Network Working Group
Internet-Draft

John Harper
Anagran Inc
Patrick McGeer
Hewlett Packard Corp
January 2007

Requirements for In-Band QoS Signalling
<draft-harper-inband-signalling-requirements-00.txt>

Status of this Memo

Abstract

This document describes the raionale and requirements for in-band signalling of Quality of Service (QoS) characteristics for the Internet Protocol (IP) There has been extensive work on QoS for IP, leading to the development of RSVP, NSIS and other protocols. New requirements emerging from the use of IP to carry services such as video and voice are leading in turn to additional requirements for QoS signalling. The authors believe that these can best be met by adding in-band signalling to IP, complementing the existing protocols for QoS signalling.

Copyright Notice

1. Introduction

The TCP/IP protocol suite was originally designed to support purely best-effort data transmission. As congestion started to become a practical problem, mechanisms were added in TCP to allow end systems to react to congestion and to slow down their transmission accordingly. As quality-sensitive applications began to use TCP/IP, work was undertaken to provide for explicit signalling of QoS requirements, resulting initially in RSVP and later in the start of work on NSIS. RSVP and NSIS are both out-of-band signalling protocols, i.e. the signalling flow takes place over a separate packet flow from the information itself. This works well for some classes of applications, such as the establishment of MPLS tunnels for traffic engineering.

The TCP/IP protocol suite is increasingly becoming a universal way to move all kinds of information, including those which have previously relied on dedicated analog or digital circuits, such as voice and video. Some of these have more stringent quality requirements than traditional data traffic, requiring a certain minimum bandwidth and maximum delay variance ("jitter") if the user requirements are to be met. Where large numbers of sessions are involved, for example when providing domestic video services, it is difficult to design equipment such that out-of-band protocols can operate quickly enough.

TCP´s ability to react to network congestion, and hence adapt its transmission speed to available capacity, depends heavily on the fact that in traditional data networks, packet loss due to transmission errors is a rare event. It also depends on gradual ramp-up to the available capacity, using the "slow-start" technique. These have both been appropriate for the networks which have predominated until recently, where links are provided by optical or electrical means and where the bandwidth required for an individual TCP connection is no more than a few megabits/second. Current technological trends mean that these assumptions do not necessarily apply any longer. Wireless links are becoming commonplace. These have significant error rates, enough to force TCP into low throughput as it incorrectly reacts to lost packets as though they signal congestion. This is particularly serious when the wireless links have high bandwidth, for example in the case of third and fourth generation mobile systems.

Bandwidths available to all classes of user are steadily increasing. For example, domestic users in many parts of Asia now have access rates of 45 or 100 Mbit/sec. Commercial and industrial users are now often connected via links having instantaneous bandwidth of 100 Mbit/sec or 1 Gbit/sec. TCP slow-start and TCP´s reaction to

occasional packet loss mean that practical achievement of these kinds of bandwidths will often not occur. In order to achieve these bandwidths, especially in the presence of errors, explicit signalling of available bandwidth in the network is required.

For all the above reasons, it is considered desirable to add means to the TCP/IP protocol suite to allow efficient signalling of QoS requirements and availability in close association with the IP traffic itself. While this could in principle be done using out-of-band signalling, the scale of the signalling to be undertaken makes it harder to achieve the necessary responsiveness and performance by this means. Additionally, in high-speed routers and switches, signalling at the envisaged scale must be performed in hardware if the performance is to be achieved. This is simpler if an in-band signalling protocol, with a simple and well-structured format, is used.

In the remainder of this draft, we present specific rationale for an in-band QoS signalling protocol, and set out the requirements that must be achieved by the architecture and the corresponding protocol.

2. QoS Service Goals

2.1 Definition of a Flow

In the remainder of this document, the term "flow" is used to mean a sequence of packets belonging to the same host-to-host association, typically either a TCP connection, or where a connectionless transport protocol is in use, a sequence of packets corresponding to the same information stream. In an IPv4 network, a flow is generally considered be a sequence of packets sharing the same source and destination IP addresses, transport protocol and transport protocol source and destination ports (if applicable). In an IPv6 network, a flow is a sequence of packets sharing the same source and destination IP addresses and the same Flow Identifier. The presence of the Flow Identifier in IPv6 means that inspection of transport layer port

information is not required in order to determine flow membership. However this does depend upon the consistent use of the Flow Identifier. It is also possible to make explicit use of transport layer port information, if IPv6 payload encryption is not in use.

2.2 Types of Flow

IP applications can be broadly divided into two categories, when considering QoS:

- those that will use whatever bandwidth is available, adjusting their rate accordingly. Such applications normally run over TCP (or

other connection-mode transport protocols such as SCTP). Flows having this characteristic are called AR ("Available Rate") flows. Because available capacity in the network is constantly changing, AR flows need to be able to receive periodic updates about the current capacity, so they can change their behavior.

- those that have a predetermined bandwidth requirement, which may be fixed or may vary. If the required bandwidth is not available then the service as seen by the user deteriorates and may become completely unusable. Applications such as voice and streaming video fall into this class. These applications can in turn be divided into three categories, as described below.

GR - Guaranteed Rate service specifies a reserved rate and is intended for those cases where bandwidth must be reserved by the network, even when it is unused. The network should not oversubscribe the allocations for any given preemption level. It requires explicit signalling of reservation and release, which are only used with the Guaranteed Rate service.

MR - Maximum Rate service has no fixed reservation. It specifies the maximum rate the flow might use and makes every attempt to assure no packet loss if the flow remains under this rate. The flow may be variable in rate, not to exceed the specified maximum rate. Flows may be policed or shaped to ensure that this rate is not exceeded. Flows may be dropped if it is known to be impossible to deliver the stated maximum rate, for example dut to congestion. Unlike guaranteed rate traffic, the bandwidth is available subject to traffic statistics; it is not an absolute guarantee. The flow is dropped if no traffic is

seen for a preset period and so no explicit release is required. The service can be used for individual video, voice or streaming media flows where very low loss and/or low delay jitter is required.

VR - Variable Rate service, where part of the rate is guaranteed and part is determined by network capacity (available rate). This is typically the case, when streaming media cannot function below a base rate but can take advantage of additional capacity if it is available. This capability is signaled as a maximum rate plus an available rate. Traffic up to the maximum rate plus the available rate will be supported. The available rate may be changed from time to time as the network capacity changes.

2.3 Other QoS Goals

In addition to the bandwidth-related QoS flow attributes described in section 2.1, there are other requirements associated with the service applied to a particular flow.

PP - Preemption Priority indicates which flows should survive when the network capacity is insufficient to support the required quality of all the flows. It is necessary to support civilian emergency services, military services, and will also have applications in other aspects of networking.

BT - Burst Tolerance specifies how much the actual rate may exceed the stated rate before policing is enforced. This not only applies to rate bursts the user may introduce at the source but is necessary to allow for bunching the network routers may introduce.

DP - Delay Priority permits explicit signalling of the relative importance of flows with respect to limiting delay variation.

CH - Charge Direction specifies who is paying for this flow, the sender or the receiver. This allows IP to provide 800 type services like the PSTN and allows for peering between networks of different sizes with fair charging.

3. Problems with Current QoS Signalling Techniques

3.1 Diffserv and its Limitations

Diffserv provides a Class of Service (COS) marking in 6 bits in the IP header. Only a subset of the 64 possible values has a globally recognised definition. Due to the low range of values, there is no way to have globally unified definitions of the meaning of the codepoints. Also there are too few bits to specify explicitly a rate for either guaranteed rate flows or for available rate feedback. This limits the utility of Diffserv to a few delay categories and some local link-by-link agreements. As we move into video and streaming media as well as improving TCP performance, Diffserv provides insufficient flexibility.

3.2 IntServ and its Limitations

The current proposals for IntServ type QoS support (as opposed to CoS support via Diffserv) revolve around a round trip call setup request using complex protocols like RSVP. These protocols require more processing than can be done in hardware and are thus obliged to operate in relativey slow software processes. This limits the total end-to-end call processing that can be made to either very large flows or to composite (VPN) flows since processor speeds are insufficient for processing all IP flows.

What is desirable is a protocol that permits the router to process QoS requests for each individual flow, including parameters such as bandwidth, delay priority, preemption priority, burst tolerance, and

charging direction. Many common flows like file transfers may not require all this information. However, increasingly flows are becoming voice, video, gaming, or streaming media where such requirements do apply.

Also, as the IP becomes the predominant protocol for the provision of voice and emergency communications, the need for call rejection and preemption priority become important and may even be life and death issues. These capabilities are therefore required as part of the IP protocol, for both TCP and UDP.

3.3 Issues Regarding TCP

TCP slow-start has worked well over the past 20 years when the individual flows were generally in the kilobits/second. However, as

wideband corporate access and broadband residential access have proliferated, the desired flow rate has increased into the megabits/second up to several gigabits/second. TCP was not designed for these rates at any significant distance.

TCP depends on detecting packet loss to adjust its rate; any packet loss at high rates over long distances creates a major slow down of the flow. Even at normal cross-country distances, maximum TCP throughput is limited to megabits/second rather than gigabits/second. The performance of TCP when operating over satellite links is poor enough that TCP "spoofing" devices are frequently used at either end of any satellite link. However, this will not work when the data is encrypted as in IPv6, in IPv4 with IPSEC or in either with encryptors inserted.

Another problem with using packet loss as a rate control mechanism is that the typical method of dealing with congestion is to discard random packets, even if the flow in question is not yet up to the speed of other flows. This is usually done using Random Early Detection (RED) or Weighted Random Early Detection (WRED). This procedure slows down the TCP rate so that it does not get up to the maximum network speed as fast as possible. This problem has been magnified as the network speeds have increased such that an average 120 K byte web page transfer over a 10 Mbps access line can take many seconds rather than the fraction of a second actually required.

The TCP situation is improved by marking packets for congestion, using Explicit Congestion Notification (ECN) before discarding becomes necessary. ECN avoids discarding the packet, but the search for the rate the network can support is still a binary search that leads to significant loss of throughput and does not improve the slow-start problem. Marking packets in this way, also presumes that the network has sufficient storage to signal congestion well before

the problem becomes critical. This in turn leads to either additional network delays due to long queue traversals, or exacerbates the existing problem of lower network utilization.

When the concept of marking packets is coupled with the feedback of the maximum rate the network could support, it is feasible for the TCP source to avoid slow-start and achieve extremely high-speed throughput. Instead of marking or discarding packets when congestion

occurs, the network can inform all TCP sources as soon as possible of the rate that they can operate at safely. As conditions change, additional feedback is required. This concept was simulated and tested extensively in ATM systems. It was found to substantially decrease the time to transfer a web page and significantly decrease the buffering required in the network.

3.4 Considerations for Streaming Protocols

While many applications use TCP, there are others which do not require the benefits it brings with regard to error recovery, reordering and flow control. Such applications typically use UDP as a connectionless transport protocol. In particular, real-time streaming applications such as video and voice operate in this way. These applications are not, in general, able to adjust their transmission rate in response to the capacity of the network. For example, an uncompressed voice flow using G.711 encoding requires a bandwidth of 64 kbit/sec for the voice payload. Similarly, real-time compressed video typically requires a bandwidth in the range 2-6 Mbit/sec, depending on the encoding and the content. A given flow will simply not work if the available bandwidth turns out to be less than this. (There are approaches to dealing with this, for example by switching to a lower-rate codec with lower quality, but they require an additional management layer and also require explicit knowledge of the available bandwidth, unlike TCP which adjusts continuously). Hence, if the total bandwidth available is enough to support say 100 such flows, and another one presents itself, it is better to block the new flow, than to accept it and downgrade the quality of the service provided to all 101 users. The capability to control admission in this way is referred to as Call Admission Control (CAC).

RSVP was designed to perform admission control, but the limited practical experience with RSVP as a generic admission control protocol shows that it does not scale to handle very large numbers of individual flows. It is adequate for handling aggregated flows (for example, VPN connections through a service provider network) but not for handling individual flows.

3.5 Protocol Complexity and Hardware Processing

A high-speed router, operating at link speeds of 1 Gbit/sec or

higher, will handle a very large number of flows and flow-setup operations. If many of those flows include explicit QoS signalling, the handling of this signalling needs to be performed at very high speed. In a practical implementation, this means that it must take place in the data plane and be performed by the same system elements that perform packet forwarding operations, i.e. hardware logic implemented in an ASIC or programmable logic, or in code executed in special-purpose Network Processors (NPUs). This logic or code is highly optimised and is not amenable to the kinds of operations typically performed in general-purpose CPUs by software. In considerations of protocol design, this means that the QoS signalling protocol should have a simple fixed structure. The very flexible structures used in existing QoS signalling protocols, such as type-length-value encoding supporting a multiplicity of optional features, cannot be implemented in hardware or NPU-based systems.

4. Implications for Signalling Protocol Design

4.1 Benefits of In-band Signalling

Signalling of QoS requirements can in principle be effected using either out-of-band signalling, like RSVP, or in-band signalling. To date, there has been no implementation of RSVP which can perform large numbers of flow-setup operations at a high rate, which suggests that such an implementation would be difficult.

In-band signalling, using a simple protocol, has the following benefits in addition to those described above.

1. In-band signalling is guaranteed to follow the same path through the network as the flow on whose behalf it is signalling. This is very difficult to achieve for an out-of-band protocol, taking into account realities of network design such as equal cost multipath routes.

2. Since in-band signalling is carried along with the payload of the flow, it is unaffected by en-route changes to network addressing such as NAT. In principle an out-of-band protocol can deal with this, but it is a complex problem which has been under study in the IETF (MidCom) for several years.

3. Accurate detection of congestion in the network requires frequent updates about the available capacity. A simple protocol carried along with the payload can provide this through an efficient data-plane implementation. Given the difficulty of scaling current out-of-band protocols to handle large numbers of flows, it would be even harder to provide frequent real-time updates about the state of the network.

4.2 Requirements for an In-band Signalling Protocol

From the above considerations, the following requirements apply to an in-band protocol for QoS signalling.

1. The protocol MUST be carried within packets which form part of the flow for which QoS information is being conveyed. This is inherent in the nature of an in-band protocol.

2. The protocol MUST be capable of operating with both IPv4 and IPv6, although variations may be required to make this possible.

3. The protocol MUST provide a means for an AR flow to signal its requested bandwidth, and to receive information from the network about the bandwidth actually available.

4. The protocol MUST provide a means for a GR, MR or VR flow to signal its requested bandwidth.

5. The protocol MUST provide a means to signal other QoS information, specifically Burst Tolerance, Preemption Priority, Delay Priority, and Charging Information.

6. The protocol MUST provide a means for a flow to periodically determine the currently available bandwidth in the network.

7. The protocol MUST provide a means for GR flows to perform explicit bandwidth reservation and release, in a confirmed and reliable way.

8. The protocol MUST define an encoding which lends itself readily to implementation in hardware, programmable logic or high-performance dedicated network processors.

5. Security Considerations

A QoS signalling protocol is capable of reserving network resources, or of providing information to a host which will lead it to make use of network resources. The protocol which is defined in support of the requirements decsribed in this document needs to address how such resources can be protected in an appropriate fashion, either through elements of the protocol itself, or through associated configuration of the systems which implement it, or both.

6. Editors´s Addresses

John Harper

Anagran Inc
2055 Woodside Road

Redwood City, CA 94065
USA

Phone: +1 650 587 8276
EMail: john@anagran.com

Patrick McGeer
Hewlett-PAckard Laboratories
1501 Page Mill Road
Palo Alto, CA 94304
USA

Email: rick.mcgeer@hp.com

This Internet-Draft will expire on July 19th, 2007.

7. Full copyright statement