Sequoia – A Robust Communication Architecture for Collaborative Security Monitoring Systems

Xun Kang, Dayi Zhou, Dan Rao
Advisors: Jun Li, Virginia Lo
Department of Computer and Information Science
University of Oregon
{kangxun, dayizhou, rao, lijun, lo}@cs.uoregon.edu
Website: http://netsec.cs.uoregon.edu/research/sequoia.php

ABSTRACT

Our work involves the design, evaluation, and deployment of *Sequoia*, a robust communication architecture for distributed Internet-scale security monitoring systems. Sequoia's carefully designed communication architecture supports a rich set of communication patterns for regional and global sharing of monitor observations, collaborative decision-making among monitors, and delivery of security information to monitors. Sequoia offers continuous high-quality service using a scalable self-organizing structure that is resilient and adaptive. Driven by realistic security applications, including our current research in worm defense and open proxy blacklists, Sequoia's design achieves its goals by building on the state-of-the-art techniques in structured and resilient overlays, application-layer multicast, and peer-to-peer networks, enriched by our own expertise in communication paradigms, scalable distributed protocols, and network security.

Sequoia comprises three key protocols: Monitor Neighbor Discovery Protocol (**MND**), Distributed Dominator Selection Protocol (**DDS**), and Communication Path Discovery Protocol (**CPD**). Using these protocols, monitors self-organize into a two-level hierarchy on which scalable, fast and trustworthy message delivery can be achieved.

The MND protocol is used to form a topology-aware flat overlay among monitors, with every monitor connected to nearby nodes as its neighbors. A monitor node joins the Sequoia monitor overlay by contacting known landmark nodes to obtain its coordinates, which are then used to query a directory server for a recommended list of nearby nodes. The monitor then chooses the closest neighbors based on round-trip measurements. Each node can further optimize and maintain its neighborhood relations through local gossiping.

The goal of the DDS protocol is to form a two-level communications hierarchy from the flat neighbor overlay constructed by MND. A monitor in the higher level of this hierarchy (*dominators*) must meet minimum requirements regarding trustworthiness and routing performance. A monitor can choose to apply for a Sequoia-certificate, or *S-certificate*, from a registry service, certifying this monitor's service type, trust level, public key, and other information. Each monitor in the lower level of the hierarchy (*dominees*) eventually selects one or more higher level monitors; thus, each dominator acts as a hub for a group of dominee nodes to reach the rest of monitors.

A dominator periodically advertises itself to its x-hop neighborhood, and presents its S-certificate and other qualifications to dominees. As needed, a dominee node can search in its y-hop neighborhood for dominators, selecting those it wishes to utilize based on the dominator's attributes. A caching mechanism is used to reduce message overhead. While improving the scalability, the two-level structure ensures that untrusted nodes will not be able to forward security information for others, providing a robust communication structure.

To support a rich set of communication modes among monitors, including a publisher-subscriber paradigm, the CPD protocol is used to discover multiple delivery paths from one or more senders to one or more destinations, considering both efficiency and security constraints. This is achieved by mapping the dominator nodes into a hierarchical CAN overlay. Disjoint paths are found using the dimensional node labeling scheme associated with the hierarchical CAN, while trusted paths are found using distributed search algorithms such as max-min bottleneck. Between each sender and each receiver, additional maximally disjoint paths can be established if stronger resiliency is desirable.

The need for an architecture in order for security monitoring systems to gather, share, and deliver information in a large-scale system without centralized control has never been more compelling. Sequoia's use of self-organized topology-aware structure to support rich, fault-tolerant communication is an important step towards this goal. We expect to further advance this research by applying Sequoia to specific monitoring applications.