Internet Engineering Task Force R. Guerin/A. Orda/D. Williams
INTERNET DRAFT IBM/Technion/IBM
5 November 1996

# QoS Routing Mechanisms and OSPF Extensions
draft-guerin-qos-routing-ospf-00.txt

## Status of This Memo



## Abstract

This memo describes extensions to the OSPF protocol to support QoS routes. The focus of the document is on the algorithms used to compute QoS routes and on the necessary modifications to OSPF to support this function, e.g., the information needed, its format, how it is distributed, and how it is used by the QoS path selection process. Aspects related to how QoS routes are established and managed are also briefly discussed, but the development of detailed specifications is left for further study. The goal of this document is to identify a framework and possible approaches to allow deployment of QoS routing capabilities with the minimum possible impact to the existing routing infrastructure.

Contents

## 1. Introduction

In this document we describe a set of proposed additions to the OSPF routing protocol (the additions are built on top of OSPF V2) to support Quality-of-Service (QoS) routing in IP. In particular we discuss the metrics required to support QoS, the associated link advertisement mechanisms, the path selection algorithm, as well as aspects of route establishment (pinning and unpinning). Our goals are to define an approach which while achieving the goals of improving performance for QoS flows (likelihood to be routed on a path capable of providing the requested QoS), does so with the least possible impact on the existing OSPF protocol. Given the inherent complexity of QoS routing, achieving this goal obviously implies trading-off ‘‘optimality’’ for ‘‘simplicity’’, but we believe this to be required in order to facilitate deployment of QoS routing capabilities.

### 1.1. Overall Framework

We consider a network (1) that supports both best-effort packets and packets with QoS guarantees. The way in which the network resources are split between the two classes is irrelevant to our proposal, except for the assumption that each QoS capable router in the network is able to dedicate some of its resources to satisfy the requirements of QoS packets. QoS capable routers are also assumed able to identify and advertise the amount of their resources that remain available for additional QoS flows. In addition, we limit ourselves to the case where all the routers involved support the QoS extensions described in this document, i.e., we do not consider the problem of establishing a route in an heterogeneous environment with routers that are QoS-capable and others that are not. Furthermore, in this document we focus on the case of unicast flows, although many of the additions we define are applicable to multicast flows as well.

We assume that a flow with QoS requirements will specify them in some fashion that is accessible to the routing protocol. For example, this could correspond to the arrival of an RSVP [RZB+96] PATH message, whose TSpec is passed to routing together with the destination address. After processing such a request, the routing protocol returns a path that it deems the most suitable given the flow’s requirements. Depending on the scope of the path selection

---------------------------

1. In this document we commit the abuse of notation of calling a ‘‘network’’ the interconnection of routers and networks through which we attempt to acompute a QoS path.

process, this returned path could range from simply identifying the best next hop, i.e., a traditional hop-by-hop routing, to specifying all intermediate nodes to the destination, i.e., a source route. Note that this decision impacts the operation of the path selection algorithm as it translates into different requirements in order to construct and return the appropriate path information. Note also that extension to multicast paths will impact differently a source routed and a hop-by-hop approach.

Once a suitable path has been identified, the flow is assigned to it (pinning) and remains assigned to it until it either releases the path (unpinning) or deems that it has become unsuitable, e.g., because of link failure or unavailability of the necessary resources. Note that resources reservation and/or accounting should help limit the frequency of the latter.

In this document, we focus on the aspect of selecting an appropriate path based on information on link metrics and flow requirements. There are obviously many other aspects that need to be specified in order to define a complete proposal for QoS routing. Issues such as those mentioned above on the scope of the path selection process and when/how paths are pinned and unpinned, must certainly be addressed and they are briefly discussed in this draft during the exposition of the path selection algorithms and then more specifically in Section 3. The discussion of a complete solution to these problems is, however, deferred to [GOW96].

## 1.2. Simplifying Assumptions

In order to achieve our goal of a minimum impact to the existing protocol, we impose certain restrictions on the range of requirements the QoS path selection algorithm needs to deal with directly. Specifically, a policy scheme is used to a priori prune from the network, those portions that would be unsuitable given the requirements of the flow. This limits the ‘‘optimization’’ performed by the path selection to a containable set of parameters, which helps keep complexity at an acceptable level. Specifically, the path selection algorithm will focus on selecting a path that is capable of satisfying the bandwidth requirement of the flow, while at the same time trying to minimize the amount of network resources that need to be allocated to support the flow, i.e., minimize the number of hops used.

This focus on bandwidth is adequate in most instances, but does not fully capture the complete range of potential QoS requirements. For example, a delay-sensitive flow of an interactive application could be put on a path using a satellite link, if that link provided a direct path and had plenty of unused bandwidth. This would clearly not be a desirable choice. Our approach to preventing such poor choices, is to assign delay-sensitive flows to a policy that would eliminate from the network all links with high propagation delay, e.g., satellite links, before invoking the path selection algorithm. In general, each existing policy would present to the path selection algorithm its correspondingly pruned network topology, and the same algorithm would then be used to generate an appropriate path.

Another important aspect in minimizing the impact of QoS routing is to develop a solution that has the smallest possible computing overhead. Additional computations are unavoidable, but it is desirable to keep the total cost of QoS routing at a level comparable to that of traditional routing algorithms. In this document, we describe several alternatives to the path selection algorithm, that represent different trade-offs between simplicity, accuracy, and computational cost. In particular, we specify algorithms that generate exact solutions based either on pre-computations or on-demand computations. We also describe algorithms that allow pre-computations at the cost of some loss in accuracy, but with possibly lower complexity or greater ease of implementation. It should be mentioned, that while several alternative algorithms are described in this document, the same algorithm needs to be used consistently within a given routing domain. This requirement can be relaxed when a source routed approach is used as the responsibility of selecting a QoS path lies with a single entity, the origin of the request, which ensures consistency. Hence, it may then be possible for each router to use a different path selection algorithm. However, in general, the use of a common path selection algorithm is recommended, if not necessary, for proper operation.

The rest of this document is structured as follows. In Section 2, we describe the path computation process and the information that it relies on. In Section 3 we briefly review some issues associated with path management and their implications. As mentioned earlier, detailed discussions on these topics is deferred to [GOW96]. In Section 4, we go over the extensions to OSPF that are needed in order to support the path selection process of Section 2. Finally, several appendices provide details on the different path selection algorithms described in Section 2, and outline several additional work items.

## 2. Path Selection Information and Algorithms

This section describes several path selection algorithms that can be used to generate QoS capable routes based on different trade-offs between accuracy, computational complexity, and ease of implementation. In addition, the section also covers aspects related

DOCKET ALARM

# Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

## Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

## Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

## Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

## API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

**LAW FIRMS**

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

**FINANCIAL INSTITUTIONS**

Litigation and bankruptcy checks for companies and debtors.

**E-DISCOVERY AND LEGAL VENDORS**

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | sales@docketalarm.com | 1-866-77-FASTCASE