

Sequoia – A Robust Communication Architecture for Collaborative Security Monitoring Systems

Xun Kang, Dayi Zhou, Dan Rao (Advisors: Jun Li, Virginia Lo)
Network Research Group, 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 supports a rich set of communication patterns for regional and global sharing of monitor observations, collaborative decision-making among monitors, and timely delivery of security information to monitors. Highly secure communication is achieved through a comprehensive set of security mechanisms for trust management of participating monitors and trust-based routing. In addition, Sequoia offers high-quality and reliable communication services using a scalable self-organizing structure that is resilient and adaptive.

The design of Sequoia is driven by our current research in open proxy blacklists and worm defense. Sequoia's communication architecture supports aggregation, integration, and dissemination of blacklists using a publisher-subscriber paradigm. We are also investigating distributed worm defense using Sequoia's infrastructure for collaborative consensus on worm signatures as well as for filtering and dissemination of worm information.

Sequoia comprises three key protocols through which monitors self-organize into a two-level hierarchy on which scalable, fast and trustworthy message delivery can be achieved:

The *Monitor Neighbor Discovery Protocol (MND)* 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 *Distributed Dominator Selection Protocol (DDS)* 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 (*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 scalability, the two-level structure ensures that untrusted nodes will not be able to forward security information for others, providing a robust communication structure.

Sequoia supports a rich set of communication modes among monitors, including unicast, multicast, broadcast, anycast, and aggregation. The *Communication Path Discovery Protocol (CPD)* discovers 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 highly trusted dominator nodes into a structured overlay network. Disjoint paths are found using node labeling properties associated with the overlay, while trusted paths are found using a distributed protocol that maximizes the trust rating of a path. Between each sender and each receiver, additional maximally disjoint paths can be established if stronger resiliency is desired.

The need for an architecture 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 a self-organized topology-aware structure to support rich, fault-tolerant, and secure communication is an important step towards this goal.

REFERENCES

- [1] J. Li, P. Reiher, and G. Popek. Resilient self-organizing overlay networks for security update delivery. *IEEE JSAC*, 22(1), 2004.
- [2] A. Rosenstein, J. Li, and S. Tong. Mash: The multicasting Archie server hierarchy. In *ACM Computer Communication Review*, 1997.
- [3] A. Rowstron, A. Kermarrec, M. Castro, and P. Druschel. SCRIBE: The design of a large-scale event notification infrastructure. In *Proc. NGC 2001*.
- [4] E. Anderson and J. Li. Aggregating detectors for new worm identification. In *USENIX 2004 WIP*.
- [5] H. Kim and B. Karp. Autograph: Toward automated, distributed worm signature detection. In *USENIX Security 2004*.