Internet Draft
Expires December 12, 1996
File: draft-ietf-rsvp-policy-arch-00.ps

Shai Herzog
USC/ISI

June 1996

Accounting and Access Control Policies for Resource Reservation Protocols

June 12, 1996

Status of Memo

Abstract

This memo provides insight into some possible generic approaches for policy enforcement in resource reservation protocols. We present sample scenarios for each of these approaches as a way to demonstrate their feasibility, and to motivate the development of supporting architectures.

1 Introduction

Reservation protocols, by definition, discriminate between users, by providing some users with better service at the expense of others. Therefore, it is reasonable to expect that these protocols be accompanied by mechanisms for controlling and enforcing access and usage policies. In this document, we refer to such policies as "access control". The term "access control" is quite broad; it ranges from simple access approval to sophisticated accounting and debiting mechanisms For scaling reasons, we concentrate on policies that follow the bilateral agreements model. The bilateral model assumes that network clouds (providers) contract with their closest point of contact (neighbor) to establish ground rules and arrangements for access control and accounting. These contracts are mostly local and do not rely on global agreements. The bilateral model has similar scaling properties to multicast and is easier to maintain in distributed environments.

Throughout this document we would use the terms "reservation protocols" and RSVP interchangeably. However, the contents of this document should be interpreted as applying to similar resource reservation protocols as well. The current admission process in RSVP uses resource (capacity) based admission control; we expand this model to include policy based admission control as well, in one atomic operation. Policy admission control is enforced locally at border/policy nodes (also see [HER96a]). Policy nodes are responsible for receiving, processing, and forwarding POLICY_DATA objects. Subject to the applicable bilateral agreements, and local policies, policy nodes may also rewrite and modify the POLICY_DATA objects as the pass through policy nodes.

In this document, we outline a few sample scenarios for access control and accounting; we provide these scenarios as motivation and as needed context for the development of policy control architectures for resource reservation protocols. These scenarios are based on two simple assumptions: (1) RSVP provides the needed transport service of carrying access control state (POLICY_DATA objects), hop-by-hop. (2) Access control policies are enforced locally, and can be based, among other factors, on bilateral agreements between neighboring providers, local policies and the contents of POLICY_DATA objects.

2 Simple access control

To provide simple access control, the local node attempts to match incoming policy objects with one or more of the pre-configured policies or bilateral agreements, in order to accept or reject the reservation.

Consider the following network scenario: one receiver from ISI and two from MIT listen to a PARC seminar. For simplicity of the scenario, let us limit ourselves to a **receiver based** access control scenario.

The bilateral agreements between each two neighboring providers (e.g., R1, R2 with ISI,

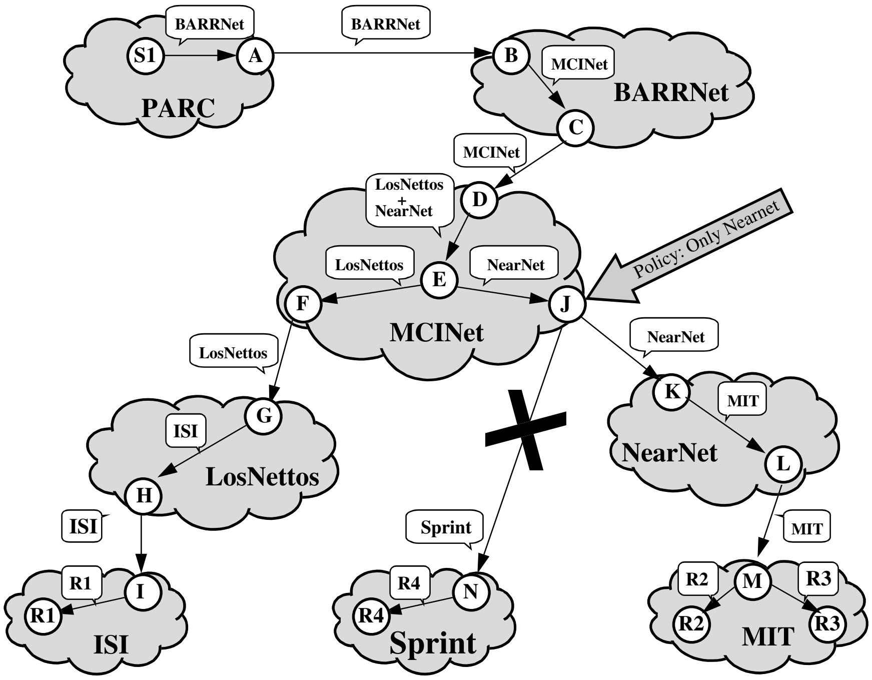

Figure 1: Simple access control

ISI with LosNettos,... BARRNet with PARC) are simple: the first provider obtains a permission to make reservations over the second provider's network. The notation $PD(cr, uid)$ represents a policy data object of type "cr" (credential) verifying that the flow belongs to *uid*. Credentials can be hierarchical, and may be rewritten on a hop by hop basis through a locally configured conversion table.

Figure 1 illustrates a reservation scenario. An typical example of a bilateral agreement could be between MCI and LosNettos: MCI would allow the LosNettos users to use its backbone. A policy data object $PD(cr, LosNettos)$ would be interpreted by MCI as a green light to accept the reservation. In this scenario, reservations from R1, R2, R3 carry policy data objects that propagate hop-by-hop (encapsulated in reservation messages) toward S1. Assuming all nodes are configured consistently, policy objects are rewritten in nodes B,D,G,I,K,M, which are entry points to clouds).

The MCI cloud is interesting. E is not a border/policy node, but still, it receives the following policy data objects: F$\rightarrow$E: $PD(cr, LosNettos)$ and J$\rightarrow$E: $PD(cr, NearNet)$. Assuming E has no authority to merge or rewrite these credentials, it must concatenate the two objects and send $PD(cr, LosNettos) + PD(cr, NearNet)$ to D. Let us further assume that D is configured with the following conversion table:

$$PD(cr, LosNettos) \Longrightarrow PD(cr, MCI)$$
$$PD(cr, NearNet) \Longrightarrow PD(cr, MCI)$$

Node D first checks if LosNettos and NearNet are authorized to reserve on their corresponding links and responds accordingly. Assuming authorization is cleared, it merges and rewrites these policy objects as $PD(cr, MCI)$ and forwards the reservation to C.

To complicate the example, assume the conversion table was:

$PD(cr, LosNettos) \Longrightarrow PD(cr, MCI1)$
$PD(cr, NearNet) \quad \Longrightarrow PD(cr, MCI2)$

Then node D would forward $PD(cr, MCI1) + PD(cr, MCI2)$ to C instead.

Local policies can also reject reservations:

In figure 1 we see that a reservation made by R4 is rejected because it arrives with insufficient credentials: the local policy in node J accepts only traffic marked as $PD(cr, NearNet)$, and R4's reservation arrives with $PD(cr, Sprint)$.

3 Advanced reservation and preemption control

Advanced reservation can be built on top of simple access control: consider the case where every advanced reservation consists of a set of bilateral agreements between different service providers, reserving network capacity at some future period of time. When advanced reservations are not public (i.e., only authorized users can use them), three classes of reservations exist: (1) walk-ins (where the conference itself does not have advanced reservations, (2) advanced reservation with unauthorized users, and (3) advanced reservation with authorized users. These numbers (1..3) can define a "preemption priority" (i.e., walk-ins are preempted first, unauthorized pre-reserved second, and authorized pre-reserved are never preempted).

The advanced reservation scenario is almost identical to the simple access control: let us assume that each bilateral pre-registration is identified by a PRID (Pre-Registration confirmation ID). Policy data objects of type AR (Advanced Reservation) would take the following form: $PD(ar, prid, uid)$. When an AR object arrives, the local node verifies the existence of pre-reservation $prid$, and checks that uid is permitted to use it. Finally, the flow is classified to one of the above three preemptive priorities and RSVP is notifies.

4 Quota enforcement/accounting/debiting

The next step is to allow for more sophisticated access control that is based on usage feedback. Here we add two additional mechanisms which (1) determine **how much** should be debited for a reservation and (2) what **debiting mechanism** should be used (if any). The following scenarios assume a pre-existing set of local accounts. These accounts are established by bilateral agreements that pre-purchase network capacity and set applicable debiting rules. The role of accounting mechanism is to verify the availability of funds/quotas in these accounts for maintaining the reservation. We consider several accounting schemes and briefly describe three: simple debiting, limited debiting, Edge Pricing, and MultiCost (MCost).

1. **Simple debiting**

   Consider the following example: lets assume that LosNettos and Nearnet each have a debit account (pre-purchased capacity) with MCI for their traffic. When node E receives the following $PD(cr, LosNettos)$ and $PD(cr, NearNet)$ for flow f, it must decide the following: (1) How much should be debited for flow f, and (2) how would that debit be shared between the account of LosNettos and NearNet. These are local configuration issues left for service providers. In this scenario, the local node would attempt to perform the debiting, and would notify RSVP on success or failure. The other aspects of the scenario (Merging policy data objects and forwarding them) is identical to that of simple access control.

2. **Limited debiting (willingness to pay)**

   Although we do not have a full understanding of the dynamics of willingness-to-pay and its properties, we can outline the basic scenario, as an extension of the simple debiting model. Willingness to pay is manifested as a limit on the policy object that authorizes the debit. For instance, $PD(crwp, ISI, 10\%\ of\ unicast)$ would represent a policy data object of type $crwp$ (Credential, Willingness to Pay), that authorizes debiting the ISI account up to 10% of the unicast cost. Here, the basic idea is that market forces would be the driving force behind what users specify as their willingness to pay.

3. **Edge Pricing**

   Edge Pricing was presented in [SHE95]. This paradigm is based on the assumption that network costs can be estimated and approximated at the edge of the network, based on purely local information. Edge Pricing is an extension of simple debiting: Edge Pricing can determine **how much** is to be debited, and the set of credentials associated with the reservation determines who **(which account)** should be debited.

4. **MultiCost (MCost)**

   MCost is an accounting scheme (and mechanism) that was introduced in [HER95]. MCost has a unique feature: it takes into account the benefits of sharing a multicast tree and distributes these savings among the members of the multicast group, according to configurable policies, basic fairness, and equality.

   MCost computes the cost allocated to each user, and that cost can be the basis for debiting. MCost can be combined with simple debiting in a similar manner to Edge Pricing.

5 Acknowledgment

This document incorporates inputs from Deborah Estrin, Scott Shenker and Bob Braden and feedback from RSVP collaborators.

References

[HER95] S. Herzog, S. Shenker and D. Estrin, Sharing the Cost of Multicast Trees: An Axiomatic Analysis, *Proceedings of ACM/SIGCOMM '95*, Cambridge, MA, Aug. 1995

[HER96a] Local Policy Modules (LPM): Policy Enforcement for Resource Reservation Protocols. *Internet-Draft*, draft-ietf-rsvp-policy-lpm-00.[ps,txt].

[SHE95] S. Shenker, D. Clark, D. Estrin, and S. Herzog Pricing in Computer Networks: Reshaping the Research Agenda, *Telecommunications Policy*, Vol. 20, No. 1, 1996 also published in *Proceedings of the Twenty-Third Annual Telecommunications Policy Research Conference*, 1995.