Network Integrity by Inference in Distributed Systems

John Mark Agosta and **Simon Crosby**
Intel Research #SC12-303, Intel Corporation, 2200 Mission College Blvd., Santa Clara, CA 95054
{john.m.agosta, simon.a.crosby}@intel.com

1. Introduction

The complexity of distributed computational systems leaves enterprises vulnerable to security threats, mis-configuration errors, and system component incompatibilities and failures. Operators face the increasingly difficult task of finding and responding to failures, but the observed symptoms may be far removed from the root cause: For instance, poor response time from a server in New York experienced by a trader in Dallas could result from failure of the server, a mis-configured router in London, or an inability to reach the DNS server. In summary our infrastructure is extremely brittle, and becoming more so.

Although it was designed to be robust to many network failures, the IP layer hides most useful information related to network conditions. End systems are left in the dark when something goes wrong, and even within the network routing decisions are based only upon reachability criteria, and ignore potentially useful information about resource availability, link error rates, vulnerabilities and organizational policy.

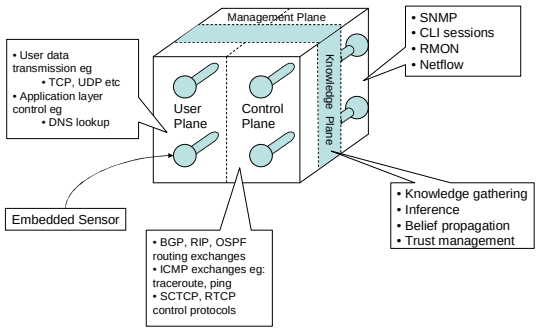
We at Intel are studying how to complement the existing IP protocol stack with a distributed layer of capabilities for information gathering and inference, with the goal of enhancing the control and management of network resources and to facilitate the delivery of meaningful, semantically rich diagnostic information to operators, users and applications. Our work is a natural extension of the Active Networking literature [6] and Clark et al.'s recently proposed "Knowledge Plane"[2] , a layer of inference-capable software that augments the user, control and management planes in the IP protocol stack.

2. Architecture

The diagnostic problem is a task that requires distributed data acquisition and processing. Observation and measurement of local traffic by capturing (*inter alia*) detailed packet-level statistics results in such vast quantities of information that it must be processed locally. Network diagnostics form an "in-band" data stream. They depend on the same transport mechanism for diagnostic information as the layers that they diagnose. Without a reliable out-of-band communication layer to carry diagnostic information, it is important to use a federated design and to take advantage of the reliability and robustness that it offers. Finally, without an ability to observe concurrent yet geographically distributed events across the entire distributed system it may be impossible to provide useful diagnostics.

The inference task of diagnosing reliability induced failures and intrusion breaches cannot be uncoupled: The same kinds of events may arise from either cause. It is only after inference that the causes can be distinguished. Systems that go no further than to classify rare events as anomalies are not capable of inference, and therefore frequently generate false alarms.

The Role of the Knowledge Plane



2.1. Detection and Sensing

Our architecture involves the addition of data gathering and inference capabilities at some subset of the nodes of the distributed system. At each one of the nodes we add an additional plane of functionality that permits detectors to be embedded within each of the data, control and management planes. Detector functions could, for example, include packet filtering, snooping routing table updates, or observing a CLI session on the management port of a router.

2.2. Internet Routing as Belief Distribution

It is perhaps easiest to view our work as a generalization of one of the processes already embedded in all routers, namely routing table computation and distribution—the problem of choosing forwarding paths for incoming traffic based upon a belief about the connectivity of the local node to other elements in the network. [3] Of course at any moment the routing tables at a particular node may be out of date due to remote topology changes, but at all times the routing table reflects the node's belief about the best forwarding path choices, given the

information that the node has collected from its peers to date.

In our system we view the problem of integrity management as a problem of belief inference and propagation between nodes in a distributed system. [5] Practically, we decompose the task as follows: Each node in the distributed system (both routers and end-systems) is augmented with a set of sensors that assemble relevant local information from each of the existing planes of communication, namely the control, user and management planes. The sensors assemble measurements that are input into a locally maintained Bayes net model that enables the node to diagnose its own state—for example to detect local intrusion attempts, repeated failures of a particular egress link, or changes in topology. This diagnosis is expressed as a set of beliefs about the node. Each node also models the state of its adjacent neighbors that is informed by belief propagation between neighbors.

2.3. Diagnosis of remote faults via distributed belief update

When a failure occurs that widely affects remote services, the responsibility of the diagnostic layer is to convey evidence of the failure from the sensing location back to the location of its cause and to the person responsible for rectifying the fault. It is an interesting question how this can be achieved in a distributed diagnostic system where the sensing location only knows about its immediate environment.

If each node has a summary model of its environment in which is embedded a model of its local state, then it can, by process of elimination, infer whether a local impairment has a local or remote cause. If the cause is remote, the impaired node can propagate belief about the impairment toward its likely source (and to other elements of the network). The accumulation of belief from multiple sources back to the source of the impairment or to the responsible human is the power of distributed diagnosis. How to combine belief without having a detailed model that includes all sources of information makes distributed diagnosis a hard problem.

A node's summary model of its neighbor consists of variables that mirror the services a node receives from its neighbor. A node has beliefs, incorporated in these variables, that each of its neighbors is available ("live"), is connected to others, and does or does not route packets to other networks and services, for example. Each variable in the node's summary model can be updated by exchange of belief messages with the corresponding neighbor's corresponding variable. The full range of variables that should be included in the summary model is the subject of ongoing research, but it is likely to include full FCAPS functionality in a rich semantic context.

To develop the analogy with operational aspects of routing protocols, nodes must also propagate beliefs to their neighbors about remote states for which the source is not identified. Such as p{"My DNS service has failed"}. A node's local model may assume that it has independent evidence of the same fault ("I can also not resolve DNS queries"), and propagate combined belief, or it may use the evidence to come up with belief about a related hypothesis. ("That DNS server relies on a connection that is intermittent, hence the quality of the connection is implicated.") We intend to pursue the design of the contents of a node's local modes, its reflection in the summary models of its neighbors and whether the design requires a static or dynamic Bayes net structure.

2.4. Management, security and trust

It is more efficient for a node to communicate beliefs rather than to communicate raw data to maintain local consistency between the local model and neighbors' models. With some luck, local belief propagation will result in global consistency of belief. [1] However if parts of the data network are operating maliciously and engaging in deception there may be no globally consistent belief state. One way to model this is to maintain beliefs about beliefs; so that the locally held beliefs about other nodes that in turn are believed to have been compromised, are discounted.

We call this additional layer of beliefs the *trust layer.* Nodes maintain belief about the trustworthiness of neighbors. A message that is received from a less than trustworthy neighbor is conditioned by the neighbor's trustworthiness. Trustworthiness of a source is inferred analogously to the chaining of trust relationships among peers in PGP. Trust update in the network of trust relationships among nodes occurs by local belief propagation, in parallel to belief propagation about network operating state, as just described. We note also that trustworthiness is a much needed attribute in IP routing, where routing table updates from a peer network should not be trusted unless the peer is known to be trustworthy[4].

3. References

[1] Christopher Crick and Avi Pfeffer. "Loopy Belief Propagation as a Basis for Communication in Sensor Networks". Proceedings of the 19th Conference on Uncertainty in AI, 2003.

[2] David Clark et. Al. "A Knowledge Plane for the Internet", Proc. Applications, technologies, architectures, and protocols for computer communication, ACM SIGCOMM 2003, Karlsruhe.

[3] B. Halabi, Internet Routing Architectures. Cisco Press, 1998.

[4] Wei Li,"Inter-Domain Routing: Problems and Solutions", http://citeseer.nj.nec.com /li03interdomain.html, 2003.

[5] A. Yuile, "CCCP algorithms to minimize Bethe and Kikuchi free energies: Convergent alternatives to belief propagation." Neural Computation, 14:1691-1722, 2002.

[6] D. Tennenhouse and D. Wetherall, "Towards an Active Network Architecture," CCR, April 1996.