A Secure Distributed Bulletin Board

E-Voting / Thesis advisor: Prof. Dr. Eric Dubuis
Expert: Prof. Dr. Andreas Steffen

More and more data are published on the Internet everyday. How can we ensure that the displayed content has not been modified? In serious contexts (e.g., e-voting) it is essential to prove the correctness of the data. Using a secure bulletin board, authorized users will be able to post messages and have the assurance that they will never be changed, moved or deleted. Also, the messages will be available to everyone. The goal of this thesis is to describe a working solution that produces correct results even in presence of actively corrupt parties.

José Beuchat
jose.beuchat@bluewin.ch

In this master thesis, we show that bulletin boards can be used in different contexts (e.g. e-voting, auctions, system logs, etc) to allow users to post and read messages that will never be removed, modified or moved. Moreover, they should always be available, possess no single point of failure and their users are able to prove it if any of those properties are not respected.

We also present a distributed solution (see Figure 2) running at n parties, of whom less than one third can be corrupt without affecting the correctness of the bulletin board. The users randomly choose a party, post their messages and receive a receipt for them. This solution, based on the master thesis of R.A. Peters, uses a secure broadcast channel described by M.K. Reiter.

As you can see in Figure 1, our solution has a relaxed layered architecture composed of the following seven layers: Network, Secure Group Membership, Echo Multicast, Reliable Multicast, Atomic Multicast, Synchronized Multicast and Application. When a message is posted at a party, it is broadcast to the n parties by going threw all of those layers. Six layers have currently been implemented and tested. The application layer remains to be done. However, we realized a simple implementation that gives us the ability run our protocols and to monitor and manage them using a JMX client. Note that this last layer can differ a lot depending on the context. A major difference in comparison to Peter's solution is that our prototype, implemented in Java, is multi-threaded. It makes us gain performance and gives more possibilities to test the system. Unfortunately, thread-safe protocols were hard to realize.

Future work will add group-threshold signatures in the Echo Multicast Protocol, making it more efficient. It will be important to use a scheme not requiring a trusted dealer. Otherwise, a single point of failure will exist. Additionally, alternative message formats (ASN.1, XML, JSON, etc.) will be used in order to make our bulletin board ready for interoperability with possible other implementations.

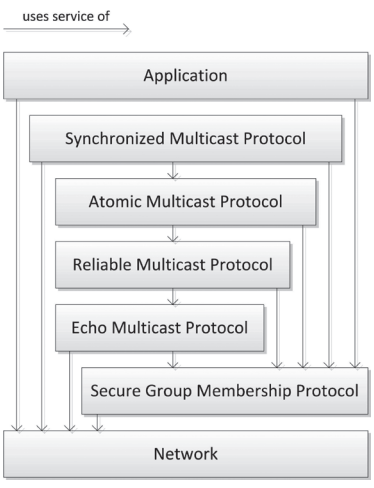

Figure 1: Relaxed Layered Architecture

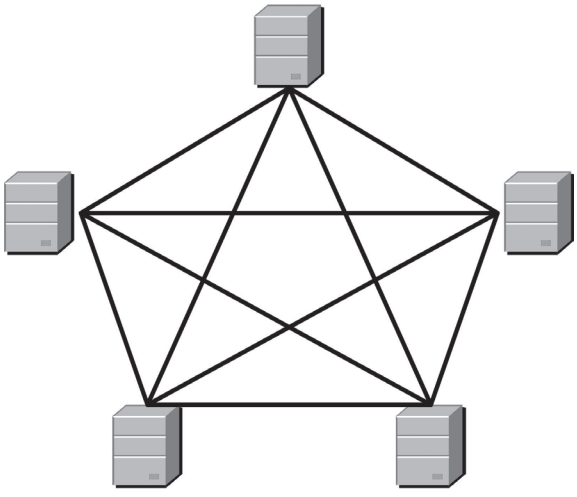

Figure 2: Distributed Solution