

Reliable and Secure Group Communication – “Scaling Network Communication Capabilities to Groups”

D. A. Agarwal, LBNL

K. Berket, LBNL

O. Chevassut, LBNL

G. Egles, LBNL

A. Essiari, LBNL

Summary

Many distributed applications require reliable ordered delivery of messages and membership services for a group of processes. This project is developing communication protocols that scale to the Internet and support multicast-based direct peer-to-peer communication within a group. The InterGroup protocol provides reliable multicast and the Secure Group Layer provides a secure shared channel similar to SSL.

Many distributed systems support communication within a group using a server or overlay network that relays messages to the members of a group. This is inefficient: sending the messages directly to the group using reliable multicast is a more efficient mechanism. Typically, reliable multicast protocols have been developed for local-area networks, and do not, in general, scale well to large numbers of nodes and wide-area networks.

The InterGroup suite of protocols is a scalable group communication system that provides reliable ordered delivery of messages to a group using multicast. It introduces an unusual approach to handling group membership, and supports a receiver-oriented selection of service to enhance scalability. The protocols are intended for a wide-area network, with a large number of nodes, that has highly variable delays and a high message loss rate, such as the Internet. The levels of the message delivery service range from unreliable unordered to reliable timestamp ordered.

The challenge with security services is how to best provide them to the application. The

approach taken by the Secure Socket Layer Protocol (SSL/TLS), the current de facto standard protocol for securing the traffic over the Internet, is to interpose a security layer between the application and the transport layer protocol. This protocol is easy to deploy since it only requires minor changes in the application to convey users' identity information/access privileges and leverages off the properties of the Transmission Control Protocol (TCP) in transmitting its own messages. The software operating in the collaborative environment, however, leverages off the multicast capabilities of the underlying network to send messages. Securing these messages requires a layer similar to SSL but for multicast.

Secure Group Layer (SGL) provides the collaborative application with a security context within which messages multicast over the wire can be cryptographically protected. The essential building block for setting up a secure multicast context is a distributed key exchange protocol that allows the participants to exchange a session key as equals and, therefore, treats them as peers. The first step in solving this problem

was to design an algorithm that allows a set of participants to agree on a session key. We refer to this kind of group genesis as the initial group Diffie-Hellman key exchange. Alone the group Diffie-Hellman key exchange is of relatively little practical use. A mechanism to enforce restrictions on who can participate in the key exchange and, therefore, the multicast group is needed. Our work has described the integration of the Diffie-Hellman key exchange and the access control mechanism into the security layer SGL.

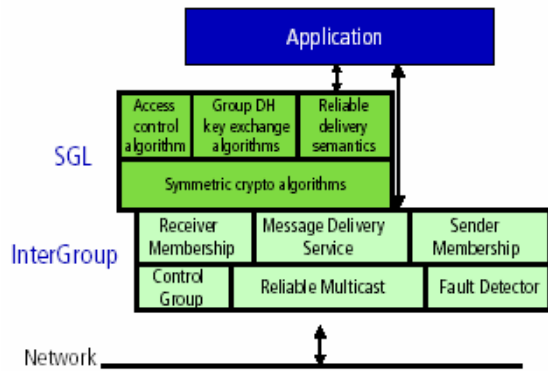


Figure 1. Illustration showing the components of the InterGroup and SGL design.

An implementation of the InterGroup protocols has been released. The implementation is divided into a node and a client module. The node module contains the core of the InterGroup protocol and is written in Java. The client module contains the application interface and is available in both Java and C++. Research, development and publication of the cryptographic protocols required for use in the SGL is largely complete. We are currently working on an implementation of the SGL.

Several collaborative applications are basing their communication on InterGroup and SGL. One example is an information-sharing and discovery system being built by

the Scalable and Secure Peer-to-Peer Information Sharing Tool Project. This system will enable scientists to store and manage information on local storage facilities while sharing them with remote participants. Another example is the Pervasive Collaborative Computing Environment Project which provides a lightweight collaboration environment. The new Access Grid 2.0 release will also include a chat tool that uses the InterGroup protocol for communication.

By using InterGroup and SGL as core communication services in the collaboration environment, existing collaborations can easily operate in either an ad hoc or infrastructure-enabled setting. This allows servers to provide added value services rather than being essential components. Thus, the dependence on centralized infrastructure is reduced and informal, spontaneous collaborations are enabled.

Besides the collaboratory tool development projects listed above, we are also collaborating closely with the Distributed Security Architectures Project at LBNL and the Complexity and Cryptographic Research Group at the Ecole Normale Supérieure in Paris, France.

For further information on this subject contact:

Dr. Deborah Agarwal, PI
 Collaboration Technologies Group
 Lawrence Berkeley National Laboratory
 URL: <http://www-itg.lbl.gov/CIF/GroupComm/>
 Phone: 510-486-7078
 DAAgarwal@lbl.gov