References

- [1] B. Dutertre and V. Crettaz. Intrusion-tolerant enclaves. In *The 2002 IEEE Symposium on Security and Privacy*, Oakland, CA, May 2002.
- [2] J. Huang and S. Mishra. A new scalable and efficient large group multicast re-keying protocol. Technical report, Department of Computer Science, University of Colorado, Boulder, CO, 2002.
- [3] J.-H. Huang and S. Mishra. Mykil: A Highly Scalable and Efficient Key Distribution Protocol for Large Group Multicast. In *ICDCS 2003 (poster paper)*, Providence, RI, May 2003.
- [4] S. Mishra. Key management in large group multicast. Technical Report CU-CS-940-02, Department of Computer Science, University of Colorado, Boulder, CO., 2002.
- [5] T. Wu, M. Malkin, and D. Boneh. Building intrusion tolerant applications. In *Proceedings of the Eighth Usenix Security Symposium*, August 1999.