A Research Agenda for Distributed Policy-based Management

Mads Dam[1,2], Gunnar Karlsson[1], Babak Sadighi Firozabadi[2], and Rolf Stadler[1]

[1] KTH IMIT, Isafjordsgatan 39, SE-16440 Kista, Sweden, E-mail: {mads, karlsson, stadler}@imit.kth.se
[2] SICS, Box 1263, SE-16429 Kista, Sweden, E-mail: {mfd, babak}@sics.se

Abstract

Policy-based management is based on defining a set of global rules, according to which a network or distributed system must operate. In the last few years, policy-based management has begun to emerge as the dominant paradigm for developing network and systems management functions, primarily, since it can reduce complexity in management applications. Although attempts are underway to standardize policy-based management, significant research challenges remain. At KTH and SICS, a joint activity has been started to focus on some of the key issues. The paper outlines the research agenda for this activity.

1 Introduction

The concept of using policies in management applications has been applied selectively for a quite some time. Only recently, however, attempts are being made at applying this concept to virtually all management functions, as well as to the development of a generic architectural framework, protocols, data models, etc., for this purpose. A recent activity of the IETF WG on policy-based management [1] has created a strong interest in the subject domain, and has inspired numerous projects in research labs.

A group at Imperial College started systematic research in this field in the early 90s [2]. While this work focuses on distributed systems management for organizations, the primary goal of the IETF initiative is to support new Internet services, such as Diff Serv, Int Serv, MPLS, IP telephony and VPNs, starting at the device level. Both approaches are to a large extent complementary.

Although standardization efforts are underway, key aspects in the area of policy-based management still neded to be been addressed. For instance, the impact of QoS policies on the network state is barely understood. Also, realizing effective mappings of high-level policies into lower-level policies remains unsolved in many cases. Finally, the IETF framework does not provide answers to the question of how to best distribute the components of a policy-based management system in medium and large-size networks.

This short paper describes our research agenda in the field of policy-based management. Supported by VINNOVA, we have started three projects that focus on different aspects of distributed policy-based management. These projects are discussed in the following sections.

2 Policy-based management for large-scale dynamic networks

This project aims at developing engineering solutions for policy-based management in large-scale dynamic networks. We focus on networks with at least 1000s of nodes, which have a dynamic state and a frequently changing topology, and which do not have a centralized database with up-to-date topology and state information. Today, such networks exist in the form of Autonomous Systems, i.e. Internet domains under independent administration, or cable networks that distribute digital TV. In the future, when the technology for ad-hoc networks will have matured and networks that are part of pervasive computing environments will become a reality, large-scale dynamic networks are expected to become even more widespread.

Such networking environments pose specific challenges for the introduction of policy-based management. We are addressing some of the key issues in this area, such as:

- Policies with end-to-end significance are formulated by network operators and are entered into the management system at one or more access points. To take effect, they need to be translated into lower-level policies and disseminated to network elements. Therefore, an architecture needs to be engineered that allows for *translating and disseminating policies*. This architecture is part of the management system and, for large networks, must be distributed to allow for efficient operation.
- In reaction to changes in the network state and network topology, lower-level policies need to be re-computed for the network to remain operational and to efficiently deliver services with guaranteed quality.

In large-scale networks, it is infeasible to perform this type of processing in a centralized way. For this reason, policies must be re-computed locally, within smaller network areas, on a time scale that depends on the frequency of changes. The question that needs to be addressed is: Which architecture and which mechanisms support best the *local, autonomous re-computation of policies*?

- In large networks, we must assume that, at any given time, some of the network elements or other network components are out of service or malfunctioning. A policy-based management system needs to be *robust* enough to keep network services running during fault conditions and to *initialize network components* with appropriate policies as they become operational.

We are specifically interested in QoS policies and performance aspects. Global policies in this context define end-to-end objectives for packets or flows that traverse a network. They can be formulated on the packet level, such as defining a limit on the end-to-end delay of a specific class of packets, or on the level of flows, such as specifying an upper bound for blocking rates for certain classes of flows.

The task of the management system is to break down the above global policies into lower level policies that apply to network regions, which, in turn must be translated into policies on the device level. As the example of the upper bound on an end-to-end delay shows, there are generally many solutions to implement such a global policy. (An end-to-end delay of, say, 50ms, can be divided in many ways into smaller delay bounds that can be enforced on the device level.) At the same time, every solution that is implemented will change the network state and, therefore, the way the network is controlled. Generally speaking, it is a hard problem to devise a system that *concurrently* supports several QoS policies in an efficient way.

Additional problems we plan to investigate in this work include the dissemination of QoS policies and the re-computation of local policies, triggered by topology changes or network faults, in the context of large networks.

We intend to address these issues from the perspective of network architectures, protocols, algorithms and software architectures. As far as aspects of network architectures are concerned, we plan to work within the IETF framework on policy management. Since that framework does not address the issues of large-scale systems, our work will likely be complementary to the IETF activities.

In terms of protocols, algorithms and software architectures, we plan to base the research on the application of pattern-based management, an approach for managing large networks, which has been developed in our group [3]. Our plan is to simulate and evaluate the developed architecture and the algorithms on high-end PCs. Such a platform will allow us to evaluate different design decisions against each other and to gain experimental results for large network configurations.

Further, in order to understand and solve some of the problems of applying our solution to today's networks, we plan to implement a policy-based QoS management on lab, testbed which contains 12 Cisco routers.

2 Decentralized Contract Management

We will address the issue of managing access to disparate resources that are not under control of a single network administrator/provider.

In a multi-domain network, a number of individuals and/or service providers interact in a collaborative environment to provide certain services organized according to a set of rules and policies that define how their resources can be shared among them.

A multi-domain network is usually a composition of heterogeneous and independently designed domains with no centrally controlled enforcement of the policies. Consequently, there is no guarantee that policies will be followed as they are prescribed: members of a network may fail to, or choose not to, comply with the rules. If there is no way of practical (physical) enforcement of policies, then it would be useful to have a *normative control mechanism* for their *soft enforcement* [4]. By soft enforcement we mean that even if service providers are practically able to avoid complying with the organization policies, they can still be subject to sanctioning and remedial action as the consequence of their behavior.

Consider following scenario: Alice wants to make a network reservation from her computer in domain A (the source domain) to Charlie's computer in domain C (the destination domain). Alice has a service level agreement (SLA) with its service provider (domain A) to get some level of service defined in terms of quality of service (QoS) parameters. The path between Alice and Charlie includes some other intermediary domains. An end-to-end reservation for Alice requires a chain of SLAs between her domain and Charlies domain. Beside these SLAs, each domain may have its own local policy that does not necessarily comply with the SLAs that the domain has signed for.

The issues are, how can Alice's request for bandwidth reservation be supported by a set of credentials that shows her right for that request? and what happens if she does not get the required QoS as she is *entitled* to?

The approach we are suggesting is that Alice's request for bandwidth reservation is submitted with evidence for her right for that in terms of digital credentials. Now it is up to each domain to decide whether they are, according to their SLAs, obliged to provide the service she requires. If Alice does not receive the service she is entitled to, then it must be possible to find the point of failure, and, perhaps, some sanctions should be imposed on those accountable for that failure.

3 Secure Policy Management

The flexibility provided by policy-based network management comes with some serious hazards, as illustrated by the recent SNMP vulnerability uncovered by Oulu university's secure programming group.

The vulnerabilities include:

- Denial of service attacks. By messing up routing policies, networks can be rendered inoperable.
- Hijacking and misuse of network equipment.
- Unauthorized access to corporate resources, such as intranets.

These types of threats are less concerned with security at the level of network elements than with the secure management of the policy base itself. A comprehensive solution, which can scale to large, decentralized networks must address at least the following issues:

- A secure and decentralized management framework for access, authorization, and auditing. This includes a rich structure for groups, roles, and delegation, to provide efficient organizational support for the management task.
- An intrusion detection component, which will monitor network operation to determine, whether and to which extent hosts and network elements have been compromised.
- A framework for secure and efficient policy distribution, including secure methods for managing the distribution mechanism itself.

We base our work on the Constrained Delegation framework developed in the context of the Amanda project at SICS [5, 7]. Our starting point is the Simple Public Key Infrastructure (SPKI) framework for authorization in distributed systems [6]. We are extending this to constrained delegation and specific network policy formats. The goal is that the framework should be flexible and efficient enough to support small footprint implementations, for instance in the context of smart cards.

Secure distribution of policies is a multifaceted problem. One aspect is privacy. Since policies describe the way a network behaves, its structure and management, and possibly even its response to attacks, quite a lot of information can be extracted from policy traffic by a potential adversary. In the context of static networks of policy servers, privacy can be supported by standard end-to-end solutions, but for large, autonomous networks, more dynamic solutions are called for. A second set of issues concerns the management of the policy engine itself. Such an engine must be able to handle a wide array of concerns, including traffic-aware mechanisms for policy updates, constraint resolution and pre-computation, and policy-control of the updating traffic itself. Finally, the need for policy protection must be weighed against the need for intrusion detection and compliance monitoring. Knowing that it is difficult at best to protect a network against attacks of various kinds, how can we ensure that a compromised network element (for example a policy repository) does not compromise the entire network? What is involved in doing this efficiently and in a scalable way?

A serious stumbling block towards the adoption of policy-based control on a larger scale is policy enforcement and integration. Policies should, as far as possible, be independent of the platforms on which they are enforced. However, to be realized, the policies must be translated into forms understandable by the individual network elements. Thus, to be able to leverage all the functionality provided by the network elements, quite a lot of generality is required of the policy framework. For instance, for access policies, the framework must know how to render a given "object-level access policy" into any of the target systems supported by the management system. Hard-coding the representation into the policy management framework can do this. A better solution, however, would be to find a way of specifying the target access mechanism as a policy object itself, albeit on the meta-level. New target architectures can then more easily be accommodated, by simply accounting for its access control regime at the level of meta data, rather than having to implement a complete new translation algorithm.

4 Acknowledgements

This research is supported by the Swedish Agency for Innovation Systems (VINNOVA) under the project name "Policy-based Network Management."
Thom Birkeland contributed to this paper.

5 Bibliography

[1] http://www.ietf.org/html.charters/policy-charter.html
[2] J. Moffett, M. Sloman, "Policy Hierarchies for Distributed Systems Management," IEEE Journal on Selected Areas in Communications, Vol. 11 No. 9, Dec. 1993, pp. 1404-1414.
[3] R. Stadler: "Decentralized and adaptable management based on active networking technology," The First International Workshop on Active Network Technologies and Applications (ANTA 2002), Tokyo, Japan, March 25-26, 2002.
[4] Babak Sadighi Firozabadi and Marek Sergot, "Contractual Access Control - Position Paper", to appear in the proceeding of Security Protocol, 10th international workshop, Cambridge, UK, April 2002.
[5] O. Bandmann, M. Dam and B. Sadighi Firozabadi, "Constrained Delegation" Proc. 23rd annual IEEE Symposium on Security and Privacy, Oakland, 2002.
[6] C.M. Ellison, B. Frantz, B. Lampson, R. Rivest, B.M. Thomas and T. Ylonen, "SPKI Certificate Theory" RFC 2693, 1999, expired, URL: ftp://ftp.isi.edu/in-notes/rfc2693.txt.
[7] O. Bandmann and M. Dam, "A Note on SPKI's Authorisation Syntax" Proc. 1st annual PKI Research Workshop, NIST, Gaithersburg, 2002.