A Market-based Bandwidth Charging Framework*

David Michael Turner
Dept. of Computer Science
Drexel University
dmt36@drexel.edu

Vassilis Prevelakis
Dept. of Computer Science
Drexel University
vp@cs.drexel.edu

Angelos D. Keromytis
Dept. of Computer Science
Columbia University
angelos@cs.columbia.edu

Abstract

The increasing demand for high-bandwidth applications such as video-on-demand and grid computing is reviving interest in bandwidth reservation schemes. Earlier attempts did not catch on for a number of reasons, notably lack of interest on the part of the bandwidth providers. This, in turn, was partially caused by the lack of an efficient way of charging for bandwidth. Thus, the viability of bandwidth reservation depends on the existence of an efficient market where bandwidth-related transactions can take place. For this market to be effective, it must be efficient for both the provider (seller) and the user (buyer) of the bandwidth. This implies that: (a) *the buyer must have a wide choice of providers that operate in a competitive environment,* (b) *the seller must be assured that a QoS transaction will be paid by the customer, and* (c) *the QoS transaction establishment must have low overheads so that it may be used by individual customers without a significant burden to the provider.*

In order to satisfy these requirements, we propose a framework that allows customers to purchase bandwidth using an open market where providers advertise links and capacities and customers bid for these services. The model is close to that of a commodities market that offers both advance bookings (futures) and a spot market. We explore the mechanisms that can support such a model.

1. Introduction

Years of research on Quality of Service (QoS) architectures for the Internet have resulted in sophisticated proposals that have not been broadly exploited commercially. In particular, Integrated Services (IntServ) [11] and Differentiated Services (DiffServ) [8] have long been supported by major router and operating system vendors, yet have only seen minimal use in practice. One explanation offered by the

*A preliminary version of this paper was published in IEEE ISCC 2005 [38].

networking and QoS community has been a lack of a commercialization model, together with the necessary accounting and charging architecture [14]. A related crucial issue is assurance of end-to-end QoS coherence in the face of multiple intervening parties, such as transit ISPs.

These two issues, taken together, are responsible for suppressing interest from both the ISPs (in commercially exploiting QoS to its full potential) and the users (in taking advantage of such services). Simply put, if an ISP cannot be paid for reserving bandwidth to a user, they will not offer QoS; if users cannot be assured of end-to-end QoS, they will not pay for the service. Compounding the problem is the issue of management: it is certainly possible for a large entity, such as a multi-national company, to coordinate with the relevant ISPs so that its various geographically dispersed networks are correctly provisioned using a series of DiffServ or IntServ tunnels. However, the effort is considerable and requires manual intervention from a number of parties. Perhaps most importantly, the ISPs' network operations centers (NOCs) will need to configure the various routers appropriately. Clearly, such an approach will not scale well if preferentially treated bandwidth is to become a commodity that can be traded, as has been recognized before [12]. Yet, the increasing use of the Internet for time-sensitive or otherwise critical applications effectively mandate some form of bandwidth reservation, often for short periods of time (*e.g.*, watching a movie).

We present a market-based approach to self-managing QoS across multiple ISPs. Our architecture introduces a Bandwidth Exchange (BAND-X), which facilitates the trading of reserved bandwidth between ISPs and users. This facility allows purchasing bandwidth in advance (effectively creating a "futures" market for bandwidth) as well as on the "spot" market. Users can select from a range of offerings by various ISPs to create an end-to-end pipe (with the desired bandwidth and QoS) piece-meal, or can choose to purchase a complete package from a single provider (or consortium of providers), where available. This is similar to the way people purchase low-cost airplane tickets online.

To ease the task of accounting and administration, we use the micropayment architecture introduced in [10] to provide both accounting and authorization. Briefly, users purchasing bandwidth on BAND-X are provided with credentials that allow them to establish the necessary QoS pipes among the necessary network elements (routers), within the constraints of their contracts. Our use of a trust-management system (KeyNote [9]) allows us to perform both billing and authorization with the same mechanism, simplifying the architecture and eliminating the need for manual configuration or universal trust of the BAND-X service (*e.g.*, to configure the relevant routers of several ISPs).

To better illustrate the use of the BAND-X architecture, we next describe a sample usage scenario involving an end user and several ISPs. In Section 2 we present the system architecture in more detail. Section 3 describes the various components of our system, in particular our micro-checks mechanism, and how they operate together, along with a security analysis. Section 4 describes our prototype implementation, the testbed used for the evaluation, and the measurements collected during the tests we run. Finally, summarize related work in Section 5, and present our conclusions and closing remarks.

1.1. Motivation

QoS provision and management has a wide-ranging literature. A lot of the early work was inspired by the QoS features of ATM networks, and the demand for multimedia traffic. The desired goal was the control of multiplexing behavior in both endpoints and network elements, with the idea that queuing disciplines such as Fair Queuing, or its many variants could be used to allocate bandwidth resources, and for the most part provide delay bounds.

However, despite the ever increasing use of time-sensitive protocols (*e.g.,* VoIP, audio on demand, *etc.*) bandwidth reservation has not been particularly successful. This has been caused mainly by the fear that since these applications have modest bandwidth requirements the operation of a reservation and payment infrastructure would not be feasible economically. Recently, however, newer applications such as video on demand, tele-presence, and Grid Computing, have bandwidth requirements that may constitute a significant portion of the available bandwidth. In such cases the overheads associated with the reservation and billing are smaller (because we are dealing with fewer more expensive reservations), while the benefits are greater because of the impact of the data flows on the infrastructure.

Nowadays, with newer applications such as video on demand, tele-presence, and Grid Computing, the unit of allocation is large enough to allow a relatively smaller number of higher value transactions that place reasonable demands on the reservation and payment components of a reservation system. Such a system must deal with billing (*i.e.,* how the cost of the reserved bandwidth can be paid by the user) and must support a reservation protocol such as RSVP that can perform bandwidth reservation in a scalable and secure manner.

Consider the following scenario of a user Alice wishing to reserve an end-to-end 50Mbps "pipe" from Rome to Dublin[1]. Using an appropriate tool (*e.g.,* auction site, database, service bureau) she decides to purchase a link from Rome to Paris offered by ISP A, and another link from Paris to Dublin offered by ISP B. However, Alice does not need the QoS pipe immediately; rather, she needs it for the time her remote presentation is scheduled, a few days later.

Payment may be effected in various ways (examples given later in the paper) depending on the policy of each ISP. Once the reservation has been booked, each ISP sends a credential to Alice authorizing her to use the required link at the desired time and date and for the appropriate time interval. The credentials are set to expire at the end of the reserved period. Again, depending on the way payment is handled and the policies of the ISPs and other involved parties, more than these two credentials may be required for access to be granted (this is explained later).

Just before the link needs to be established, Alice's QoS negotiation agent (QNA) will send a QoS request to the network elements (NEs) of the two ISPs to ensure that the appropriate resources have been allocated. Since two providers are involved, Alice's QNA will need to contact each ISP separately. Depending on the bandwidth reservation protocol used, Alice's QNA may communicate with a central entity within the ISP, or may negotiate a path through the ISP's network and then reserve the desired bandwidth with each network element separately.

For this discussion, we have limited ourselves to bandwidth reservation; additional QoS requirements (such as latency) may be specified within the same framework.

Spot Market Given an efficient purchasing mechanism, an "advance" booking such as the one mentioned earlier may be made even seconds before the channel will be used, so the term "spot market" is used to define a different payment regime that may be used to sell the unused network capacity. The "spot market" allows premium best-effort services to be sold. In this case, we are not making any promises regarding availability of bandwidth, but we say that by paying a small premium, packets may be treated favorably in the allocation of the remaining bandwidth (after the booked commitments are served).

[1]We use geographical identifiers instead of IP addresses to simplify the example.

2. Architecture

2.1. Operation of the Spot Market

Initially, the various bandwidth providers post their available capacities in the BAND-X clearing house. The system can accommodate one or more such clearing houses, since they function as announcement boards. Apart from that, the clearing house is not involved in the purchase of bandwidth (see Figure 1), but may provide (and charge for) secondary services such as monitoring and reputation/complaints tracking for the participating ISPs, akin to the way commodity markets operators monitor the participating traders.

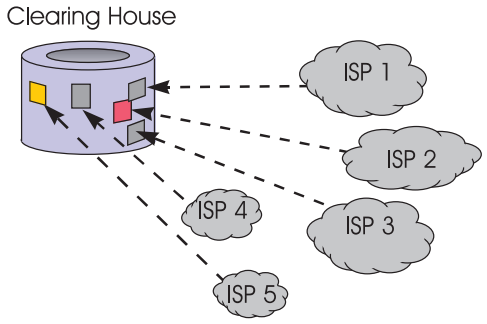

Figure 1. The BAND-X Clearing House acts as a repository of all the offers for bandwidth issued by the ISPs.

The postings are of the form of credentials that describe the identity of the ISP and promise to abide by a set of QoS specifications between two points of the ISPs network. The credential may also contain the time period that the offer is valid (which may be different from the expiration of the credential), the price of the concession, and additional ISP-related information, such as the path that should be taken between the two points. Offer credentials are signed by the ISP who issues them.

Customers contact the Clearing House to collect offers from the ISPs. For complex paths, a customer may need to collect more than one offer and use them together. It is the responsibility of the customer (or someone acting on their behalf) to make the appropriate reservations. In an environment with a single clearing house, the customer can issue queries to get lists of offers matching his or her requirements. If there are many clearing houses, the customer may dispatch an intelligent agent to collect the offers and come back with a recommendation that meets preassigned constraints (price, ISP reliability *etc.*), query each clearing house independently, or use a meta-search engine.

At the end of the search, the customer will hold one or more offer credentials that describe the desired path and QoS specs, as shown in Figure 2.

At this point, the customer has not actually purchased the bandwidth. In order to issue payment and reserve the bandwidth, a number of steps have to be taken. The customer (or the host at one of the end-points of the connection) contacts the first-hop network element (NE) and activates the reservation protocol. The NE issues a challenge which is then returned signed by the customer. This response also contains the offer credentials collected by the customer and a credit-worthiness credential issued by the customer's credit institution, as shown in Figure 3.

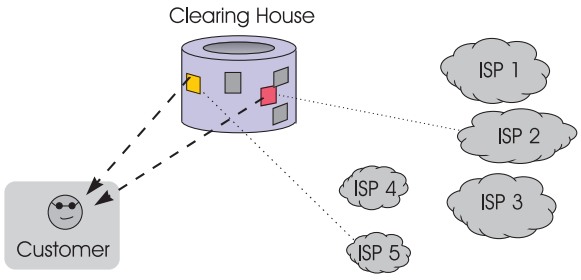

Figure 2. Customer finalizes the path selection by downloading the offer credentials.

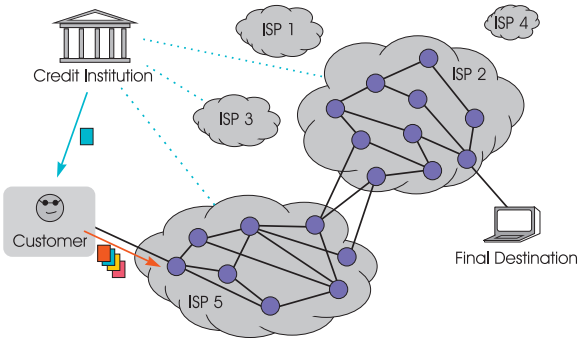

Figure 3. The customer issues a reservation request by sending the offer credentials collected from the BAND-X Clearing House along with a credit-worthiness credential issued by his or her credit institution.

This exchange accomplishes the following: (a) identifies the customer (the key that has signed the NE challenge), (b) provides proof of good standing (the credential issued by the credit institution to the customer's key), (c) limits payment only to the offer credentials provided, (d) can be used only for that particular transaction since it depends on the challenge issued by the NE. On the basis of this transaction, the first hop NE contacts other NEs within the ISPs network establishing the purchased path. If the path crosses ISP boundaries, additional transactions have to be carried out between the NE of the new ISP and the end user, as shown in Figure 4.If an ISP in the path cannot provide the requested bandwidth, the client may have to cancel existing reservations and try to find (and negotiate) another path.

When the last hop is reached, the connection is considered established and the final destination host can initiate a connection with the customer's host over the reserved path (Figure 5).

There is no need for the ISPs offers to match exactly the requirements of the customer. For example, if Alice requires a 50Mbps link from Atlanta to Dublin, she may use an offer for a 100Mbps connection, but purchase only 50Mbps. The providers may include clauses in their offer credentials allowing or prohibiting such un-bundling. The flexibility of the policy language used in BAND-X allows many such special considerations to be encoded within the offer credentials. The advantage of having these restrictions expressed as policy is that they can be used directly by the ISP's infrastructure without any need for conversion. Moreover, the customer cannot alter these restrictions since they are an integral part of the credential (and are protected by the ISP's signing of the offer credentials).

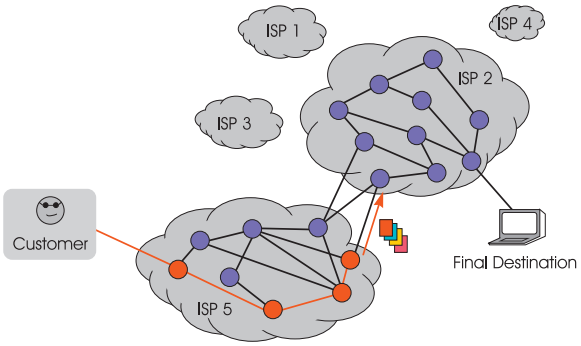

Figure 4. Each time the path crosses ISP boundaries, additional negotiations have to be carried out, to ensure that the next-hop ISP can be paid for passage.

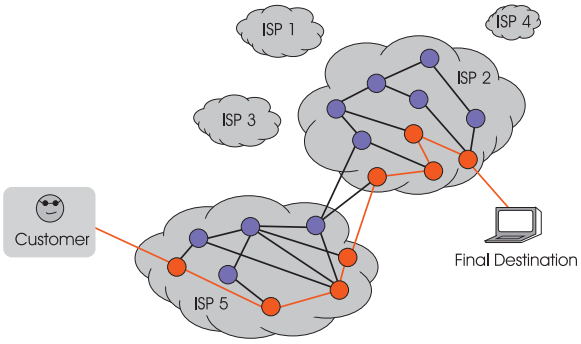

Figure 5. The path has now been established and communication can proceed.

2.2. Operation of the Futures Market

In the Spot Market, the customer collects the offers and sets up the path in short order, because the offers are effective immediately and have a short lifetime. There is no need to negotiate with the ISPs before the reservation.

In the Futures Market the situation is different, since the ISPs need to know what bandwidth has been purchased to plan their resource allocation. Once the customer collects the offers, a notional reservation negotiation will be initiated. The negotiation is notional because no state changes are actually effected on the network elements. The customer's QNA will not detect any change in the negotiation. Within the ISPs network, no path is created; rather the reservation is entered in the ISP's database, and a reservation credential is sent to the end user. This credential will then be used in the same manner as the offer credential was used in the Spot Market scenario. Since the bandwidth has been paid for, the reservation credential commits the ISPs to provide the requested resources at the appropriate future time.

At that time (when the path is actually required) the customer initiates a reservation negotiation, but sends only the reservation credential (instead of the offer and credit institution credentials). The ISP network elements will reserve the path as specified in the reservation credential. The case of multiple ISPs is handled in a similar manner. For popular, pre-planned events, it is possible that groups of ISPs

will create bundles (represented by groups of credentials) that allow for the creation of paths that are predicted to be in high demand, *e.g.*, a path from a large residential ISP to a streaming-content provider, perhaps for the duration of an online music concert.

2.3. Role of the Credit Institution

Like the Clearing House, there is no requirement to have a single Credit Institution. It is, however, important that the ISPs have a way of confirming the keys of the various Credit Institutions. This is because the credit-worthiness credentials (CWCs) issued by the Credit Institutions to their customers will have to be verified by each ISP. If an ISP cannot verify a CWC, then it may be fake; trusting it may result in the equivalent of a bounced check. Furthermore, ISPs may contact the Credit Institution to verify that a user has sufficient funds to pay for a particular transaction (similar to credit card authorization), which means that the Credit Institutions need to be online. However, the interaction between ISPs and Credit Institution is relatively simple, and the experience from real-life credit card payment processors indicate that the infrastructure can scale well.

3. Implementation

3.1. KeyNote Microchecks

The micro-payments system introduced in [10] forms the basis of our approach. The general architecture of this micro-billing system is shown in Figure 6. Under BAND-X, a Merchant is an ISP selling bandwidth and a Payer is a client wishing to make a QoS reservation.

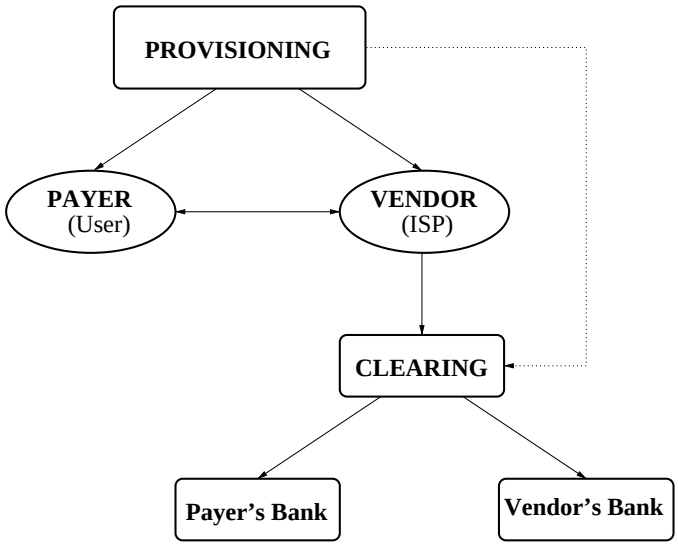

Figure 6. Microbilling architecture diagram.

In this system, Provisioning issues KeyNote [9] credentials to users (Payers) and ISPs (Merchants). These credentials describe the conditions under which a user is allowed to perform a transaction (*i.e.*, the user's credit limit) and the fact that a Merchant is authorized to participate in a particular transaction.

Initially, the ISP encodes the details of the available bandwidth into an *offer* which is uploaded to the BAND-X site, along with a credential that authorizing any user to utilize the bandwidth under the same conditions as those enclosed in the offer. Once the user finds an offer (and associated credential) that is acceptable, she must issue to the ISP a microcheck for this offer. The microchecks are encoded as KeyNote credentials that authorize payment for a specific transaction. The user creates a KeyNote credential signed with her private key and sends it, along with her credential from Provisioning, to the first network element of the ISP. This credential is effectively a check signed by the user (the Authorizer) and payable to the ISP (the Licensee). The conditions under which this check is valid match the offer sent to the user by the ISP. Part of the offer is a nonce, which maps payments to specific transactions, and prevents double-depositing of microchecks by the ISP.

To determine whether he can expect to be paid (and therefore whether to accept the payment), the ISP passes the action description (the attributes and values in the offer) and the user's key along with the ISP's policy (that identifies the Provisioning key), the user credential (signed by BAND-X), the offer credential (signed by the ISP), and the microchecks credential (signed by the user) to his local KeyNote compliance checker. If the compliance checker authorizes the transaction, the ISP is guaranteed that Provisioning will allow payment. The correct linkage among the Merchant's policy, the Provisioning key, the user key, and the transaction details follow from KeyNote's semantics [9]. If the transaction is approved, the ISP can configure the appropriate routers such that the user's traffic is treated according to the offer, and store a copy of the microcheck along with the user credential and associated offer details for later settlement and payment.

Periodically, the ISP will 'deposit' the microchecks (and associated transaction details) he has collected to the Clearing and Settlement Center (CSC). The CSC may or may not be run by the same company as the Provisioning, but it must have the proper authorization to transmit billing and payment records to the Provisioning for the customers. The CSC receives payment records from the various ISPs; these records consist of the offer, and the KeyNote microcheck and credential from the user sent in response to the offer. In order to verify that a microcheck is good, the CSC goes through a similar procedure as the ISP did when accepting the microcheck. If the KeyNote compliance checker approves, the check is accepted. Using her public key as an index, the user's account is debited for the amount of the transaction. Similarly, the ISP's account is credited for the same amount.

3.2. BAND-X Operation

Having seen the overall system architecture, let us look at a particular example. *Alice* is a user who wants to reserve some bandwidth for a particular link with *Nick's* ISP. Every evening Alice contacts her banker and obtains a fresh *Check Guarantor* credential, which allows her to issue KeyNote microchecks. The CG credential shown below (most of the base64 digits from the keys have been removed for brevity) allows Alice to write checks for up to 5 US Dollars, and she can do so until March 24th, 2006.

```
Keynote-Version:  2
Local-Constants:
      ALICE_KEY = "rsa-base64:MCgCIQ...
      CG_KEY = "rsa-base64:MIGJAo..."
Authorizer:  CG_KEY
Licensees:  ALICE_KEY
Conditions:  app_domain == "Band-X" &&
      currency == "USD" && &amount <= 5.00
      && date <= "20060324" -> "true";
Signature:  "sig-rsa-sha1-base64:QU6SZ..."
```

Alice now wants to reserve some bandwidth to Dublin. She searches BAND-X for a suitable offer, and locates one issued by Nick's ISP that contains the following Offer Credential, indicating that she could purchase 50Mbps on the specific link ("Dublin-NYC") for 3 US dollars:

```
Keynote-Version:  2
Local-Constants:
      ISP_KEY = "rsa-base64:7231f..."
      ROUTE_KEY = "rsa-base64:33a41..."
Authorizer:  ISP_KEY
Licensees:  ROUTE_KEY
Conditions:  app_domain == "Band-X" &&
      currency == "USD" &&
      bandwidth <= "50Mbps" &&
      link_name == "Dublin-NYC" &&
      &amount >= 3.00
      && date < "20061120" -> "true";
Signature:  "sig-rsa-sha1-base64:ab1XXA..."
```

As we shall see later, in practice an Offer Credential includes QoS attributes, such as bandwidth, using the Intserv FLOWSPEC notation defined in RFC 2210.
With the offer credential on hand, Alice then writes a check for the appropriate amount:

```
Keynote-Version:  2
Local-Constants:
      ALICE_KEY = "rsa-base64:Mcg..."
      ISP_KEY = "rsa-base64:7231f..."
Authorizer:  ALICE_KEY
Licensees:  ISP_KEY
Conditions:  app_domain == "BAND-X" &&
      currency == "USD" && amount == "4.25"
      && nonce == "eb2c3dfc8e9a" &&
      date == "20060324" -> "true";
Signature:  "sig-rsa-sha1-base64:Qsd..."
```

The nonce is a random number that must be different for each check, guaranteeing that there will be no double-depositing of checks. Alice then sends the Offer Credential and the micro-check to Nick's

router using a protocol such as RSVP. Nick receives these credentials, validates the microcheck to make sure that he will get paid, and configures the router appropriately. If the check is not good, Nick will say so, and refuse to make the reservation. Nick will verify that he will get paid, and will evaluate the Offer Credential and the microcheck using a simple policy such as:

```
Keynote-Version:  2
Local-Constants:
      NICK_KEY = "rsa-base64:7231f..."
      CG_KEY = "rsa-base64:MIGJAo..."
Authorizer:  POLICY
Licensees:  CG_KEY && NICK_KEY
Conditions:
      app_domain == "BAND-X" -> "true";
```

This policy says that anything that Nick's key *and* the Check Guarantor's key jointly authorize is allowed. Thus, Alice must submit a valid payment and a valid Offer Credential. Since the bandwidth was paid for, and a path can be found from POLICY to a user (Alice) that has delegated to Nick's key, which in turn has created an open-access Offer Credential, the operation is allowed. As a matter of business practice, Nick may require periodic payments from Alice in order to keep the bandwidth reserved. Alice must know that and send microchecks at the appropriate intervals.

If additional routers need to be configured in Nick's ISP, the first router forwards the necessary information to the next. Note that it is not necessary for the router itself to perform the signature verifications and policy validations: it can simply refer these operations to a Policy Decision Point (PDP), as is envisioned by the IntServ architecture.

3.3. Security Analysis

Similar to previous work on credential-based micropayments [10, 24], our system has three types of communication: provisioning, reconciliation, and transaction. Although delegation of credentials (and thus access rights to reserved bandwidth) is possible, we do not consider it in this paper. We shall not worry about any value transfers to banks, as there already exist systems for handling those (those used by e-commerce sites, for example). All communications between BAND-X, ISPs, and users can be protected with existing protocols such as IPsec or TLS. This covers both provisioning and reconciliation, which occur off-line from the actual bandwidth reservation and use. Furthermore, the transactions themselves (establishing the QoS pipes, or the right to use existing pipes) can be protected through the same means; the only requirement is that the user can authenticate with each ISP.

The confidentiality of the transmitted data itself is not within the purview of our system, nor is it a responsibility of the ISP; if the users do not trust the network with respect to data confidentiality or integrity, they should use end-to-end security protocols, *e.g.,* IPsec or TLS. We do not impose any limitations that would preclude the use of these protocols.

The user needs to ensure that the ISPs provide the promised service. This can be easily verified by the user using a number of existing protocols and tools [28, 7]. Protecting against over-charging ISPs is also straightforward: the details of each transaction can be verified at any point in time, by verifying the credentials and the offer. Since only the user can create microchecks, a dispute claim can be resolved by "running" the transaction again. Thus, the user is safe even from a collusion between any number of

ISPs and the BAND-X service. The ISP must ensure that they are paid for the services offered. Since it has a copy of all transactions (the BAND-X credential, the microcheck, and the offer), it can prove to the BAND-X, or any other party, that a transaction was in fact performed.

The Credit Institution also needs to be paid for the services offered. Since it handles the microchecks, the ISP has to provide the transaction logs to it. The Credit Institution can then verify that a transaction was done, and at what value. A collusion between the ISP and a user is somewhat self-contradicting: the user's goal is to minimize cost, while the ISP's is to maximize revenue, each at the expense of the other. The function of the Credit Institution is to verify each transaction (perhaps sampling, for very large numbers of transactions), debit the ISP and credit the user (presumably keeping some commission or small fee in the process): if the ISP does not give any credentials to the Credit Institution, then no work was done as far as the latter is concerned (and no payments are made, which benefits the user); claiming more transactions than really happened is not in the best interest of the user (so no collaboration could be expected in the direction), and the ISP cannot "fabricate" transactions. Since value is not stored in either the ISP or the user, only a reliable log of the transactions is needed at the ISP (and, optionally, at the user).

4. Prototype

This section describes our prototype implementation and the environment we used to create a small-scale network to test our implementation. We describe two experiments we ran on our testbed and provide measurements indicating the performance of our prototype in normal reservation situations as well as fault-recovery situations. We based our prototype on ISI's implementation of the RSVP protocol [2], because we did not wish to implement yet another reservation protocol. Nevertheless, we are confident that our concept and mechanism will work with other reservation protocols as well.

4.1. RSVP

The Resource Reservation Protocol (RSVP) is the quality of service signaling protocol we have chosen to support the test implementation of the BAND-X architecture. RSVP is a receiver oriented signaling protocol that allows receivers to request QoS reservations along a network path to any number of senders. The RSVP protocol begins with the senders generating PATH messages that travel through the network downstream to the receivers. PATH messages include information regarding the kind of traffic that the senders will generate and details about the routers along the reservation path. Receivers then generate RESV messages that are sent upstream to the senders specifying the QoS they wish to reserve on each router along the way.

RSVP messages are composed of objects that specify important parameters for the reservation exchange. Two of these objects, RSVP's FLOWSPEC and POLICY_DATA, are relevant to our implementation discussion. A FLOWSPEC contains the requested QoS parameters and the POLICY_DATA object contains information regarding authorization policies for the request. These objects are both checked before a reservation is made to ensure that the request is possible. RSVP uses the FLOWSPEC in admission control to check whether the router actually supports and has adequate resources for the desired QoS. Additionally, policy control checks whether the reservation is authorized using the information contained within the POLICY_DATA object and most likely, a local policy. Both objects were designed to be completely opaque to the RSVP specification. That is, RSVP was not designed for a specific QoS or policy model

so that it could be extended easily for future QoS and policy control services. RFC 2210 specifies an implementation of IETF Integrated Services with RSVP which is probably the most common form of the FLOWSPEC in current implementations. Our test implementation uses the POLICY_DATA object to convey policy information for the BAND-X architecture. RFC 2750 describes the POLICY_DATA object as being composed of any number of policy elements. The information within these elements is application defined and is not dictated by RSVP.

4.2. Implementation

The test implementation we have developed is a modified release of ISI's RSVP distribution (release 4.2a4) [2]. In addition to the BAND-X specific code, development included significant changes to the RSVP daemon and test applications to provide support for protocol features that were not yet implemented. The development process included: (a) the design of a BAND-X Policy Element containing information used for QoS authorization, (b) adding support to the ISI code for the passing of these policy objects during the reservation exchange, and (c) BAND-X specific logic to process the newly supported policy data and make security decisions accordingly.

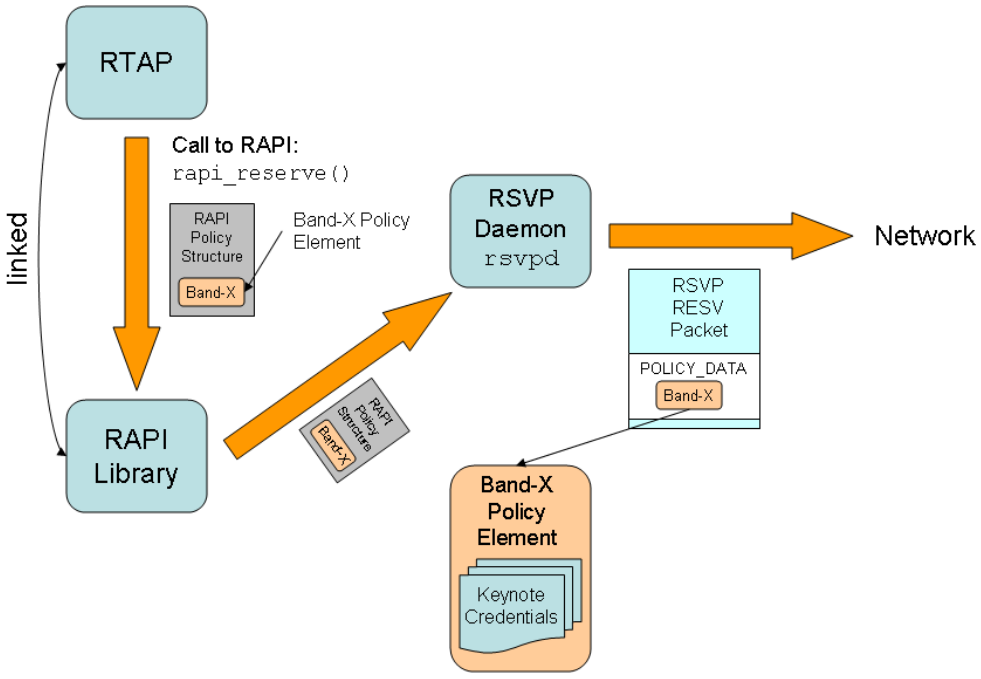

Figure 7. RTAP **places credentials into a** BAND-X **Policy Element, packages it into a** RAPI **Policy Structure, and makes a reservation request via the** API**. The** RAPI **Policy Structure is then sent to the daemon within an** API **reservation request. Finally, the Daemon puts the** BAND-X **Policy Element into a** POLICY_DATA **object and adds it to the outgoing** RSVP RESV **packet.**

4.2.1 BAND-X Policy Element

All information needed for policy decisions within the BAND-X architecture is encapsulated into a policy element. This structure is packaged in a POLICY_DATA object and passed along the reservation route within the RESV message. Our BAND-X policy element is actually very simple. It contains any number of KeyNote credentials that are used to specify policy information and requirements for all parties concerned with the reservation. Each credential is stored simply as ASCII data and a corresponding unsigned integer specifying its length. A set of these credential structures are placed in sequence and another unsigned integer specifies how many there are. The unsigned integer values are encoded into a portable representation (we use Sun's XDR format, RFC 1014) because RSVP itself cannot know the details of the object and therefore it cannot ensure correct byte ordering. The ASCII data is not encoded or compressed in any way.

4.2.2 Adding Policy Control Support to ISI's RSVP

The ISI RSVP distribution, as well as many other implementations, lack support for the policy control mechanisms specified in RFC 2205 and RFC 2750. Providing the bare minimum of these features was needed to allow the transport of our BAND-X policy elements along the reservation path. While the ISI code did provide declarations for the key policy control data structures, we still had to add code to all of the major components of the system. Figure 7 represents an overview of these modifications in the context of ISI's RSVP distribution.

The first such component was the RSVP Test Application (RTAP). RTAP is an application that interfaces with the RSVP daemon process and is used to control a reservation session. RTAP provides a set of commands for creating and closing sessions, sending PATH messages downstream, and of course sending RESV messages to signal a QoS request. In order to pass our policy objects into the daemon process we needed to first provide RTAP commands to specify this. By adding an extra argument to the RTAP reservation command we were able to specify a directory to the application that holds a set of files containing all necessary data for the desired BAND-X policy element. Specifically, these files are just ASCII KeyNote credentials. Our modified RTAP application examines this directory and composes the appropriate BAND-X policy element. This structure is then XDR encoded and placed inside an intermediate object defined by the RSVP API (RAPI). A pointer to this intermediate RAPI policy object is then passed as an argument to RAPI.

The only significant change made to RAPI was inside the code for the `rapi_reserve()` call. The ISI implementation of this routine would simply assume that the RAPI policy object it received via argument was NULL. We modified this behavior to add the policy object to the reservation request sent to the RSVP daemon. A simple routine copies the RAPI policy object into the reservation request structure using a `memcpy`. The reservation request structure is then sent over an IPC socket to the daemon process.

The daemon process is waiting on the other end of this socket for any API requests made through RAPI. Upon receiving a reservation request, the daemon has a structure that contains a copy of the RAPI policy object. Within this object is our own BAND-X policy element constructed earlier within RTAP. At this point the reservation request is translated from an API request to a standard RSVP RESV packet. The daemon treats this newly created packet as if it has arrived from another router, except for the fact that it is in host byte order. We translate the RAPI policy object into the standardized RSVP POLICY_DATA object that is a part of the RESV message. This requires yet another `memcpy` of the opaque BAND-X policy element to the POLICY_DATA structure. After this copy the POLICY_DATA object is inside an

RESV packet that can be sent to the next router in the path.

The final major modification involved changing how RESV packets (received either from a router or from an API request) are processed. To add support for policy control we first check to see if the packet contains any POLICY_DATA objects. If so, we pass these objects to a newly created policy control module. The policy control module that we have implemented is very limited as it only currently supports our BAND-X policy elements, though it could be modified to support others with minimal effort. The policy control module uses a value within the POLICY_DATA object's header called the P-Type to determine what kind of policy element is inside. Every type of opaque policy element is given a unique P-Type value so that RSVP will be able to pass it to a separate module of code that knows how to deal with that element specifically. When the P-Type specifies a BAND-X element the policy control module passes it to the appropriate BAND-X processing routine. The entire process, both the BAND-X processing and policy control, return either a 'yes' or 'no' response. Based on this response the daemon either goes on with the reservation process or generates a policy reservation error message. This error (or any reservation error), in turn, may trigger the cancellation of recently made reservations and the acquisition of fresh credentials (perhaps through a different set of ISPs). It is important to note that this policy control check happens before the admission control check. That is, we check if the user is allowed to make the reservation within the BAND-X system before the check for whether the router even has the resources for the reservation. If there are multiple POLICY_DATA objects within the RESV message we keep checking them until either they all pass or one fails.

4.2.3 BAND-X Processing and Decision Making

When the BAND-X logic is invoked by policy control it has one goal to accomplish; using the policy information given, make a 'yes' or 'no' decision as to whether the customer is authorized to make this reservation. Fortunately, the use of KeyNote credentials to specify BAND-X policy makes this process very simple. First, we need to decode the BAND-X policy object from its XDR representation to the host's. Second, we need to initialize the KeyNote trust management engine and pass it the appropriate credentials, authorizer, and action attribute set. The first credential that is given to the engine is not actually part of the BAND-X object, it is the Local Policy that resides somewhere on the filesystem. This Local Policy is at the highest level of trust and authorizes entities such as the ISP and various credit institutions. Additionally, the public key of our "stub" action authorizer is stored locally and must be added to KeyNote's list of authorizers. The role of this key is explained in detail within the discussion of credentials used in the BAND-X system. We then add all of the credentials contained within the BAND-X policy element to the KeyNote engine.

The final step is to submit all the appropriate values as an action attribute set to KeyNote. To ensure that the reservation is for the appropriate QoS, all parameters within the FLOWSPEC for the reservation are submitted as well. Because KeyNote only supports strings for its action attributes we must convert the floating point and integer values of the Intserv flowspec to string representations. This is done simply by using an appropriate call to `sprintf`. The code is capable of handling both guaranteed and controlled load QoS requests. Once the action attribute set is complete we issue a query to the KeyNote engine and it provides us with the 'yes' or 'no' answer to whether the reservation is authorized. This answer is returned to policy control and then to the RSVP reservation code.

4.2.4 Limitations

A serious consideration during the design and implementation of the prototype was to keep it simple by concentrating only on BAND-X-related aspects. Thus, while the implementation we have developed, works adequately for our testing needs, it does have several important limitations.

- Our prototype intentionally avoids implementing the full RSVP policy control capability, as described in RFC 2205 and RFC 2750. Rather, it concentrates on the exchange of the required BAND-X policy information to each router along the reservation path. Features, such as POLICY_DATA options, which are unrelated to BAND-X operation were not implemented.

- Policy control for multicast reservations was not considered as it is outside the scope of our prototype. We expect that multicast will play an important role in ensuring efficient use of network resources if/when the Internet becomes a significant real-time digital content delivery system. Multicast would make pricing much more variable, since the cost of a reservation in a particular link is amortized over the number of customers subscribing to that reservation. This introduces several challenges, which we leave for future work.

- Policy control is only implemented for RESV messages since the BAND-X architecture does not need policy information within any other type of RSVP messages.

- Error messages generated by policy control failures do not explain how the policy actually failed. Normally, reservation error messages are supposed to contain information saying specifically how the policy failed. Unfortunately, KeyNote does not always report why an action was not authorized by the policy, so the transmission of detailed failure messages is not always possible.

- Although our system can handle link failures within the same ISP (as we show experimentally in Section 4.4), failures in the links between ISPs require the customer to take action to create a new path. Depending on the details of the Service Level Agreement (SLA) that accompanies the path reservation, one or both ISP may instead assume the risk of link failure and create a new path on behalf of (and perhaps unbeknownst to) the customer, using their own credentials (and budget) to pay the cost.

4.3. Experiments

4.3.1 Testbed

Our experiments assume the typical situation where two users wish to establish a path over a number of distinct but interconnected networks. The BAND-X system will then have to negotiate a path over these networks thus creating the link between the two users.

We used our network testbed (NEST) which provides the infrastructure for research in various areas related to networking and network security. The NEST equipment centers on a cluster of 12 machines connected into an adaptable network topology. The machines can accommodate various configurations so that each machine can serve as a network endpoint or an active network element (*e.g.*, router, firewall, etc.). The flexibility of this network provides the enabling infrastructure for research in a number of areas within the overall framework of network security and educational opportunities for under-graduate and graduate students.

The Network Security Testbed can simulate accurately various network topologies and configurations. All the computers have multiple network interfaces so that they can assume the role of routers, bridges, firewalls, or other network elements. The interconnection of all the computers is handled by a high-end Ethernet switch that functions as a virtual plug-board. By changing the configuration of the switch, we can simulate different network topologies without any actual re-wiring. For example, consider the network topology shown on the left side of Figure 8. We can simulate this topology through a set of Ethernet broadcast domains (VLAN configurations), as shown on the right side diagram of Figure 8.

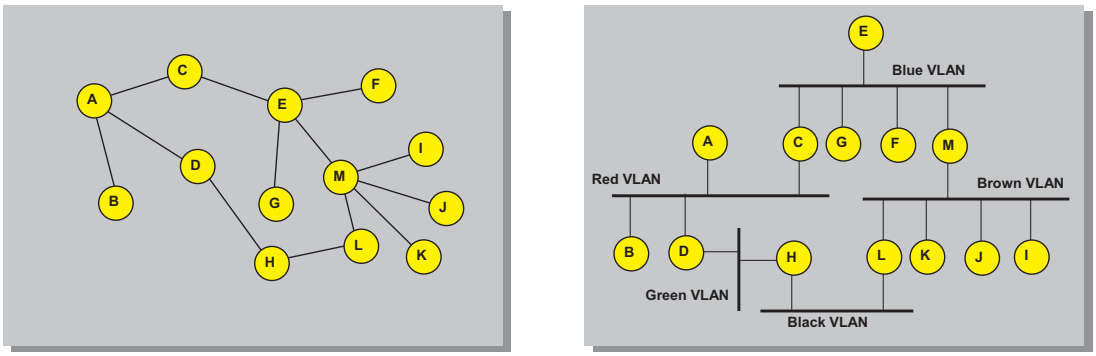

Figure 8. Arbitrary topologies (left) can be represented by configuring VLANs on the Ethernet switch (right).

Some nodes can also be used to impose restrictions on the bandwidth associated with paths that go through them, thus making the simulated environment more realistic. We achieve this by using the dummynet environment [33].

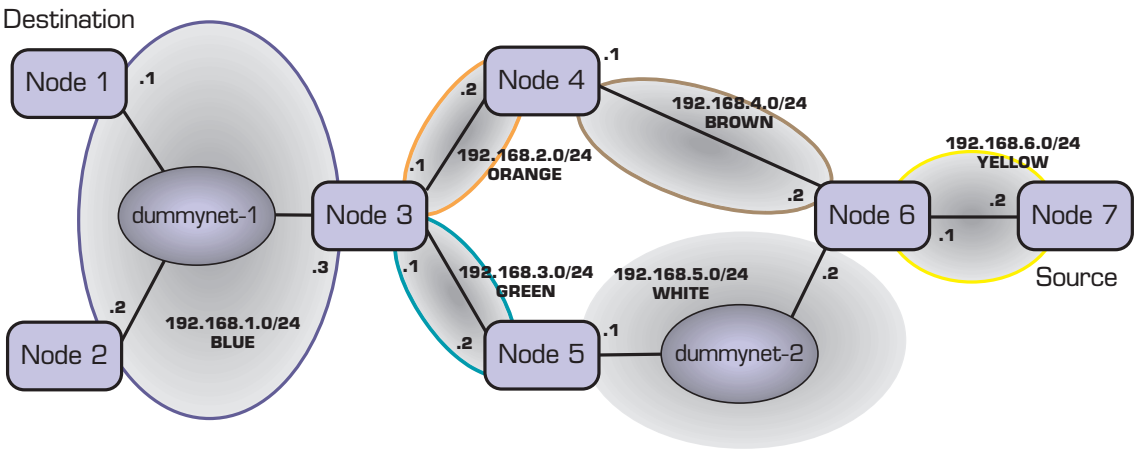

Figure 9. Network used for the BAND-X tests.

Figure 9, shows the topology of the network used to carry out the experiments discussed below. The network consists of three end-nodes (nodes 1, 2 and 7), four nodes acting as routers (nodes 3, 4, 5, and

6). There are also two dummynet nodes that may be used to create disruptions in the data flow. The dummynet nodes are configured as switches, thus they do not require IP addresses, and they do not take part in the RSVP negotiation.

The test layout allows two paths to be created between the two endpoints, thus providing the ability to test the response of the system to network disruptions. The shaded areas define different IP networks, connected by hosts acting as routers. Each node is a single Dell PowerEdge 1550 with an Intel Pentium III (927.11-MHz 686-class CPU) and 256MB RAM.

4.3.2 Normal Reservation Scenario

In this experiment we measure the time taken to create a new path over the network. The intent is to quantify the overhead that we have added to ISI's RSVP implementation. We decided to take timing measurements on a single node along the reservation path. Since the BAND-X processing in the current implementation is practically identical for all hops along the path, recording the computing time at one should give an accurate picture of the complexity per node in a BAND-X enabled RSVP path. We have measured the time it takes from when a RSVP RESV packet is received by the daemon until the time it is forwarded to the next hop. This "in-and-out time" is a sufficient measure because it includes both BAND-X processing and memory copying time.

Figure 10 represents two sets of data. Both sets are the RESV packet in-and-out times as described before. One is the time taken with BAND-X processing enabled. That is, these are measurements of the complete working BAND-X prototype implementation and all the decision making code that this involves. The other set of times is of a system that skips any of the policy decisions that are made in the BAND-X system. We show timings for different RSVP RESV messages containing increasing numbers of credentials. The first message is 2612 bytes and contains exactly 3 credentials. This accurately reflects the approximate minimum message size for any BAND-X enabled reservation in the currently implemented system. This, however, depends highly on particulars such as the cryptographic key length chosen. For this test we used RSA with 512-bit keys encoded into base64 ASCII text. Each additional set of measurements adds a single credential to the policy object. This credential is essentially a direct copy of the BAND-X offer credential. The credential itself if 936 bytes, and thus the packet size increases uniformly linearly.

Number of Credentials	Packet Size (bytes)	Mean Time $\pm$ 95% CI With BAND-X (usec)	Mean Time $\pm$ 95% CI Without BAND-X (usec)	Mean Difference (usec)
3	2612	4243.69 ± 26.42	2015.14 ± 15.32	2228.55
4	3548	4908.68 ± 21.24	2039.33 ± 15.52	2869.35
5	4484	5589.59 ± 22.69	2060.68 ± 11.90	3528.91
6	5420	6248.29 ± 26.31	2097.92 ± 40.33	4150.37
7	6356	6897.98 ± 23.21	2099.61 ± 21.07	4798.37

The overhead the BAND-X code adds to the reservation process can be gleaned by examining the average difference between the times for each size. As packet size increases, there appears to be a slight growth in the computing time of ISI's RSVP daemon that does not include the final BAND-X processing and decision making code. This growth appears to be fairly negligible when compared to that of the BAND-X enabled daemon. This overhead can be attributed to a number of things that are done

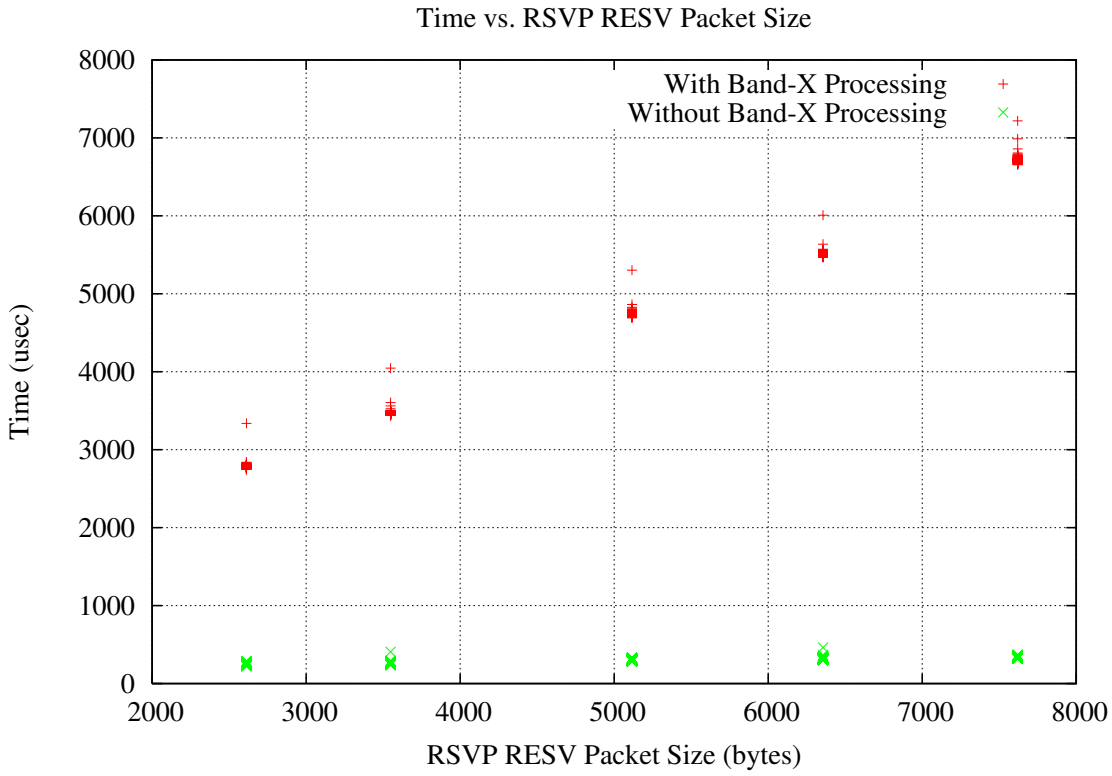

Figure 10. Times elapsed between receiving and forwarding of RSVP RESV packets. The X-Axis shows the number of credentials within messages. All measurements taken at a single node along the reservation path. One hundred samples were taken for each credential set. In addition, a linear fit is depicted. The machine was a Dell PowerEdge 1550 with an Intel Pentium III (927.11-MHz 686-class CPU) and 256MB RAM.

within the BAND-X processing routines. The setup and query that is performed with the KeyNote Trust Management Engine accounts for most of the overhead. This involves going through each credential, adding it to the KeyNote session, and then invoking the KeyNote query itself. The apparent linear growth as a function of the packet size, or more accurately, the number of credentials within the RSVP packet, is a result of the linear complexity of the KeyNote query and these memory copies.

The data also shows that there does not seem to be a great deal of cost for larger RESV packets in the non-BAND-X related code of ISI's RSVP daemon. This allows us to concentrate specifically on the BAND-X decision and processing code that is called at the moment of reservation. Optimizations should target this portion of the code as it is the obvious bottleneck. One obvious optimization would be to develop a slightly more sophisticated policy object that identified which sets of credentials were applicable to particular domains. Currently, we just add every credential that is in the policy object to the keynote session and let keynote sort out which ones are applicable to the decision. Results from the experiment show that this approach is costly. A better approach would be to pack the credentials with information that indicated what domain they were for. With that approach, we would expect to see a

Sign	Verify	Sig/sec	Ver/sec
0.0037 sec	0.0002 sec	270	5055

Table 1. Signing and verification times for 1024-bit RSA keys.

constant processing cost equivalent to the 3 credential range. This type of simple optimization would greatly improve the computational scalability of the system.

It is important to note that the above analysis does not provide any consideration for impact that the increased RSVP RESV packet size has on network transmission latency. With an increase in packet size by a rough factor of at least 20, this should perhaps be taken into account. Compression of the credentials could mitigate this overhead. In addition, credentials can be dropped from the policy object as it is forwarded through a new domain. This particular optimization theoretically would allow the RSVP message to shrink as it goes through the path.

Despite the significant increase in processing latency, we believe that this overhead is acceptable in most of the scenarios where BAND-X would be used, since the setup cost would be amortized over a long-lived session. The primary reason for minimizing this overhead is to allow for quick path reconstruction when a failure occurs. We believe that the optimizations we identified above are promising in minimizing (but not altogether eliminating) overheads.

To provide a more complete overhead analysis, we measured the number of public key verifications we can perform, which indicates how many credentials/payments the back-end systems (payment infrastructure) can validate per unit time. We used a 3 GHz Pentium4 processor machine running Linux with the OpenSSL V 0.9.7c library for the measurements. As shown in Table 1, a single such system can validate over 5,000 credentials per second. Assuming a scenario where reservations utilize 20 links within a single ISP, that ISP's back-end system would be able to process 250 such reservations per second. If a higher load is expected, it is relatively straightforward to use hardware cryptographic accelerators [26] or add more back-end processing systems.

4.4. Recovery from Route Failure

In this experiment we investigate the response of the system in the event of a change in the routing path. Such changes may be due to external factors (*e.g.,* link failure) that affect an already established reservation. This experiment uses the BAND-X system to reserve a path over our testbed network with a redundant route. We then simulate a route failure over the reserved path and examine RSVP's and BAND-X's ability to recover, re-propagate, and establish a new reservation along the alternative path. Finally, we provide some rough timings to give an idea of the service interruptions that might occur with such a scenario.

4.4.1 Procedure

The test begins with the BAND-X enabled RSVP daemon being run on all nodes in the network with the exception Node 2, whose role will be explained later. The dummynet bridges are unused in the exercise, RSVP is not running on them and no reservations are made on their interfaces. Node 1 is designated the RSVP receiver and Node 7 the sender. The process begins with Node 7 sending a RSVP PATH message

addressed to the receiver of the RSVP session, Node 1. This PATH message propagates through the network on the lower path designated with a bold line in the diagram below.

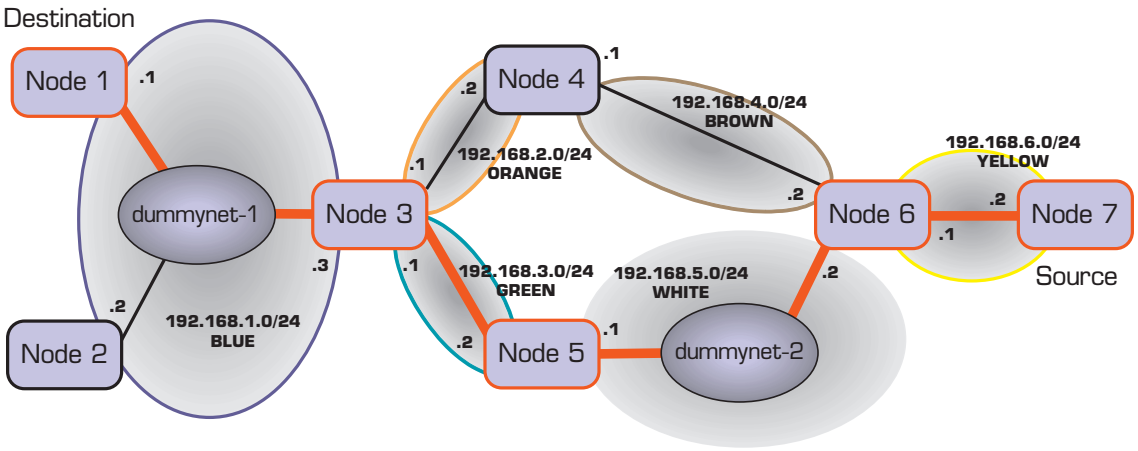

Figure 11. Initial path from Node 7 to Node 1.

This PATH message is forwarded through the network along the highlighted path based upon the kernel routing tables, not by any intervention from RSVP or BAND-X. The PATH message does not contain any BAND-X related information and is completely unchanged from how the original ISI's RSVP implementation generates it. When the PATH message is received by Node 1 it examines it and issues an RSVP RESV message detailing the desired QoS and the necessary credentials bundled within a BAND-X Policy Object. This RESV message is sent specifically to the next hop (Node 3) in the recently established reservation path. The BAND-X enabled RSVP daemon on Node 3 examines the RESV message, extracts the policy information from the message, and runs the BAND-X processing and decision making routines. We will shortly detail the exact credentials used in this experiment but for the sake of discussion assume that this check is made and the policy allows the reservation. Then the QoS parameters will be set on Node 3 and it will forward the RESV message to the next hop (Node 5) in the path. This process will continue until the RESV message reaches the sender and all BAND-X checks and reservations have been made. Once this process is complete the reserved path is ready to be used. However, in order to keep the reservations intact, RSVP must send periodic PATH and RESV refresh messages. If these messages are not received by a node in the path, its reservation state will timeout and the QoS settings will be reset.

After this reservation path has been established a route failure is simulated by a simple manual change to the routing tables on Node's 3 and 6. The alternative route is entered into their tables and the PATH and RESV refresh messages begin propagating through the alternative route.

Node 4 will ignore any RESV refresh messages until it receives a PATH message. When it does receive a PATH refresh, the PATH state is created and then upon receiving the next RESV message it makes a BAND-X policy check and if it passes on the new route the reservation is made. The reservation made on Node 5 will timeout eventually because it will no longer receive refresh messages. The reservations made on interfaces of Node 3 and Node 6 for the first path will be switched to the new path and service is restored.

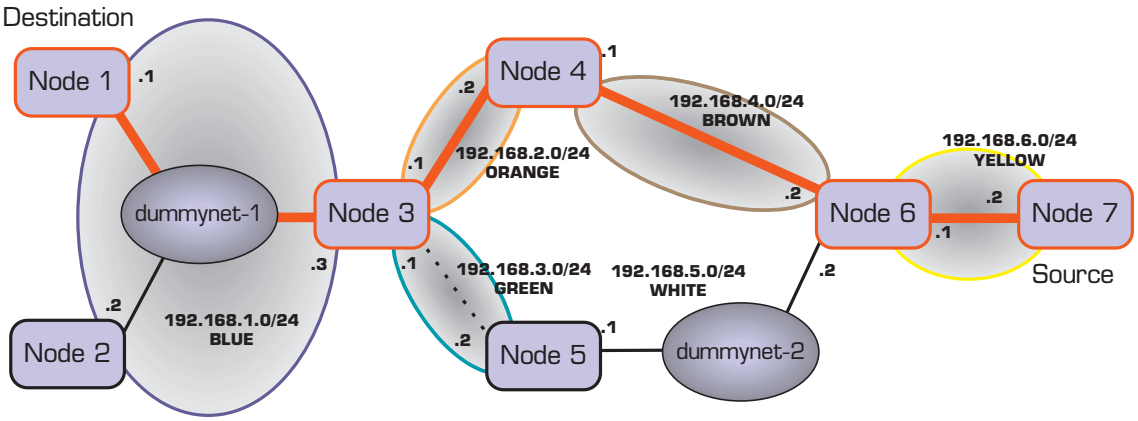

Figure 12. After a failure in the lower branch, a new path is created via the upper branch.

4.4.2 Credentials Used

The credentials used for this experiment show a fairly straightforward use of the BAND-X system. Each node has a policy credential of the form:

```
Keynote-Version:  2
Local-Constants:
      ISP_KEY = "rsa-base64:MEgCQQC/H..."
      GC_KEY = "rsa-base64:MEgCQAAE=..."
Authorizer:  "POLICY"
Licensees:  CG_KEY && ISP_KEY
Comment:  This is the local policy for making bandex reservations.
Conditions:  app_domain == "Band-X" -> "true";
Signature:  "sig-rsa-sha1-base64:ab1XXA..."
```

The reservation request is made with four credentials. They consist of a credit institution credential signed by the bank, a check credential signed by Alice, and two offer credentials issued from the same ISP. The bank and check credentials are virtually identical to the example provided earlier. However, the offer credentials differ from the ones presented in the earlier example. Most notably, there are two, one for each path in the network.

```
Keynote-Version:  2
Authorizer:  ISP_KEY
Licensees:  ROUTE1_KEY
Local-Constants:
      ROUTE1_KEY = "rsa-base64:MEgCQQCf05..."
      ISP_KEY = "rsa-base64:MEgCQQC/HQ..."
Conditions:  app_domain == "Band-X" && currency == "USD" &&
      link_name == "Dublin-NYC" && &amount >= 3.00 &&
      date < "20060924" &&
      flowspec_service_type == "CL" &&
      flowspec_bucket_rate == "1.000000" &&
      flowspec_bucket_size == "1.000000" &&
      flowspec_min_unit == "1" &&
      flowspec_max_packet == "1" -> "true";
Signature:  "sig-rsa-sha1-base64:enN+vj+6..."
```

This offer has the Bandwidth specified as an Intserv flowspec specifying the QoS parameters. The numbers are simply all 1's for sake of simplicity. Note that this credential is signed by ROUTE1_KEY. Alternatively, the other offer credential is identical except it is signed by a different route key (ROUTE2_KEY). ROUTE1_KEY is stored on every node in the first path and is used as the action authorizer for the KeyNote query when BAND-X performs its policy check. Node 4 on the other path stores a copy of ROUTE2_KEY locally and uses that as its action authorizer in the query. Thus, when the route switches, the chain of authorization goes through the offer credential signed by ROUTE2_KEY.

4.4.3 Results

The exercise was executed for ten trials to test whether our architecture could handle this type of service disruption gracefully. In all ten trials the alternative route was established successfully upon failure of the original route and all BAND-X policy decisions were executed to ensure policy compliance. To get a rough idea of the length of service disruption such a scenario could cause we decided to introduce Node 2 as an external monitoring node. Node 2 would begin querying Node 4's PATH and RESV state when the routing tables were changed on Nodes 3 and 6. This monitoring allowed us to time approximately how long it took the path and reservations to be reestablished.

Trial Number	PATH Time (s)	RESV Time (s)
1	24	28
2	16	38
3	36	38
4	34	50
5	28	54
6	32	66
7	6	18
8	4	24
9	16	35
10	42	68

The table above shows the times measured for each of the ten trials. The PATH time represents the time in seconds it took for the RSVP daemon on Node 4 to reestablish a path state by receiving a PATH refresh message. Similarly, the RESV time represents the time in seconds it took to receive the RESV refresh message, perform the BAND-X policy check, and make the reservation. It is important to note that these numbers were gathered by querying the node over the test network from the monitor node. This was done every 2 seconds in order to limit the amount of network and processing expense. Thus these numbers are at best off by plus or minus 2 seconds. Additionally, there was the obvious overhead needed to perform the query. These measurements were taken simply to give a rough idea on the order of magnitude that we were dealing with in terms of service disruption time. That being said, we can see that the time seems to range from roughly 20 seconds to over a minute. This wide variability can be attributed to when exactly the route failure occurs. If it occurs at an opportune moment right before a PATH refresh message is to be sent out then the time will be relatively short. Alternatively, it could have just missed a message and be stuck waiting for it.

We can see from the data that within the current implementation of the BAND-X system using ISI's RSVP implementation, that service disruption will be considerable in the case of route failure. It is obvious that a disruption of over a minute in some cases could be unacceptable for a real-time application requiring QoS support. This is completely a function of RSVP and its use of soft state reservations and not a limitation on the BAND-X architecture itself. In fact, if we decrease the time between RSVP refresh messages then we could drastically reduce the time in which the route failure is detected and recovered from. Though, the corresponding increased load on the network from the now much larger RESV messages could be prohibitive.

5. Related Work

5.1. Grid Computing

In Grid Computing, efforts are already underway to make the network a schedulable resource, just as compute and data resources are. The Grid High-Performance Networking (GHPN) [35] research group, part of the Global Grid Forum (GGF), has been formed to address issues of Grid support in optical networks. This work recognizes the need for user controlled dynamic provisioning of network resources, in which said resources are owned by users. Such work will be vital in allowing Grid applications to utilize modern optical networks. [1, 15, 3, 4, 5]

Work sponsored by Canarie Inc. has lead to the development of User Controlled LightPath (UCLP) [22], designed to allow end-users to create end-to-end light paths (optical links that allow unstructured access to the fiber infrastructure) by combining individual segments very much as we described in the introduction. The current systems, however, are targeted towards the academic community and hence assume that end-users have the required expertise and have non-competitive usage strategies. Specifically under the "User Controlled Light Paths" framework [22], (a) end-users have to be known by the system in advance, (b) policy enforcement is not addressed, (c) there is no purchasing of bandwidth, since the network is considered a common resource. In a commercial environment, a similar system must deal with billing (*i.e.*, how the reserved bandwidth can be paid by the user) and must support bandwidth reservation in a scalable and secure manner.

Motivated by Canarie's signaling approach, others [37, 21] have tried to provide autonomous domains the ability to enforce their own management policies. This work, similar to BAND-X tries to bridge the

gap between independently managed network domains and their policies. An approach presented in [37] foregoes the lightpath repository of UCLP and instead queries domains for their best appropriate and available lightpath segments. This realtime lightpath search allows autonomous domains to check local management policy at the time of the reservation request. In BAND-X, any such management policy would be a part of the offer credential that would be presented to the domain's Policy Decision Point (PDP). A similar approach is presented in [21] that uses dedicated AAA (authentication, authorization and accounting) agents within domains to perform policy checks and provide an authorization token to be presented at reservation time. BAND-X possesses some advantages to both of these architectures when considering the centralized nature of their approaches. The presence of a single centralized "PolicyServer" or "AAA agent" for each domain would provide for single point of failure. Such a problem could occur from direct failure or through the PDP becoming isolated due to partitioning of the network as a result of congestion, failure, or attacks. These issues could somewhat be alleviated by employing multiple PDP's kept in sync via replicated databases. However, these techniques would introduce their own scalability issues for large domains. BAND-X does not suffer from such centralization problems, at least in the reservation phase, because the policy decision is made on-site at each relevant network element. This provides greater insurance that customers who posses a copies of Band-X offers will not encounter reservations failures due to network failure of non-relevant nodes. That being said, issues of revocation are not handled in BAND-X as they are much more difficult without a centralized policy database. However, we feel the costs of such a limitation in a bandwidth market will be negligible in the face of gains made from selling unused resources.

The problem of jointly enforcing a Virtual Organization's (VO) policy and a resource's policy has been addressed in the literature. [31, 30]. Where a VO's policy delegates what a user, as a member of the VO, can do with a Grid resource. A local resource's policy limits the user even further to actions allowed by the resource's owner. Theoretically, reservations could be made on routers that provide network QoS as a schedulable Grid resource to members of a VO. The Community Authorization Service [31, 30] is similar to BAND-X in its use of "signed assertions", like Keynote, to provide on-site evaluation of policy at the resource. Users access a CAS which provides them with a signed assertion containing their identity and provided privileges as a member of the VO. This assertion is then presented to the resource. The resource verifies the conditions of the assertion on behalf of the VO and additionally ensures none of its own local policies are being violated. While this system uses a similar approach to BAND-X it was not designed to support a market for QoS reservations. BAND-X uses Keynote to achieve joint authentication between the ISP and a credit institution that vouches for the customer's payment. It might be possible to develop extensions to the CAS to allow users to provide payment and credit credentials to the service as part of the authentication. In this system it would be the ISP or a collection of ISP's and their customers would be in the role of a VO. The authors do not discuss such an extension to their scheme as it wasn't a goal but it seems possible based on their descriptions. Another system similar to CAS is VOMS [6]. VOMS provides signed assertions that contain group membership associations of the user. It is up to the resource to enforce any VO policy using information contained in the VOMS assertion. [31] notes that such and approach is less centralized than CAS and will lead to difficulty supporting dynamic VO policies.

5.2. Billing

Internet telephony (or voice over IP) is widely considered to be the "killer" application that will convince users that they need QoS (and the higher prices this implies). This is underlined by the fact that the literature concentrates on QoS for VoIP applications. Systems such as OSP [17] provide a way for large organizations to settle payments related to VoIP call clearing. Although OSP is very close to BAND-X, it does not involve the end-user, but instead concentrates on the ISPs. For example OSP only exchanges Call Detail Records, the ISPs are responsible for handling customer billing and payment. In other words the model is that of the traditional TELCO whereby payment is handled either via prepaid cards, or monthly telephone bills. BAND-X is not bound to a particular signaling mechanism (such as H.323) and provides far greater flexibility in that users that have no prior relationship with an ISP can use the reservation protocol and pay for their bandwidth. Although many papers have been written on market-based routing (*e.g.*, [18], [27], [32], [34], [19]) these are concerned with the use of market-based techniques in routers, ignoring the problems of accounting, billing and payment. BAND-X can use any router that supports a reservation protocol (and the BAND-X extensions).

5.3. Secure QoS Reservations

A secure reservation protocol is required to provide a number of assurances including (a) that only authorized users can make reservations, (b) that a reservation made by a user can be traced back to that user, and (c) that users cannot make reservations over their allocated quota. These are to protect against starvation or, perhaps even worse, denial of service that can occur when multiple unauthorized requests result in the allocation of all available bandwidth thus preventing legitimate users from reserving bandwidth. The above considerations imply some authentication mechanism and the use of integrity checks on the transmitted data. OSP runs over TLS which encrypts the exchanged data. X.509 certificates are used to authenticate both ends of a transaction. However, this secure communication is used only for the data exchange between the ISP nodes running OSP. Customer identification is still handled via a separate system that is operated by the ISPs and usually involves some kind of PIN or password authentication. In [16] the actual charging is delegated to a "payment-agent" that is assumed to run on the same machine as the user. However, no details are provided on how the "payment-agent" effects payment.

All the systems we have looked at assume that the user trusts some provider who determines the cost of the connection. No system tries to empower the user by providing choice. BAND-X allows the user to select the best (as defined by the user) providers to handle the connection and makes sure that at the end of the day everybody gets paid. This approach is far superior to the piecemeal approaches found in the literature.

5.4. Scalability

Each reservation carries with it some overhead. This includes both protocol overhead, but also state that must be maintained by routers for each reservation. As the number of reservations increases so does the overhead. Unless there is some kind of aggregation of requests this overhead will ultimately define an upper bound on the number of reservations that can be accommodated by the existing infrastructure. The complexity of some of the proposed systems (*e.g.*, [25], and [16]) and the small scale of their test-beds (*e.g.*, 200 nodes in [23]) casts grave doubts on their ability to scale to millions of users and thousands

of network elements. Various techniques that attempt to improve scalability through aggregation are vulnerable to abuse. For example, in [39] the authors describe request aggregation whereby multiple requests are merged into a single larger request for the total bandwidth asked for by the individual requests. This approach, however, may result in an upstream node declining the single request thus denying access to all the requests, even through some of the individual requests could have gone through [36].

Since BAND-X covers both reservation and payment, the problem of scalability has to be addressed in both areas. As far as reservation is concerned, BAND-X uses the RSVP protocol and so can take advantage of the optimizations and efficiencies that have either been integrated, or are being considered for inclusion into the protocol. In the area of billing, the use of the KeyNote-based micro-payment architecture has been shown to scale well [10].

5.5. Signaling

The BAND-X system is not dependent on a specific signaling mechanism such as RSVP . A signaling protocol simply provides a means for passing the BAND-X credentials to the relevant network elements. Many signaling protocols have been proposed to address issues with RSVP in terms of per-flow overhead, simplicity of implementation, explicit routing, and support for broader and shared resource reservation. The YESSIR protocol [29] uses the Real-time Transport Protocol (RTP) to perform in-band signaling of QoS parameters. The motivation behind the protocol is to improve upon the overhead and complexity associated with RSVP and similar protocols. YESSIR employs a sender based reservation process to eliminate the need for tracking of next hops that RSVP implementations must handle because of its receiver oriented nature. RTP is used to reduce the need to modiy of existing multimedia applications that require differential QoS and are already using RTP. Currently the protocol does not support features for authentication. Boomerang [20], another protocol developed with similar goals, tries to limit per-flow overhead as much as possible. It uses ICMP messages to signal network elements and simpler QoS parameters to decrease state on routers. Boomerang achieves much smaller messages sizes than RSVP at the cost of sacrificing some functionality. BAND-X can easily be integrated with either of these protocols to serve as their authentication and policy enforcement mechanism. For example, in the case of YESSIR the integration would simply involve modifying the RTCP protocol messages to include BAND-X credentials.

6. Summary and Concluding Remarks

To minimize network congestion which can cause complaints and dissatisfaction among users, ISPs overprovision their networks [13]. Unfortunately, unused bandwidth is wasted since it cannot be saved for later use. While bandwidth remains cheap, the ISPs can continue to add capacity ahead of the actual demand, but this state of affairs will only last as long as users of time-sensitive services prefer the telephony network. The enormous cost difference between the telephony network and the Internet provides an implicit subsidy. However, as users switch to the Internet for their time-sensitive services, ISPs will no longer be able to expand their networks. We believe that the framework described in this paper offers a migration path for both users and ISPs through the creation of an open market for bandwidth over the Internet. The reason is that the BAND-X framework supports a competitive market offering transparency, and security. At the same time the low overheads of the BAND-X framework

ensure scalability through the use of a micro-payment environment.

The benefits offered by BAND-X include: (a) "instant" purchases of bandwidth and advanced purchases allowing the ISPs to plan ahead their resource allocation strategies, while being able to auction off unused capacity rather that letting it go at Best-Effort prices, (b) efficiency, requiring only a few exchanges between a buyer and sellers to effect a reservation. Moreover, the use of the KeyNote-based micro-payment framework provides system-wide efficiency and scalability, (c) compatibility with existing standards: by utilizing an existing reservation protocol (RSVP), a BAND-X system may be be deployed with minimum disruption. (d) trades between parties that have no established business relationships: The Credit Institution(s) link buyers and sellers, thus allowing a transaction to go through without the need for a buyer to be known to the seller. This is a key requirement for the bandwidth market to work freely with the buyer being able to select the seller offering the best value for money. (e) openness: the BAND-X model allows the presence of multiple entities for each role (*i.e.*, we can have multiple Credit Institutions, Clearing Houses, buyers and sellers) operating within a single market. This increases the competition and overall reliability of the entire system.

References

[1] Global Lambda Integrated Facility (GLIF) Homepage, `http://www.glif.is/`.

[2] ISI RSVP Distribution, `http://www.isi.edu/rsvp/`.

[3] StarLight Homepage, `http://www.startap.net/starlight/`.

[4] SURFnet Homepage, `http://www.surfnet.nl`.

[5] UKLight Homepage, `http://www.uklight.ac.uk`.

[6] R. Alfieri, R. Cecchini, V. Ciaschini, L. Agnello, A. Frohner, A. Gianoli, K. Lorentey, and F. Spataro. VOMS: An authorization system for virtual organizations. In *Proceedings of the* 1^{st} *European Across Grids Conference*, February 2003.

[7] AtWatch Advanced Website Monitoring. `http://www.atwatch.com/`.

[8] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss. An Architecture for Differentiated Services. Technical report, IETF RFC 2475, December 1998.

[9] M. Blaze, J. Feigenbaum, J. Ioannidis, and A. D. Keromytis. The KeyNote Trust Management System Version 2. Internet RFC 2704, September 1999.

[10] M. Blaze, J. Ioannidis, and A. D. Keromytis. Offline Micropayments without Trusted Hardware. In *Proceedings of the Fifth International Conference on Financial Cryptography*, 2001.

[11] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. Resource ReSerVation Protocol (RSVP) – Version 1 Functional Specification. Internet RFC 2208, 1997.

[12] L. Burgstahler, K. Dolzer, C. Hauser, J. Jähnert, S. Junghans, C. Macián, and W. Payer. Beyond Technology: The Missing Pieces for QoS Success. In *Proceedings of the ACM SIGCOMM Workshop on Revisiting IP QoS (RIPQOS), held in conjunction with the ACM SIGCOMM conference*, August 2003.

[13] M. Currence, A. Kurzon, D. Smud, and L. Trias. A Causal Analysis of Usage-Based Billing on IP Networks, 2000.

[14] B. Davie. Deployment Experience with Differentiated Services. In *Proceedings of the ACM SIGCOMM Workshop on Revisiting IP QoS (RIPQOS), held in conjunction with the ACM SIGCOMM conference*, August 2003.

[15] T. DeFanti, C. de Laat, J. Mambretti, K. Neggers, and B. S. Arnaud. TransLight: a global-scale LamdaGrid for e-science. *Communications of the ACM*, 46(11), November 2003.

[16] R. J. Edell, N. McKeown, and P. Varaiya. Billing Users and Pricing for TCP. *IEEE Journal on Selected Areas in Communications*, 13(7):1162–1175, 1995.

[17] ETSI. Telecommunications and Internet Protocol Harmonization Over Networks (TIPHON): Inter-domain pricing, authorisation, and usage exchange. In *ETSI DTS/TIPHON-03004 V1.3.0 (1998-09)*. 1998.

[18] G. Fankhauser, B. Stiller, C. Vogtli, and B. Plattner. Reservation -based Charging in an Integrated Services Network. In *Proceedings of the 4th INFORMS Telecommunications Conference*, Boca Raton, Florida, USA, March 1998.

[19] E. Fulp, M. Ott, D. Reininger, and D. Reeves. Paying for QoS: an optimal distributed algorithm for pricing network resources. In *Sixth International Workshop on Quality of Service (IWQoS'98)*, pages 75–84, 1998.

[20] G. Feher, K. Nemeth, and M. Maliosz. Boomerang: A simple protocol for resource reservation in ip networks. In *IEEE Workshop on QoS Support for Real-Time Internet Applications*, June 1999.

[21] L. Gommans, B. van Oudenaarde, F. Dijkstra, C. de Laat, T. Lavian, I. Monga, A. Taal, F. Travostino, and A. Wan. Applications Drive Secure Lightpath Creation across Heterogeneous Domains. *IEEE Communications Magazine*, 44(3), March 2006.

[22] H. Guy. Everything you ever wanted to know before you use and/or deploy UCLP on your advanced network. In *Proceedings of the CANARIE's Advanced Networks Workshop*, November 2004.

[23] F. Hao. Scalability Techniques in QoS Routing, 2000.

[24] J. Ioannidis, S. Ioannidis, A. Keromytis, and V. Prevelakis. Fileteller: Paying and Getting Paid for File Storage. In *Proceedings of the Sixth International Conference on Financial Cryptography*, March 2002.

[25] W. Jarrett, T. Michalareas, and L. Sacks. Operational Support Issues for IP QoS Based Networks. In *Proceedings of the IEE Services over the Internet Colloquium*, June 1999.

[26] A. D. Keromytis, J. L. Wright, and T. de Raadt. The Design of the OpenBSD Cryptographic Framework. In *Proceedings of the USENIX Annual Technical Conference*, pages 181–196, June 2003.

[27] H. Kneer, U. Zurfluh, G. Dermler, and B. Stiller. A business model for charging and accounting of internet services. In *EC-Web*, pages 429–441, 2000.

[28] H. Ludwig, A. Keller, A. Dan, R. King, and R. Franck. A service level agreement language for dynamic electronic services. *Electronic Commerce Research*, 3(1-2):43–59, 2003.

[29] P. Pan and H. Schulzrinne. Yessir: A simple reservation mechanism for the internet. *Computer Communication Review*, 29(2), April 1999.

[30] L. Pearlman, C. Kesselman, V. Welch, I. Foster, and S. Tuecke. The Community Authorization Service: Status and Future. In *Proceedings of the Conference for Computing in High Energy and Nuclear Physics*, March 2003.

[31] L. Pearlman, V. Welch, I. Foster, C. Kesselman, and S. Tuecke. A Community Authorization Service for Group Collaboration. In *Proceedings of the IEEE* 3^{rd} *International Workshop on Policies for Distributed Systems and Networks*, 2002.

[32] P. Reichl, G. Fankhauser, and B. Stiller. Auction models for multi-provider internet connections. In *MMB (Kurzvortrage)*, pages 71–75, 1999.

[33] L. Rizzo. Dummynet Home Page, 1999.

[34] N. Semret and A. Lazar. Spot and derivative markets in admission control. In *Proceedings of 16th International Teletra Congress*, pages 925–941, 1999.

[35] D. Simeonidou, R. Nejabati, G. Karmous-Edwards, J. Leigh, F. Travostino, B. Berde, and F. Dijkstra.

[36] M. Talwar. RSVP Killer Reservations, IETF Draft (draft-talwar-rsvp-kr-01.txt), 1999.

[37] D. L. Truong, O. Cherkaoui, H. Elbiaze, N. Rico, and M. Aboulhamid.

[38] D. M. Turner, V. Prevelakis, and A. D. Keromytis. The Bandwidth Exchange Architecture. In *Proceedings of the* 10^{th} *IEEE Symposium on Computers and Communications (ISCC)*, pages 939–944, June 2005.

[39] L. Zhang, S. Deering, and D. Estrin. RSVP: A new resource ReSerVation protocol. *IEEE network*, 7(5), September 1993.