Survivable, Real Time Network Services

XDefense Advanced Research Projects Agency
Contract F30602-98-1-0225, DARPA Order G409/J175

Option: Autonomous, Distributed, Hierarchical Authentication Scheme

David L. Mills
Electrical and Computer Engineering Department
University of Delaware
Newark, DE 19716

15 February 2001

1. Introduction

We consider scenarios where sensors of one kind or another are deployed over some geographic area such as a battlefield or planetary surface. The intended application is for military intelligence, weather surveillance or planetary exploration. However, the sensors might be captured by an enemy or destroyed under a tank tread or by a volcanic eruption. We assume, although this is not required, that the sensors do not know in advance the coordinates of deployment or the orientation of antennas, etc. However, the sensors have onboard computing and storage resources, as well as wireless links with sufficient power and directivity to reach at least a fraction of the other sensors in the deployment. Furthermore, we assume the deployment, individually or collectively, has the means to preprocess the sensor data and communicate to a collection center or centers.

Among the issues of special importance in sensor networks are low detection/intercept probability and battery power conservation. In general, this requires the use of spread spectrum, directional antenna and power management systems. These considerations lead to designs with large spreading gains and relatively low data rates. In any case, the sensors must provide both for data collection and transmission, either directly or indirectly via a neighbor. Considerations of power management suggest that communications between the sensors as a network and the collection center(s) be delegated to only one or a few sensors, but the delegations can change from time to time as individuals die from a rocket or exhausted battery.

We assume the network of sensors must operate autonomously and without prior configuration. Once deployed, they must find each other, determine such things as antenna orientation, power level, code rate and code/spread parameters. They must also determine the network routing, synchronization and other functions necessary for the overall network operation. The use of a broadcast/multicast technology makes these functions simpler and more robust, but is in principle not necessary beyond the initial acquisition or repair stages.

Along with the requirement for autonomous configuration is the requirement for strong security. This assumes a security model with defined protocol and certificate schemes. In conventional wired networks this can be done using symmetric key cryptography or public key cryptography, each with its own advantages and disadvantages. In wireless networks and even more so in sensor networks, these technologies seriously challenge the resource limitations. For example, symmet-

ric key cryptography complicates key distribution and does not scale well. On the other hand, public key cryptography requires significant processing and communication resources.

Our approach to a sensor network security strategy is derived from the autokey and autoconfigure technology developed for NTP and reported previously in management reports and the web. This technology provides autonomous, secure, network configuration and repair with no per-entity configuration. The autokey technology, which is based on a special protocol, provides a secure authentication trail from each entity via the network to previously secured entities. The autoconfigure technology, which is based on an add-drop heuristic, provides automatic network configuration with respect to defined metrics like transmission delay, bandwidth, etc. Both technologies can survive the occasional loss of an entity or addition of a new one.

One thing the autokey/autoconfigure technology lacks is a means to verify certificates which bind the identification values for each sensor to its public key. In the conventional model, certificates are loaded before deployment or obtained after deployment from some kind of directory service. However, the assumptions above preclude loading in advance. In addition, centralized directory services are vulnerable to attack and distributed services require consistency and access protocols that can clog the network resources.

2. Approach

In conventional security models an entity is presumed either secured or not and with no provision for some shadings between. However, a sensor network might be seriously challenged where data may be useful even if somehow its authenticity was not completely assured. For instance, the chain of command might be temporarily or permanently broken, yet various portions of the network might continue to believe the sensors that were previously secured. It might also happen that certificate trails cannot always be traced to the source because the network has fragmented.

As in PGP, this suggests two metrics, one ranking the level of trust for sensor reports and another ranking the level of trust for reports from that sensor about other sensors. In generalizing these notions, there might be a need for other related metrics or security indices as well. Just for the purposes of this proposal, we will call this approach the autotrust model. To support it, we need a strawman architecture, protocol and algorithms. A preliminary requirements list might include:

- The architecture must provide a security assessment in real time and without explicit messages to network services other than nearest autoconfigure neighbors.
- The protocol must support a dynamically changing network topology as autoconfigure adds new neighbors and drops old ones.
- The algorithms must not require the use of a centralized database. They must function when a database has fragmented in disconnected segments.
- The intrinsic network primitive in the trust model is signing certificates, which carries an increment in trust depending on the number of signatures and the trust of the signatories.

In a preliminary conceptualization of the autotrust scheme, consider the autoconfigure scheme adapted to autotrust. This scheme uses an add-drop heuristic with a single metric based on synchronization distance, which is generally equivalent to roundtrip delay. The scheme uses an expanding ring search to discover candidate entities, evaluate the metric and either drop them or

add them to the current server population, but constrained by a maximum number of servers. In simple, the autoconfigure scheme operates as the common decentralized form of the Bellman-Ford routing algorithm, but with an add-drop neighbor router configuration scheme.

In our concept trust is based on certificates and what has been called a signing party. The sensors with direct links to collection entities might ask a trusted agent to sign their certificates, which are then stored at the sensor and retrieved by a suitably augmented autokey protocol. Autoconfigure neighbors then exchange and sign certificates all the while augmenting their own trust metrics by that of the signatories. A metric is assigned as a function of the protocol and eventually this metric percolates throughout the network. Should the network be degraded or partitioned, the trust metric is recalibrated as the segments reconfigure.

We have suggested two metrics above to represent the level of trust assigned each sensor, but there could be others as well. It is not a large hop of faith to extend the add-drop heuristic to include a multi-metric capability. Central to the success of this scheme is the algorithm that constructs the actual routing/discovery metric from an ensemble of submetrics including delay and possibly several trust indices. Since data transmission and trust transmission might form different routing trees, the add-drop population might include some neighbors for transmitting data and others for trusting sources.

3. Research Plan

The point of departure for this work is the final working version of the autokey and autoconfigure implementation for the Network Time Protocol (NTP). It is understood that the time synchronization function of this implementation is only ancillary and the software is used only as an implementation and evaluation vehicle for the proposed work.

The period of performance is from the contract date through the end of September, 2002. The staffing level includes the principal investigator, one graduate student and one undergraduate summer intern. We will need to replace two aging Sun workstations and expect to replace one or more cesium tubes for our three aging cesium oscillators.We will also need to install software and hardware upgrades for our Windows PCs. Finally, we will need to augment our laboratory router configuration to support needed features for virtual networking.

The deliverables will include a final report, web documentation and software suitable for testing and further development in a sensor network environment.

4. Statement of Work

1. Evaluate the physical and environmental environments for typical sensor networks, in particular limitations on power, spectrum and data rate.
2. Evaluate the current autokey and autoconfigure protocols with respect to the limitations in (1). Construct strawman scenarios showing the feasibility of possible adaptations.
3. Evaluate possible approaches for a fully hierarchical, distributed security management scheme suitable for sensor networks.
4. Select a likely candidate resulting from (2) and design a protocol suitable for testing in a local environment.

5. Implement the protocol to operate in the NTP test vehicle.
6. Test and evaluate the implementation first in a local network and later in the CAIRN testbed. Write a final report summarizing the lessons learned and test results. Evaluate the performance limitations with respect to the sensor network environment.