TSVWG
Internet Draft
draft-briscoe-tsvwg-cl-architecture-04.txt
Expires: April 2007

B. Briscoe
P. Eardley
D. Songhurst
BT

F. Le Faucheur
A. Charny
Cisco Systems, Inc

J. Babiarz
K. Chan
S. Dudley
Nortel

G. Karagiannis
University of Twente / Ericsson

A. Bader
L. Westberg
Ericsson

25 October, 2006

An edge-to-edge Deployment Model for Pre-Congestion Notification: Admission Control over a DiffServ Region
draft-briscoe-tsvwg-cl-architecture-04.txt

Status of this Memo

By submitting this Internet-Draft, each author represents that any applicable patent or other IPR claims of which he or she is aware have been or will be disclosed, and any of which he or she becomes aware will be disclosed, in accordance with Section 6 of BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress".

The list of current Internet-Drafts can be accessed at
http://www.ietf.org/1id-abstracts.html

The list of Internet-Draft Shadow Directories can be accessed at http://www.ietf.org/shadow.html

This Internet-Draft will expire on September 6, 2006.

Copyright Notice

Copyright (C) The Internet Society (2006). All Rights Reserved.

Abstract

This document describes a deployment model for pre-congestion notification (PCN) operating in a large DiffServ-based region of the Internet. PCN-based admission control protects the quality of service of existing flows in normal circumstances, whilst if necessary (eg after a large failure) pre-emption of some flows preserves the quality of service of the remaining flows. Each link has a configured-admission-rate and a configured-pre-emption-rate, and a router marks packets that exceed these rates. Hence routers give an early warning of their own potential congestion, before packets need to be dropped. Gateways around the edges of the PCN-region convert measurements of packet rates and their markings into decisions about whether to admit new flows, and (if necessary) into the rate of excess traffic that should be pre-empted. Per-flow admission states are kept at the gateways only, while the PCN markers that are required for all routers operate on the aggregate traffic - hence there is no scalability impact on interior routers.

Authors' Note (TO BE DELETED BY THE RFC EDITOR UPON PUBLICATION)

This document is posted as an Internet-Draft with the intention of eventually becoming an INFORMATIONAL RFC.

Table of Contents

1. Introduction

1.1. Summary

This document describes a deployment model to achieve an end-to-end Controlled Load service by using (within a large region of the Internet) DiffServ and edge-to-edge distributed measurement-based admission control and flow pre-emption. Controlled load service is a quality of service (QoS) closely approximating the QoS that the same flow would receive from a lightly loaded network element [RFC2211]. Controlled Load (CL) is useful for inelastic flows such as those for real-time media.

In line with the "IntServ over DiffServ" framework defined in [RFC2998], the CL service is supported end-to-end and RSVP signalling [RFC2205] is used end-to-end, over an edge-to-edge DiffServ region. We call the DiffServ region the "CL-region".

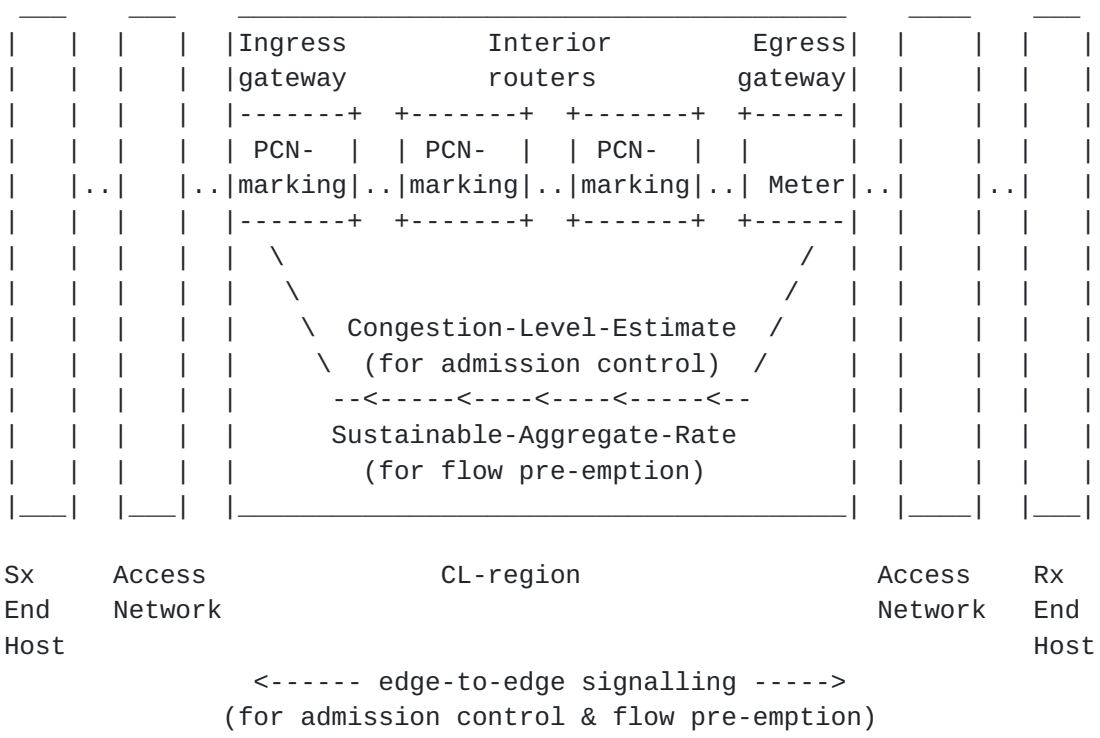


Figure 1: Overall QoS architecture (NB terminology explained later)

Figure 1 shows an example of an overall QoS architecture, where the two access networks are connected by a CL-region. Another possibility is that there are several CL-regions between the access networks - each would operate the Pre-Congestion Notification mechanisms separately. The document assumes RSVP as the end-to-end QoS signalling protocol. However, the RSVP signalling may itself be originated or terminated by proxies still closer to the edge of the network, such as home hubs or the like, triggered in turn by application layer signalling. [RFC2998] and our approach are compared further in Section 6.2.

Flows must enter and leave the CL-region through its ingress and egress gateways, and they need traffic descriptors that are policed by the ingress gateway (NB the policing function is out of this document's scope). The overall CL-traffic between two border routers is called a "CL-region-aggregate".

The document introduces a mechanism for flow admission control: should a new flow be admitted into a specific CL-region-aggregate? Admission control protects the QoS of existing CL-flows in normal circumstances. In abnormal circumstances, for instance a disaster affecting multiple interior routers, then the QoS on existing CL microflows may degrade even if care was exercised when admitting those microflows before those circumstances. Therefore we also propose a mechanism for flow pre-emption: how much traffic, in a specific CL-region-aggregate, should be pre-empted in order to preserve the QoS of the remaining CL-flows? Flow pre-emption also restores QoS to lower priority traffic.

As a fundamental building block to enable these two mechanisms, each link of the CL-region is associated with a configured-admission-rate and configured-pre-emption-rate; the former is usually significantly larger than the latter. If traffic in a specific DiffServ class ("CL-traffic") on the link exceeds these rates then packets are marked with "Admission Marking" or "Pre-emption Marking". The algorithms that determine the number of packets marked are outlined in Section 3 and detailed in [PCN]. PCN marking (Pre-Congestion Notification) builds on the concepts of RFC 3168, "The addition of Explicit Congestion Notification to IP" (which is briefly summarised in Appendix A).

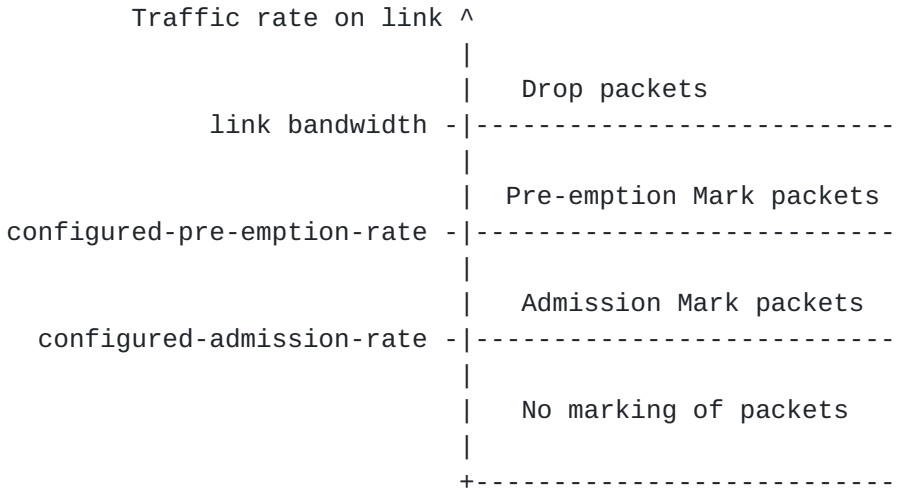


Figure 2: Packet Marking by Routers

Gateways of the CL-region make measurements of packet rates and their PCN markings and convert them into decisions about whether to admit new flows, and (if necessary) into the rate of excess traffic that should be pre-empted. These mechanisms are detailed in Section 3 and briefly outlined in the next few paragraphs.

The admission control mechanism for a new flow entering the network at ingress gateway G0 and leaving it at egress gateway G1 relies on feedback from the egress gateway G1 about the existing CL-region-aggregate between G0 and G1. This feedback is generated as follows. All routers meter the rate of the CL-traffic on their outgoing links and mark the packets with the Admission Mark if the configured-admission-rate is exceeded. Egress gateway G1 measures the Admission Marks for each of its CL-region-aggregates separately. If the fraction of traffic on a CL-region-aggregate that is Admission Marked exceeds some threshold, no further flows should be admitted into this CL-region-aggregate. Because sources vary their data rates (amongst other reasons) the rate of the CL-traffic on a link may fluctuate above and below the configured-admission-rate. Hence to get more stable information, the egress gateway measures the fraction as a moving average, called the Congestion-Level-Estimate. This is signalled from the egress G1 to the ingress G0, to enable the ingress to block new flows.

Admission control seems most useful for DiffServ's Controlled load service. In order to support CL traffic we would expect PCN to supplement the existing scheduling behaviour Expedited Forwarding

(EF). Since PCN gives an "early warning" of potential congestion (hence "pre-congestion notification"), admission control can kick in before there is any significant build up of packets in routers - which is exactly the performance required for CL. However, PCN is not only intended to supplement EF. PCN is specified (in [PCN]) as a building block which can supplement the scheduling behaviour of other PHBs.

The function to pre-empt flows (or allow the potential to pre-empt them) relies on feedback from the egress gateway about the CL-region-aggregates. This feedback is generated as follows. All routers meter the rate of the CL-traffic on their outgoing links, and if the rate is in excess of the configured-pre-emption-rate then packets amounting to the excess rate are Pre-emption Marked. If the egress gateway G1 sees a Pre-emption Marked packet then it measures, for this CL-region-aggregate, the rate of all received packets that aren't Pre-emption Marked. This is the rate of CL-traffic that the network can actually support from G0 to G1, and we thus call it the Sustainable-Aggregate-Rate. The ingress gateway G0 compares the Sustainable-Aggregate-Rate with the rate that it is sending towards G1, and hence determines the required traffic rate reduction. The document assumes flow pre-emption as the way of reacting to this information, ie stopping sufficient flows to reduce the rate to the Sustainable-Aggregate-Rate. However, this isn't mandated, for instance policy or regulation may prevent pre-emption of some flows - such considerations are out of scope of this document.

1.2. Key benefits

We believe that the mechanisms described in this document are simple, scalable, and robust because:

- Per flow state is only required at the ingress gateways to prevent non-admitted CL traffic from entering the PCN-region. Other network entities are not aware of individual flows.

- For each of its links a router has Admission Marking and Pre-emption Marking behaviours. These markers operate on the overall CL traffic of the respective link. Therefore, there are no scalability concerns.

- The information of these measurements is implicitly signalled to the egress gateways by the marks in the packet headers. No protocol actions (explicit messages) are required.

- The egress gateways make separate measurements for each ingress gateway of packets. Each meter operates on the overall CL traffic of a particular CL-region-aggregate. Therefore, there are no scalability concerns as long as the number of ingress gateways is not overwhelmingly large.

- Feedback signalling is required between all pairs of ingress and egress gateways and the signalled information is on the basis of the corresponding CL-region-aggregate, i.e. it is also unaware of individual flows.

- The configured-admission-rates can be chosen small enough that admitted traffic can still be carried after a rerouting in most failure cases. This is an important feature as QoS violations in core networks due to link failures are more likely than QoS violations due to increased traffic volume.

- The admitted load is controlled dynamically. Therefore it adapts as the traffic matrix changes, and also if the network topology changes (eg after a link failure). Hence an operator can be less conservative when deploying network capacity, and less accurate in their prediction of the traffic matrix. Also, controlling the load using statically provisioned capacity per ingress (regardless of the egress of a flow), as is typical in the DiffServ architecture [RFC2475], can lead to focussed overload: many flows happen to focus on a particular link and then all flows through the congested link fail catastrophically (Section 6.2).

- The pre-emption function complements admission control. It allows the network to recover from sudden unexpected surges of CL-traffic on some links, thus restoring QoS to the remaining flows. Such scenarios are very unlikely but not impossible. They can be caused by large network failures that redirect lots of admitted CL-traffic to other links, or by malfunction of the measurement-based admission control in the presence of admitted flows that send for a while with an atypically low rate and increase their rates in a correlated way.

1.3. Terminology

EDITOR'S NOTE: Terminology in this document is (hopefully) consistent with that in [PCN]. However, it may not be consistent with the

terminology in other PCN-related documents. The PCN Working Group (if formed) will need to agree a single set of terminology.

This terminology is copied from the pre-congestion notification marking draft [PCN]:

- Pre-Congestion Notification (PCN): two new algorithms that determine when a PCN-enabled router Admission Marks and Pre-emption Marks a packet, depending on the traffic level.

- Admission Marking condition: the traffic level is such that the router Admission Marks packets. The router provides an "early warning" that the load is nearing the engineered admission control capacity, before there is any significant build-up of CL packets in the queue.

- Pre-emption Marking condition: the traffic level is such that the router Pre-emption Marks packets. The router warns explicitly that pre-emption may be needed.

- Configured-admission-rate: the reference rate used by the admission marking algorithm in a PCN-enabled router.

- Configured-pre-emption-rate - the reference rate used by the pre-emption marking algorithm in a PCN-enabled router.

The following terms are defined here:

- Ingress gateway: router at an ingress to the CL-region. A CL-region may have several ingress gateways.

- Egress gateway: router at an egress from the CL-region. A CL-region may have several egress gateways.

- Interior router: a router which is part of the CL-region, but isn't an ingress or egress gateway.

- CL-region: A region of the Internet in which all traffic enters/leaves through an ingress/egress gateway and all routers run Pre-Congestion Notification marking. A CL-region is a DiffServ region (a DiffServ region is either a single DiffServ domain or set of contiguous DiffServ domains), but note that the CL-region does not use the traffic conditioning agreements (TCAs) of the (informational) DiffServ architecture.

- o CL-region-aggregate: all the microflows between a specific pair of ingress and egress gateways. Note there is no field in the flow packet headers that uniquely identifies the aggregate.

- o Congestion-Level-Estimate: the number of bits in CL packets that are admission marked (or pre-emption marked), divided by the number of bits in all CL packets. It is calculated as an exponentially weighted moving average. It is calculated by an egress gateway for the CL packets from a particular ingress gateway, i.e. there is a Congestion-Level-Estimate for each CL-region-aggregate.

- o Sustainable-Aggregate-Rate: the rate of traffic that the network can actually support for a specific CL-region-aggregate. So it is measured by an egress gateway for the CL packets from a particular ingress gateway.

- o Ingress-Aggregate-Rate: the rate of traffic that is being sent on a specific CL-region-aggregate. So it is measured by an ingress gateway for the CL packets sent towards a particular egress gateway.

1.4. Existing terminology

This is a placeholder for useful terminology that is defined elsewhere.

1.5. Standardisation requirements

The framework described in this document has two new standardisation requirements:

- o new Pre-Congestion Notification for Admission Marking and Pre-emption Marking are required, as detailed in [PCN].

- o the end-to-end signalling protocol needs to be modified to carry the Congestion-Level-Estimate report (for admission control) and the Sustainable-Aggregate-Rate (for flow pre-emption). With our assumption of RSVP (Section 2.2) as the end-to-end signalling protocol, it means that extensions to RSVP are required, as detailed in [RSVP-PCN], for example to carry the Congestion-Level-Estimate and Sustainable-Aggregate-Rate information from egress gateway to ingress gateway.

- We are discussing what to standardise about the gateway's behaviour.

Other than these things, the arrangement uses existing IETF protocols throughout, although not in their usual architecture.

1.6. Structure of rest of the document

Section 2 describes some key aspects of the deployment model: our goals and assumptions. Section 3 describes the deployment model, whilst Section 4 summarises the required changes to the various routers in the CL-region. Section 5 outlines some limitations of PCN that we've identified in this deployment model; it also discusses some potential solutions, and other possible extensions. Section 6 provides some comparison with existing QoS mechanisms.

2. Key aspects of the deployment model

EDITOR'S NOTE: The material in Section 2 will eventually disappear, as it will be covered by the problem statement of the PCN Working Group (if formed).

In this section we discuss the key aspects of the deployment model:

- At a high level, our key goals, i.e. the functionality that we want to achieve

- The assumptions that we're prepared to make

2.1. Key goals

The deployment model achieves an end-to-end controlled load (CL) service where a segment of the end-to-end path is an edge-to-edge Pre-Congestion Notification region. CL is a quality of service (QoS) closely approximating the QoS that the same flow would receive from a lightly loaded network element [RFC2211]. It is useful for inelastic flows such as those for real-time media.

- The CL service should be achieved despite varying load levels of other sorts of traffic, which may or may not be rate adaptive (i.e. responsive to packet drops or ECN marks).

- The CL service should be supported for a variety of possible CL sources: Constant Bit Rate (CBR), Variable Bit Rate (VBR) and voice with silence suppression. VBR is the most challenging to support.

- After a localised failure in the interior of the CL-region causing heavy congestion, the CL service should recover gracefully by pre-empting (dropping) some of the admitted CL microflows, whilst preserving as many of them as possible with their full CL QoS.

- It needs to be possible to complete flow pre-emption within 1-2 seconds. Operators will have varying requirements but, at least for voice, it has been estimated that after a few seconds then many affected users will start to hang up, making the flow pre-emption mechanism redundant and possibly even counter-productive. Until flow pre-emption kicks in, other applications using CL (e.g. video) and lower priority traffic (e.g. Assured Forwarding (AF)) could be receiving reduced service. Therefore an even faster flow pre-emption mechanism would be desirable (even if, in practice, operators have to add a deliberate pause to ride out a transient while the natural rate of call tear down or lower layer protection mechanisms kick in).

- The CL service should support emergency services ([EMERG-RQTS], [EMERG-TEL]) as well as the Assured Service which is the IP implementation of the existing ITU-T/NATO/DoD telephone system architecture known as Multi-Level Pre-emption and Precedence [ITU.MLPP.1990] [ANSI.MLPP.Spec][ANSI.MLPP.Supplement], or MLPP. In particular, this involves admitting new flows that are part of high priority sessions even when admission control would reject new routine flows. Similarly, when having to choose which flows to pre-empt, this involves taking into account the priorities and properties of the sessions that flows are part of.

2.2. Key assumptions

The framework does not try to deliver the above functionality in all scenarios. We make the following assumptions about the type of scenario to be solved.

- Edge-to-edge: all the routers in the CL-region are upgraded with Pre-Congestion Notification, and all the ingress and egress gateways are upgraded to perform the measurement-based admission control and flow pre-emption. Note that although the upgrades required are edge-to-edge, the CL service is provided end-to-end.

- Additional load: we assume that any additional load offered within the reaction time of the admission control mechanism doesn't move the CL-region directly from no congestion to overload. So it assumes there will always be an intermediate stage where some CL packets are Admission Marked, but they are still delivered without significant QoS degradation. We believe this is valid for core and backbone networks with typical call arrival patterns (given the reaction time is little more than one round trip time across the CL-region), but is unlikely to be valid in access networks where the granularity of an individual call becomes significant.

- Aggregation: we assume that in normal operations, there are many CL microflows within the CL-region, typically at least hundreds between any pair of ingress and egress gateways. The implication is that the solution is targeted at core and backbone networks and possibly parts of large access networks.

- Trust: we assume that there is trust between all the routers in the CL-region. For example, this trust model is satisfied if one operator runs the whole of the CL-region. But we make no such assumptions about the end hosts, i.e. depending on the scenario they may be trusted or untrusted by the CL-region.

- Signalling: we assume that the end-to-end signalling protocol is RSVP. Section 3 describes how the CL-region fits into such an end-to-end QoS scenario, whilst [RSVP-PCN] describes the extensions to RSVP that are required.

- Separation: we assume that all routers within the CL-region are upgraded with the CL mechanism, so the requirements of [Floyd] are met because the CL-region is an enclosed environment. Also, an operator separates CL-traffic in the CL-region from outside traffic by administrative configuration of the ring of gateways around the region. Within the CL-region we assume that the CL-traffic is separated from non-CL traffic.

- Routing: we assume that all packets between a pair of ingress and egress gateways follow the same path, or that they follow different paths but that the load balancing scheme is tuned in the CL-region to distribute load such that the different paths always receive comparable relative load. This ensures that the Congestion-Level-Estimate used in the admission control procedure (and which is computed taking into account packets travelling on all the paths) approximately reflects the status of the actual path that will be followed by the new microflow's packets.

We are investigating ways of loosening the restrictions set by some of these assumptions, for instance:

- Trust: to allow the CL-region to span multiple, non-trusting operators, using the technique of [Re-PCN] as mentioned in Section 5.7.2.

- Signalling: we believe that the solution could operate with another signalling protocol, such as the one produced by the NSIS working group. It could also work with application level signalling as suggested in [RT-ECN].

- Additional load: we believe that the assumption is valid for core and backbone networks, with an appropriate margin between the configured-admission-rate and the capacity for CL traffic. However, in principle a burst of admission requests can occur in a short time. We expect this to be a rare event under normal conditions, but it could happen e.g. due to a 'flash crowd'. If it does, then more flows may be admitted than should be, triggering the pre-emption mechanism. There are various ways an operator might try to alleviate this issue, which are discussed in the 'Flash crowds' section 5.5 later.

- Separation: the assumption that CL traffic is separated from non-CL traffic implies that the CL traffic has its own PHB, not shared with other traffic. We are looking at whether it could share Expedited Forwarding's PHB, but supplemented with Pre-Congestion Notification. If this is possible, other PHBs (like Assured Forwarding) could be supplemented with the same new behaviours. This is similar to how RFC3168 ECN was defined to supplement any PHB.

- Routing: we are looking in greater detail at the solution in the presence of Equal Cost Multi-Path routing and at suitable enhancements. See also the 'ECMP' section 5.1 later.

3. Deployment model

3.1. Admission control

In this section we describe the admission control mechanism. We discuss the three pieces of the solution and then give an example of how they fit together in a use case:

- o the new Pre-Congestion Notification for Admission Marking used by all routers in the CL-region
- o how the measurements made support our admission control mechanism
- o how the edge to edge mechanism fits into the end to end RSVP signalling

3.1.1. Pre-Congestion Notification for Admission Marking

This is discussed in [PCN]. Here we only give a brief outline.

To support our admission control mechanism, each router in the CL-region runs an algorithm to determine whether to Admission Mark the packet. The algorithm measures the aggregate CL traffic on the link and ensures that packets are admission marked before the actual queue builds up, but when it is in danger of doing so soon; the probability of admission marking increases with the danger. The algorithm's main parameter is the configured-admission-rate, which is set lower than the link speed, perhaps considerably so. Admission marked packets indicate that the CL traffic rate is reaching the configured-admission-rate and so act as an "early warning" that the engineered capacity is nearly reached. Therefore they indicate that requests to admit prospective new CL flows may need to be refused.

3.1.2. Measurements to support admission control

To support our admission control mechanism the egress measures the Congestion-Level-Estimate for traffic from each remote ingress gateway, i.e. per CL-region-aggregate. The Congestion-Level-Estimate is the number of bits in CL packets that are admission marked or pre-emption marked, divided by the number of bits in all CL packets. It is calculated as an exponentially weighted moving average. It is calculated by an egress gateway separately for the CL packets from each particular ingress gateway.

Why are pre-emption marked packets included in the Congestion-Level-Estimate? Pre-emption marking over-writes admission marking, i.e. a packet cannot be both admission and pre-emption marked. So if pre-emption marked packets weren't counted we would have the anomaly that as the traffic rate grew above the configured-pre-emption-rate, the Congestion-Level-Estimate would fall. If a particular encoding scheme is chosen where a packet can be both admission and pre-emption marked (such as Alternative 4 in Appendix C of [PCN]), then this is not necessary.

This Congestion-Level-Estimate provides an estimate of how near the links on the path inside the CL-region are getting to the configured-admission-rate. Note that the metering is done separately per ingress gateway, because there may be sufficient capacity on all the routers on the path between one ingress gateway and a particular egress, but not from a second ingress to that same egress gateway.

3.1.3. How edge-to-edge admission control supports end-to-end QoS signalling

Consider a scenario that consists of two end hosts, each connected to their own access networks, which are linked by the CL-region. A source tries to set up a new CL microflow by sending an RSVP PATH message, and the receiving end host replies with an RSVP RESV message. Outside the CL-region some other method, for instance IntServ, is used to provide QoS. From the perspective of RSVP the CL-region is a single hop, so the RSVP PATH and RESV messages are processed by the ingress and egress gateways but are carried transparently across all the interior routers; hence, the ingress and egress gateways hold per microflow state, whilst no per microflow state is kept by the interior routers. So far this is as in IntServ over DiffServ [RFC2998]. However, in order to support our admission control mechanism, the egress gateway adds to the RESV message an opaque object which states the current Congestion-Level-Estimate for the relevant CL-region-aggregate. Details of the corresponding RSVP extensions are described in [RSVP-PCN].

3.1.4. Use case

To see how the three pieces of the solution fit together, we imagine a scenario where some microflows are already in place between a given pair of ingress and egress gateways, but the traffic load is such that no packets from these flows are admission marked as they travel across the CL-region. A source wanting to start a new CL microflow sends an RSVP PATH message. The egress gateway adds an object to the RESV message with the Congestion-Level-Estimate, which is zero. The ingress gateway sees this and consequently admits the new flow. It

then forwards the RSVP RESV message upstream towards the source end host. Hence, assuming there's sufficient capacity in the access networks, the new microflow is admitted end-to-end.

The source now sends CL packets, which arrive at the ingress gateway. The ingress uses a five-tuple filter to identify that the packets are part of a previously admitted CL microflow, and it also polices the microflow to ensure it remains within its traffic profile. (The ingress has learnt the required information from the RSVP messages.) When forwarding a packet belonging to an admitted microflow, the ingress sets the packet's DSCP and ECN fields to the appropriate values configured for the CL region. The CL packet now travels across the CL-region, getting admission marked if necessary.

Next, we imagine the same scenario but at a later time when load is higher at one (or more) of the interior routers, which start to Admission Mark CL packets, because their load on the outgoing link is nearing the configured-admission-rate. The next time a source tries to set up a CL microflow, the ingress gateway learns (from the egress) the relevant Congestion-Level-Estimate. If it is greater than some CLE-threshold value then the ingress refuses the request, otherwise it is accepted. The ingress gateway could also take into account attributes of the RSVP reservation (such as for example the RSVP pre-emption priority of [RSVP-PREEMPTION] or the RSVP admission priority of [RSVP-EMERGENCY]) as well as information provided by a policy decision point in order to make a more sophisticated admission decision. This way, flow admission can help emergency/military calls by taking into account the corresponding priorities (as conveyed in RSVP policy elements) when deciding to admit or reject a new reservation. Use of RSVP for the support of emergency/military applications is discussed in further detail in [RFC4542] and [RSVP-EMERGENCY].

It is also possible for an egress gateway to get a RSVP RESV message and not know what the Congestion-Level-Estimate is. For example, if there are no CL microflows at present between the relevant ingress and egress gateways. In this case the egress requests the ingress to send probe packets, from which it can initialise its meter. RSVP Extensions for such a request to send probe data can be found in [RSVP-PCN].

3.2. Flow pre-emption

In this section we describe the flow pre-emption mechanism. We discuss the two parts of the solution and then give an example of how they fit together in a use case:

- How an ingress gateway is triggered to test whether flow pre-emption may be needed

- How an ingress gateway determines the right amount of CL traffic to drop

The mechanism is defined in [PCN] and [RSVP-PCN].

Two subsequent steps could be:

- Choose which flows to shed, influenced by their priority and other policy information

- Tear down the reservations for the chosen flows

We provide some hints about these latter two steps in Section 3.2.3, but don't try to provide full guidance as it greatly depends on the particular detailed operational situation.

An essential QoS issue in core and backbone networks is being able to cope with failures of routers and links. The consequent re-routing can cause severe congestion on some links and hence degrade the QoS experienced by on-going microflows and other, lower priority traffic. Even when the network is engineered to sustain a single link failure, multiple link failures (e.g. due to a fibre cut, router failure or a natural disaster) can cause violation of capacity constraints and resulting QoS failures. Our solution uses rate-based flow pre-emption, so that sufficient of the previously admitted CL microflows are dropped to ensure that the remaining ones again receive QoS commensurate with the CL service and at least some QoS is quickly restored to other traffic classes.

3.2.1. Alerting an ingress gateway that flow pre-emption may be needed

Alerting an ingress gateway that flow pre-emption may be needed is a two stage process: a router in the CL-region alerts an egress gateway that flow pre-emption may be needed; in turn the egress gateway alerts the relevant ingress gateway. Every router in the CL-region has the ability to alert egress gateways, which may be done either explicitly or implicitly:

- Explicit - the router per-hop behaviour is supplemented with a new Pre-emption Marking behaviour, which is outlined below. Reception of such a packet by the egress gateway alerts it that pre-emption may be needed.

- Implicit - the router behaviour is unchanged from the Admission Marking behaviour described earlier. The egress gateway treats a Congestion-Level-Estimate of (almost) 100% as an implicit alert that pre-emption may be required. ('Almost' because the Congestion-Level-Estimate is a moving average, so can never reach exactly 100%.)

To support explicit pre-emption alerting, each router in the CL-region runs an algorithm to determine whether to Pre-emption Mark the packet. The algorithm measures the aggregate CL traffic and ensures that packets are pre-emption marked before the actual queue builds up. The algorithm's main parameter is the configured-pre-emption-rate, which is set lower than the link speed (but higher than the configured-admission-rate). Thus pre-emption marked packets indicate that the CL traffic rate is reaching the configured-pre-emption-rate and so act as an "early warning" that the engineered capacity is nearly reached. Therefore they indicate that it may be advisable to pre-empt some of the existing CL flows in order to preserve the QoS of the others.

Note that the pre-emption marking algorithm doesn't measure the packets that are already Pre-emption Marked. This ensures that in a scenario with several links that are above their configured-pre-emption-rate, then at the egress gateway the rate of packets excluding Pre-emption Marked ones truly does represent the Sustainable-Aggregate-Rate(see below for explanation).

Note that the explicit mechanism only makes sense if all the routers in the CL-region have the functionality so that the egress gateways can rely on the explicit mechanism. Otherwise there is the danger that the traffic happens to focus on a router without it, and egress gateways then have also to watch for implicit pre-emption alerts.

When one or more packets in a CL-region-aggregate alert the egress gateway of the need for flow pre-emption, whether explicitly or implicitly, the egress puts that CL-region-aggregate into the Pre-emption Alert state. For each CL-region-aggregate in alert state it measures the rate of traffic at the egress gateway (i.e. the traffic rate of the appropriate CL-region-aggregate) and reports this to the relevant ingress gateway. The steps are:

- Determine the relevant ingress gateway - for the explicit case the egress gateway examines the pre-emption marked packet and uses the state installed at the time of admission to determine which ingress gateway the packet came from. For the implicit case the egress gateway has already determined this information, because the Congestion-Level-Estimate is calculated per ingress gateway.

- Measure the traffic rate of CL packets - as soon as the egress gateway is alerted (whether explicitly or implicitly) it measures the rate of CL traffic from this ingress gateway (i.e. for this CL-region-aggregate). Note that pre-emption marked packets are excluded from that measurement. It should make its measurement quickly and accurately, but exactly how is up to the implementation.

- Alert the ingress gateway - the egress gateway then immediately alerts the relevant ingress gateway about the fact that flow pre-emption may be required. This Alert message also includes the measured Sustainable-Aggregate-Rate, i.e. the rate of CL-traffic received from this ingress gateway. The Alert message is sent using reliable delivery. Procedures for the support of such an Alert using RSVP are defined in [RSVP-PCN].

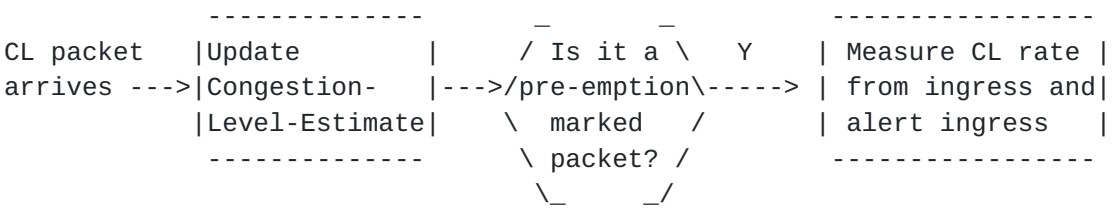


Figure 2: Egress gateway action for explicit Pre-emption Alert

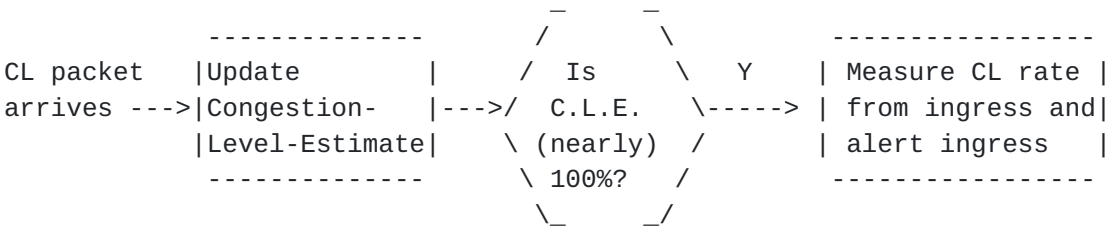


Figure 3: Egress gateway action for implicit Pre-emption Alert

3.2.2. Determining the right amount of CL traffic to drop

The method relies on the insight that the amount of CL traffic that can be supported between a particular pair of ingress and egress gateways, is the amount of CL traffic that is actually getting across the CL-region to the egress gateway without being Pre-emption Marked. Hence we term it the Sustainable-Aggregate-Rate.

So when the ingress gateway gets the Alert message from an egress gateway, it compares:

o The traffic rate that it is sending to this particular egress gateway (which we term Ingress-Aggregate-Rate)

o The traffic rate that the egress gateway reports (in the Alert message) that it is receiving from this ingress gateway (which is the Sustainable-Aggregate-Rate)

If the difference is significant, then the ingress gateway pre-empts some microflows. It only pre-empts if:

Ingress-Aggregate-Rate > Sustainable-Aggregate-Rate + error

The "error" term is partly to allow for inaccuracies in the measurements of the rates. It is also needed because the Ingress-Aggregate-Rate is measured at a slightly later moment than the Sustainable-Aggregate-Rate, and it is quite possible that the Ingress-Aggregate-Rate has increased in the interim due to natural variation of the bit rate of the CL sources. So the "error" term allows for some variation in the ingress rate without triggering pre-emption.

The ingress gateway should pre-empt enough microflows to ensure that:

New Ingress-Aggregate-Rate < Sustainable-Aggregate-Rate - error

The "error" term here is used for similar reasons but in the other direction, to ensure slightly more load is shed than seems necessary, in case the two measurements were taken during a short-term fall in load.

When the routers in the CL-region are using explicit pre-emption alerting, the ingress gateway would normally pre-empt microflows whenever it gets an alert (it always would if it were possible to set "error" equal to zero). For the implicit case however this is not so. It receives an Alert message when the Congestion-Level-Estimate reaches (almost) 100%, which is roughly when traffic exceeds the

configured-admission-rate. However, it is only when packets are indeed dropped en route that the Sustainable-Aggregate-Rate becomes less than the Ingress-Aggregate-Rate so only then will pre-emption actually occur on the ingress gateway.

Hence with the implicit scheme, pre-emption can only be triggered once the system starts dropping packets and thus the QoS of flows starts being significantly degraded. This is in contrast with the explicit scheme which allows flow pre-emption to be triggered before any packet drop, simply when the traffic reaches the configured-pre-emption-rate. Therefore we believe that the explicit mechanism is superior. However it does require new functionality on all the routers (although this is little more than a bulk token bucket - see [PCN] for details).

3.2.3. Use case for flow pre-emption

To see how the pieces of the solution fit together in a use case, we imagine a scenario where many microflows have already been admitted. We confine our description to the explicit pre-emption mechanism. Now an interior router in the CL-region fails. The network layer routing protocol re-routes round the problem, but as a consequence traffic on other links increases. In fact let's assume the traffic on one link now exceeds its configured-pre-emption-rate and so the router pre-emption marks CL packets. When the egress sees the first one of the pre-emption marked packets it immediately determines which microflow this packet is part of (by using a five-tuple filter and comparing it with state installed at admission) and hence which ingress gateway the packet came from. It sets up a meter to measure the traffic rate from this ingress gateway, and as soon as possible sends a message to the ingress gateway. This message alerts the ingress gateway that pre-emption may be needed and contains the traffic rate measured by the egress gateway. Then the ingress gateway determines the traffic rate that it is sending towards this egress gateway and hence it can calculate the amount of traffic that needs to be pre-empted.

The solution operates within a little over one round trip time - the time required for microflow packets that have experienced Pre-emption Marking to travel downstream through the CL-region and arrive at the egress gateway, plus some additional time for the egress gateway to measure the rate seen after it has been alerted that pre-emption may be needed, and the time for the egress gateway to report this information to the ingress gateway.

The ingress gateway could now just shed random microflows, but it is better if the least important ones are dropped. The ingress gateway could use information stored locally in each reservation's state (such as for example the RSVP pre-emption priority of [RSVP-PREEMPTION] or the RSVP admission priority of [RSVP-EMERGENCY]) as well as information provided by a policy decision point in order to decide which of the flows to shed (or perhaps which ones not to shed). This way, flow pre-emption can also helps emergency/military calls by taking into account the corresponding priorities (as conveyed in RSVP policy elements) when selecting calls to be pre-empted, which is likely to be particularly important in a disaster scenario. Use of RSVP for support of emergency/military applications is discussed in further details in [RFC4542] and [RSVP-EMERGENCY].

The ingress gateway then initiates RSVP signalling to instruct the relevant destinations that their reservation has been terminated, and to tell (RSVP) nodes along the path to tear down associated RSVP state. To guard against recalcitrant sources, normal IntServ policing may be used to block any future traffic from the dropped flows from entering the CL-region. Note that - with the explicit Pre-emption Alert mechanism - since the configured-pre-emption-rate may be significantly less than the physical line capacity, flow pre-emption may be triggered before any congestion has actually occurred and before any packet is dropped.

We extend the scenario further by imagining that (due to a disaster of some kind) further routers in the CL-region fail during the time taken by the pre-emption process described above. This is handled naturally, as packets will continue to be pre-emption marked and so the pre-emption process will happen for a second time.

3.3. Both admission control and pre-emption

This document describes both the admission control and pre-emption mechanisms, and we suggest that an operator uses both. However, we do not require this and some operators may want to implement only one.

For example, an operator could use just admission control, solving heavy congestion (caused by re-routing) by 'just waiting' - as sessions end, existing microflows naturally depart from the system over time, and the admission control mechanism will prevent admission of new microflows that use the affected links. So the CL-region will naturally return to normal controlled load service, but with reduced capacity. The drawback of this approach would be that until flows naturally depart to relieve the congestion, all flows and lower

priority services will be adversely affected. As another example, an operator could use just admission control, avoiding heavy congestion (caused by re-routing) by 'capacity planning' - by configuring admission control thresholds to lower levels than the network could accept in normal situations such that the load after failure is expected to stay below acceptable levels even with reduced network resources.

On the other hand, an operator could just rely for admission control on the traffic conditioning agreements of the DiffServ architecture [RFC2475]. The pre-emption mechanism described in this document would be used to counteract the problem described at the end of Section 1.1.1.

4. Summary of Functionality

This section is intended to provide a systematic summary of the new functionality required by the routers in the CL-region.

A network operator upgrades normal IP routers by:

- Adding functionality related to admission control and flow pre-emption to all its ingress and egress gateways

- Adding Pre-Congestion Notification for Admission Marking and Pre-emption Marking to all the routers in the CL-region.

We consider the detailed actions required for each of the types of router in turn.

4.1. Ingress gateways

Ingress gateways perform the following tasks:

- Classify incoming packets - decide whether they are CL or non-CL packets. This is done using an IntServ filter spec (source and destination addresses and port numbers), whose details have been gathered from the RSVP messaging.

- Police - check that the microflow conforms with what has been agreed (i.e. it keeps to its agreed data rate). If necessary, packets which do not correspond to any reservations, packets which are in excess of the rate agreed for their reservation, and packets for a reservation that has earlier been pre-empted may be policed. Policing may be achieved via dropping or via re-marking of the packet's DSCP to a value different from the CL behaviour aggregate.

- ECN colouring packets - for CL microflows, set the ECN field of packets appropriately (see [PCN] for some discussion of encoding).

- Perform 'interior router' functions (see next sub-section).

- Admission Control - on new session establishment, consider the Congestion-Level-Estimate received from the corresponding egress gateway and most likely based on a simple configured CLE-threshold decide if a new call is to be admitted or rejected (taking into account local policy information as well as optionally information provided by a policy decision point).

- Probe - if requested by the egress gateway to do so, the ingress gateway generates probe traffic so that the egress gateway can compute the Congestion-Level-Estimate from this ingress gateway. Probe packets may be simple data addressed to the egress gateway and require no protocol standardisation, although there will be best practice for their number, size and rate.

- Measure - when it receives a Pre-emption Alert message from an egress gateway, it determines the rate at which it is sending packets to that egress gateway

- Pre-empt - calculate how much CL traffic needs to be pre-empted; decide which microflows should be dropped, perhaps in consultation with a Policy Decision Point; and do the necessary signalling to drop them.

4.2. Interior routers

Interior routers do the following tasks:

- Classify packets - examine the DSCP and ECN field to see if it's a CL packet

- Non-CL packets are handled as usual, with respect to dropping them or setting their CE codepoint.

- Pre-Congestion Notification - CL packets are Admission Marked and Pre-emption Marked according to the algorithm detailed in [PCN] and outlined in Section 3.

4.3. Egress gateways

Egress gateways do the following tasks:

- Classify packets - determine which ingress gateway a CL packet has come from. This is the previous RSVP hop, hence the necessary details are obtained just as with IntServ from the state associated with the packet five-tuple, which has been built using information from the RSVP messages.

- Meter - for CL packets, calculate the fraction of the total number of bits which are in Admission marked packets or in Pre-emption Marked packets. The calculation is done as an exponentially weighted moving average (see Appendix C). A separate calculation is made for CL packets from each ingress gateway. The meter works on an aggregate basis and not per microflow.

- Signal the Congestion-Level-Estimate - this is piggy-backed on the reservation reply. An egress gateway's interface is configured to know it is an egress gateway, so it always appends this to the RESV message. If the Congestion-Level-Estimate is unknown or is too stale, then the egress gateway can request the ingress gateway to send probes.

- Packet colouring - for CL packets, set the DSCP and the ECN field to whatever has been agreed as appropriate for the next domain. By default the ECN field is set to the Not-ECT codepoint. See also the discussion in the Tunnelling section later.

- Measure the rate - measure the rate of CL traffic from a particular ingress gateway, excluding packets that are Pre-emption Marked (i.e. the Sustainable-Aggregate-Rate for the CL-region-aggregate), when alerted (either explicitly or implicitly) that pre-emption may be required. The measured rate is reported back to the appropriate ingress gateway [RSVP-PCN].

4.4. Failures

If an interior router fails, then the regular IP routing protocol will re-route round it. If the new route can carry all the admitted traffic, flows will gracefully continue. If instead this causes early warning of pre-congestion on the new route, then admission control based on pre-congestion notification will ensure new flows will not be admitted until enough existing flows have departed. Finally re-routing may result in heavy congestion, when the flow pre-emption mechanism will kick in.

If a gateway fails then we would like regular RSVP procedures [RFC2205] to take care of things. With the local repair mechanism of [RFC2205], when a route changes the next RSVP PATH refresh message will establish path state along the new route, and thus attempt to re-establish reservations through the new ingress gateway. Essentially the same procedure is used as described earlier in this document, with the re-routed session treated as a new session request.

In more detail, consider what happens if an ingress gateway of the CL-region fails. Then RSVP routers upstream of it do IP re-routing to a new ingress gateway. The next time the upstream RSVP router sends a PATH refresh message it reaches the new ingress gateway which therefore installs the associated RSVP state. The next RSVP RESV refresh will pick up the Congestion-Level-Estimate from the egress gateway, and the ingress compares this with its threshold to decide whether to admit the new session. This could result in some of the flows being rejected, but those accepted will receive the full QoS.

An issue with this is that we have to wait until a PATH and RESV refresh messages are sent - which may not be very often - the default value is 30 seconds. [RFC2205] discusses how to speed up the local repair mechanism. First, the RSVP module is notified by the local routing protocol module of a route change to particular destinations, which triggers it to rapidly send out PATH refresh messages. Further, when a PATH refresh arrives with a previous hop address different from the one stored, then RESV refreshes are immediately sent to that previous hop. Where RSVP is operating hop-by-hop, i.e. on every router, then triggering the PATH refresh is easy as the router can simply monitor its local link. Thus, this fast local repair mechanism can be used to deal with failures upstream of the ingress gateway, with failures of the ingress gateway and with failures downstream of the egress gateway.

But where RSVP is not operating hop-by-hop (as is the case within the CL-region), it is not so easy to trigger the PATH refresh.

Unfortunately, this problem applies if an egress gateway fails, since it's very likely that an egress gateway is several IP hops from the ingress gateway. (If the ingress is several IP hops from its previous RSVP node, then there is the same issue.) The options appear to be:

- the ingress gateway has a link state database for the CL-region, so it can detect that an egress gateway has failed or became unreachable

- there is an inter-gateway protocol, so the ingress can continuously check that the egress gateways are still alive

- (default) do nothing and wait for the regular PATH/RESV refreshes (and, if needed, the pre-emption mechanism) to sort things out.

5. Limitations and some potential solutions

In this section we describe various limitations of the deployment model, and some suggestions about potential ways of alleviating them. The limitations fall into three broad categories:

- o ECMP (Section 5.1): the assumption about routing (Section 2.2) is that all packets between a pair of ingress and egress gateways follow the same path; ECMP breaks this assumption. A study regarding the accuracy of load balancing schemes can be found in [LoadBalancing-a] and [LoadBalancing-b].

- o The lack of global coordination (Sections 5.2, 5.3 and 5.4): a decision about admission control or flow pre-emption is made for one aggregate independently of other aggregates

- o Timing and accuracy of measurements (Sections 5.5 and 5.6): the assumption (Section 2.2) that additional load, offered within the reaction time of the measurement-based admission control mechanism, doesn't move the system directly from no congestion to overload (dropping packets). A 'flash crowd' may break this assumption (Section 5.5). There are a variety of more general issues associated with marking measurements, which may mean it's a good idea to do pre-emption 'slower' (Section 5.6).

Each section describes a limitation and some possible solutions to alleviate the limitation. These are intended as options for an operator to consider, based on their particular requirements.

We would welcome feedback, for example suggestions as to which potential solutions are worth working out in more detail, and ideas on new potential solutions.

Finally Section 5.7 considers some other potential extensions.

5.1. ECMP

If the CL-region uses Equal Cost Multipath Routing (ECMP), then traffic between a particular pair of ingress and egress gateways may follow several different paths.

Why? An ECMP-enabled router runs an algorithm to choose between potential outgoing links, based on a hash of fields such as the packet's source and destination addresses - exactly what depends on the proprietary algorithm. Packets are addressed to the CL flow's

end-point, and therefore different flows may follow different paths through the CL-region. (All packets of an individual flow follow the same ECMP path.)

The problem is that if one of the paths is congested such that packets are being admission marked, then the Congestion-Level-Estimate measured by the egress gateway will be diluted by unmarked packets from other non-congested paths. Similarly, the measurement of the Sustainable-Aggregate-Rate will also be diluted.

Possible solution approaches are:

- o tunnel: traffic is tunnelled across the CL-region. Then the destination address (and so on) seen by the ECMP algorithm is that of the egress gateway, so all flows follow the same path. Effectively ECMP is turned off. As a compromise, to try to retain some of the benefits of ECMP, there could be several tunnels, each following a different ECMP path, with flows randomly assigned to different tunnels.

- o assume worst case: the operator sets the configured-admission-rate (and configured-pre-emption-rate) to below the optimum level to compensate for the fact that the effect on the Congestion-Level-Estimate (and Sustainable-Aggregate-Rate) of the congestion experienced over one of the paths may be diluted by traffic received over non-congested paths. Hence lower thresholds need to be used to ensure early admission control rejection and pre-emption over the congested path. This approach will waste capacity (e.g. flows following a non-congested ECMP path are not admitted or are pre-empted), and there is still the danger that for some traffic mixes the operator hasn't been cautious enough.

- o for admission control, probe to obtain a flow-specific congestion-level-estimate. Earlier this document suggests continuously monitoring the congestion-level-estimate. Instead, probe packets could be sent for each prospective new flow. The probe packets have the same IP address etc as the data packets would have, and hence follow the same ECMP path. However, probing is an extra overhead, depending on how many probe packets need to be sent to get a sufficiently accurate congestion-level-estimate. Probes also cause a processing overhead, either for the machine at the destination address or for the egress gateway to identify and remove the probe packets.

- for flow pre-emption, only select flows for pre-emption from amongst those that have actually received a Pre-emption Marked packet. Because these flows must have followed an ECMP path that goes through an overloaded router. However, it needs some extra work by the egress gateway, to record this information and report it to the ingress gateway.

- for flow pre-emption, a variant of this idea involves introducing a new marking behaviour, 'Router Marking'. A router that is pre-emption marking packets on an outgoing link, also 'Router Marks' all other packets. When selecting flows for pre-emption, the selection is made from amongst those that have actually received a Router Marked or Pre-emption Marked packet. Hence compared with the previous bullet, it may extend the range of flows from which the pre-emption selection is made (i.e. it includes those which, by chance, haven't had any pre-emption marked packets). However, it also requires that the 'Router Marking' state is somehow encoded into a packet, i.e. it makes harder the encoding challenge discussed in Appendix C of [PCN]. The extra work required by the egress gateway would also be somewhat higher than for the previous bullet.

5.2. Beat down effect

This limitation concerns the pre-emption mechanism in the case where more than one router is pre-emption marking packets. The result (explained in the next paragraph) is that the measurement of sustainable-aggregate-rate is lower than its true value, so more traffic is pre-empted than necessary.

Imagine the scenario:

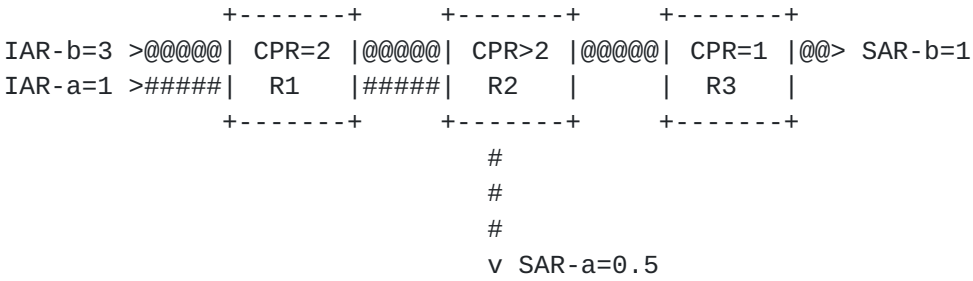


Figure 4: Scenario to illustrate 'beat down effect' limitation

Aggregate-a (ingress-aggregate-rate, IAR, 1 unit) takes a 'short' route through two routers, one of which (R1) is above its configured-

pre-emption-rate (CPR, 2 units). Aggregate-b takes a 'long' route, going through a second congested router (R3, with a CPR of 1 unit).

R1's input traffic is 4 units, twice its configured-pre-emption-rate, so 50% of packets are pre-emption marked. Hence the measured sustainable-aggregate-rate (SAR) for aggregate-a is 0.5, and half of its traffic will be pre-empted.

R3's input of non-pre-emption-marked traffic is 1.5 units, and therefore it has to do further marking.

But this means that aggregate-a has taken a bigger hit than it needed to; the router R1 could have let through all of aggregate-a's traffic unmarked if it had known that the second router R2 was going to "beat down" aggregate-b's traffic further.

Generalising, the result is that in a scenario where more than one router is pre-emption marking packets, only the final router is sure to be fully loaded after flow pre-emption. The fundamental reason is that a router makes a local decision about which packets to pre-emption mark, i.e. independently of how other routers are pre-emption marking. A very similar effect has been noted in XCP [Low].

Potential solutions:

- o a full solution would involve routers learning about other routers that are pre-emption marking, and being able to differentially mark flows (e.g. in the example above, aggregate-a's packets wouldn't be marked by R1). This seems hard and complex.

- o do nothing about this limitation. It causes over-pre-emption, which is safe. At the moment this is our suggested option.

- o do pre-emption 'slowly'. The description earlier in this document assumes that after the measurements of ingress-aggregate-rate and sustainable-aggregate-rate, then sufficient flows are pre-empted in 'one shot' to eliminate the excess traffic. An alternative is to spread pre-emption over several rounds: initially, only pre-empt enough to eliminate some of the excess traffic, then re-measure the sustainable-aggregate-rate, and then pre-empt some more, etc. In the scenario above, the re-measurement would be lower than expected, due to the beat down effect, and hence in the second round of pre-emption less of aggregate-a's traffic would be pre-empted (perhaps none). Overall, therefore the impact of the 'beat down' effect would be lessened, i.e. there would be a smaller degree of over pre-emption. The downside is that the overall pre-emption is slower, and therefore routers will be congested longer.

5.3. Bi-directional sessions

The document earlier describes how to decide whether or not to admit (or pre-empt) a particular flow. However, from a user/application perspective, the session is the relevant unit of granularity. A session can consist of several flows which may not all be part of the same aggregate. The most obvious example is a bi-directional session, where the two flows should ideally be admitted or pre-empted as a pair - for instance a voice call only makes sense if A can send to B as well as B to A! But the admission and pre-emption mechanisms described earlier in this document operate on a per-aggregate basis, independently of what's happening with other aggregates. For admission control the problem isn't serious: e.g. the SIP server for the voice call can easily detect that the A-to-B flow has been admitted but the B-to-A flow blocked, and inform the user perhaps via a busy tone. For flow pre-emption, the problem is similar but more serious. If both the aggregate-1-to-2 (i.e. from gateway 1 to gateway 2) and the aggregate-2-to-1 have to pre-empt flows, then it would be good if either all of the flows of a particular session were pre-empted or none of them. Therefore if the two aggregates pre-empt flows independently of each other, more sessions will end up being torn down than is really necessary. For instance, pre-empting one direction of a voice call will result in the SIP server tearing down the other direction anyway.

Potential solutions:

- o if it's known that all session are bi-directional, simply pre-empting roughly half as many flows as suggested by the measurements of {ingress-aggregate-rate - sustainable-aggregate-rate}. But this makes a big assumption about the nature of sessions, and also that the aggregate-1-to-2 and aggregate-2-to-1 are equally overloaded.

- o ignore the limitation. The penalty will be quite small if most sessions consist of one flow or of flows part of the same aggregate.

- o introduce a gateway controller. It would receive reports for all aggregates where the ingress-aggregate-rate exceeds the sustainable-aggregate-rate. It then would make a global decision about which flows to pre-empt. However it requires quite some complexity, for example the controller needs to understand which flows map to which sessions. This may be an option in some scenarios, for example where gateways aren't handling too many flows (but note that this breaks the aggregation assumption of Section 2.2). A variant of this idea would be to introduce a gateway controller per pair of gateways, in order to handle bi-directional sessions but not try to deal with more complex sessions that include flows from an arbitrary number of aggregates.

- o do pre-emption 'slowly'. As in the "beat down" solution 4, this would reduce the impact of this limitation. The downside is that the overall pre-emption is slower, and therefore router(s) will be congested longer.

- each ingress gateway 'loosely coordinates' with other gateways its decision about which specific flows to pre-empt. Each gateway numbers flows in the order they arrive (note that this number has no meaning outside the gateway), and when pre-empting flows, the most recent (or most recent low priority flow) is selected for pre-emption; the gateway then works backwards selecting as many flows as needed. Gateways will therefore tend to pre-empt flows that are part of the same session (as they were admitted at the same time). Of course this isn't guaranteed for several reasons, for instance gateway A's most recent bi-directional sessions may be with gateway C, whereas gateway B's are with gateway A (so gateway A will pre-empt A-to-C flows and gateway B will pre-empt B-to-A flows). Rather than pre-empting the most recent (low priority) flow, an alternative algorithm (for further study) may be to select flows based on a hash of particular fields in the packet, such that both gateways produce the same hash for flows of the same bi-directional session. We believe that this approach should be investigated further.

5.4. Global fairness

The limitation here is that 'high priority' traffic may be pre-empted (or not admitted) when a global decision would instead pre-empt (or not admit) 'lower priority' traffic on a different aggregate.

Imagine the following scenario (extreme to illustrate the point clearly). Aggregate_a is all Assured Services (MLPP) traffic, whilst aggregate_b is all ordinary traffic (i.e. comparatively low priority). Together the two aggregates cause a router to be at twice its configured-pre-emption-rate. Ideally we'd like all of aggregate_b to be pre-empted, as then all of aggregate_a could be carried. However, the approach described earlier in this document leads to half of each aggregate being pre-empted.

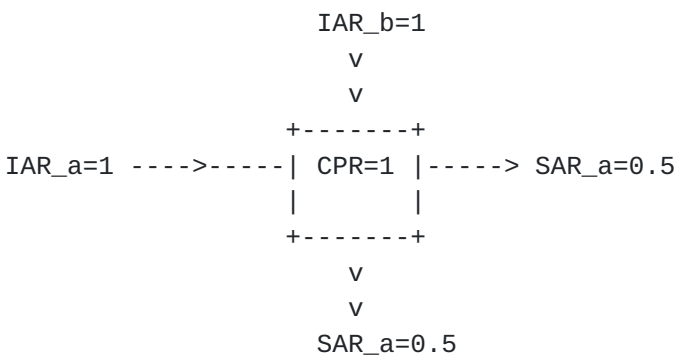


Figure 5: Scenario to illustrate 'global fairness' limitation

Similarly, for admission control - Section 4.1 describes how if the Congestion-Level-Estimate is greater than the CLE-threshold all new sessions are refused. But it is unsatisfactory to block emergency calls, for instance.

Potential solutions:

- o in the admission control case, it is recommended that an 'emergency / Assured Services' call is admitted immediately even if the CLE-threshold is exceeded. Usually the network can actually handle the additional microflow, because there is a safety margin between the configured-admission-rate and the configured-pre-emption-rate. Normal call termination behaviour will soon bring the traffic level down below the configured-admission-rate. However, in exceptional circumstances the 'emergency / higher precedence' call may cause the traffic level to exceed the configured-pre-emption-rate; then the usual pre-emption mechanism will pre-empt enough (non 'emergency / higher precedence') microflows to bring the total traffic back under the configured-pre-emption-rate.

- o all egress gateways report to a global coordinator that makes decisions about what flows to pre-empt. However this solution adds complexity and probably isn't scalable, but it may be an option in some scenarios, for example where gateways aren't handling too many flows (but note that this breaks the aggregation assumption of Section 2.2).

- introduce a heuristic rule: before pre-empting a 'high priority' flow the egress gateway should wait to see if sufficient (lower priority) traffic is pre-empted on other aggregates. This is a reasonable option.

- enhance the functionality of all the interior routers, so they can detect the priority of a packet, and then differentially mark them. As well as adding complexity, in general this would be an unacceptable security risk for MLPP traffic, since only controlled nodes (like gateways) should know which packets are high priority, as this information can be abused by an attacker.

- do nothing, i.e. accept the limitation. Whilst it's unlikely that high priority calls will be quite so unbalanced as in the scenario above, just accepting this limitation may be risky. The sorts of situations that cause routers to start pre-emption marking are also likely to cause a surge of emergency / MLPP calls.

5.5. Flash crowds

This limitation concerns admission control and arises because there is a time lag between the admission control decision (which depends on the Congestion-Level-Estimate during RSVP signalling during call set-up) and when the data is actually sent (after the called party has answered). In PSTN terms this is the time the phone rings. Normally the time lag doesn't matter much because (1) in the CL-region there are many flows and they terminate and are answered at roughly the same rate, and (2) the network can still operate safely when the traffic level is some margin above the configured-admission-rate.

A 'flash crowd' occurs when something causes many calls to be initiated in a short period of time - for instance a 'tele-vote'. So there is a danger that a 'flash' of calls is accepted, but when the calls are answered and data flows the traffic overloads the network. Therefore potentially the 'additional load' assumption of Section 2.2 doesn't hold.

Potential solutions:

- The simplest option is to do nothing; an operator relies on the pre-emption mechanism if there is a problem. This doesn't seem a good choice, as 'flash crowds' are reasonably common on the PSTN, unless the operator can ensure that nearly all 'flash crowd' events are blocked in the access network and so do not impact on the CL-region.

- A second option is to send 'dummy data' as soon as the call is admitted, thus effectively reserving the bandwidth whilst waiting for the called party to answer. Reserving bandwidth in advance means that the network cannot admit as many calls. For example, suppose sessions last 100 seconds and ringing for 10 seconds, the cost is a 10% loss of capacity. It may be possible to offset this somewhat by increasing the configured-admission-rate in the routers, but it would need further investigation. A concern with this 'dummy data' option is that it may allow an attacker to initiate many calls that are never answered (by a cooperating attacker), so eventually the network would only be carrying 'dummy data'. The attack exploits that charging only starts when the call is answered and not when it is dialled. It may be possible to alleviate the attack at the session layer - for example, when the ingress gateway gets an RSVP PATH message it checks that the source has been well-behaved recently; and limiting the maximum time that ringing can last. We believe that if this attack can be dealt with then this is a good option.

- A third option is that the egress gateway limits the rate at which it sends out the Congestion-Level-Estimate, or limits the rate at which calls are accepted by replying with a Congestion-Level-Estimate of 100% (this is the equivalent of 'call gapping' in the PSTN). There is a trade-off, which would need to be investigated further, between the degree of protection and possible adverse side-effects like slowing down call set-up.

- A final option is to re-perform admission control before the call is answered. The ingress gateway monitors Congestion-Level-Estimate updates received from each egress. If it notices that a Congestion-Level-Estimate has risen above the CLE-threshold, then it terminates all unanswered calls through that egress (e.g. by instructing the session protocol to stop the 'ringing tone'). For extra safety the Congestion-Level-Estimate could be re-checked when the call is answered. A potential drawback for an operator that wants to emulate the PSTN is that the PSTN network never drops a 'ringing' PSTN call.

5.6. Pre-empting too fast

As a general idea it seems good to pre-empt excess flows rapidly, so that the full QoS is restored to the remaining CL users as soon as possible, and partial service is restored to lower priority traffic classes on shared links. Therefore the pre-emption mechanism described earlier in this document works in 'one shot', i.e. one measurement is made of the sustainable-aggregate-rate and the ingress-aggregate-rate, and the excess is pre-empted immediately. However, there are some reasons why an operator may potentially want to pre-empt 'more slowly':

- To allow time to modify the ingress gateway's policer, as the ingress wants to be able to drop any packets that arrive from a pre-empted flow. There will be a limit on how many new filters an ingress gateway can install in a certain time period. Otherwise the source may cheat and ignore the instruction to drop its flow.

- The operator may decide to slow down pre-emption in order to ameliorate the 'beat down' and/or 'bi-directional sessions' limitations (see above)

- To help combat inaccuracies in measurements of the sustainable-aggregate-rate and ingress-aggregate-rate. For a CL-region where it's assumed there are many flows in an aggregate these measurements can be obtained in a short period of time, but where there are fewer flows it will take longer.

- To help combat over pre-emption because, during the time it takes to pre-empt flows, others may be ending anyway (either the call has naturally ended, or the user hangs up due to poor QoS). Slowing pre-emption may seem counter-intuitive here, as it makes it more likely that calls will terminate anyway - however it also gives time to adjust the amount pre-empted to take account of this.

- Earlier in this document we said that an egress starts measuring the sustainable-aggregate-rate immediately it sees a single pre-emption marked packet. However, when a link or router fails the network's underlying recovery mechanism will kick in (e.g. switching to a back up path), which may result in the network again being able to support all the traffic.

Potential solutions

o To combat the final issue, the egress could measure the sustainable-aggregate-rate over a longer time period than the network recovery time (say 100ms vs. 50ms). If it detects no pre-emption marked packets towards the end of its measurement period (say in the last 30 ms) then it doesn't send a pre-emption alert message to the ingress.

o We suggest that optionally (the choice of the operator) pre-emption is slowed by pre-empting traffic in several rounds rather than in one shot. One possible algorithm is to pre-empt most of the traffic in the first round and the rest in the second round; the amount pre-empted in the second round is influenced by both the first and second round measurements: * Round 1: pre-empt h * S_1 where 0.5 <= h <= 1 where S_1 is the amount the normal mechanism calculates that it should shed, i.e. {ingress-aggregate-rate - sustainable-aggregate-rate} * Round 2: pre-empt Predicted-S_2 - h * (Predicted-S_2 - Measured-S_2) where Predicted-S_2 = (1-h)*S_1 Note that the second measurement should be made when sufficient time has elapsed for the first round of pre-emption to have happened. One idea to achieve this is for the egress gateway to continuously measure and report its sustainable-aggregate-rate, in (say) 100ms windows. Therefore the ingress gateway knows when the egress gateway made its measurement (assuming the round trip time is known). Therefore the ingress gateway knows when measurements should reflect that it has pre-empted flows.

5.7. Other potential extensions

In this section we discuss some other potential extensions not already covered above.

5.7.1. Tunnelling

It is possible to tunnel all CL packets across the CL-region. Although there is a cost of tunnelling (additional header on each packet, additional processing at tunnel ingress and egress), there are three reasons it may be interesting.

ECMP:

Tunnelling is one of the possible solutions given earlier in Section 5.1 on Equal Cost Multipath Routing (ECMP).

Ingress gateway determination:

If packets are tunnelled from ingress gateway to egress gateway, the egress gateway can very easily determine in the data path which ingress gateway a packet comes from (by simply looking at the source address of the tunnel header). This can facilitate operations such as computing the Congestion-Level-Estimate on a per ingress gateway basis.

End-to-end ECN:

The ECN field is used for PCN marking (see [PCN] for details), and so it needs to be re-set by the egress gateway to whatever has been agreed as appropriate for the next domain. Therefore if a packet arrives at the ingress gateway with its ECN field already set (i.e. not '00'), it may leave the egress gateway with a different value. Hence the end-to-end meaning of the ECN field is lost.

It is open to debate whether end-to-end congestion control is ever necessary within an end-to-end reservation. But if a genuine need is identified for end-to-end ECN semantics within a reservation, then one solution is to tunnel CL packets across the CL-region. When the egress gateway decapsulates them the original ECN field is recovered.

5.7.2. Multi-domain and multi-operator usage

This potential extension would eliminate the trust assumption (Section 2.2), so that the CL-region could consist of multiple domains run by different operators that did not trust each other. Then only the ingress and egress gateways of the CL-region would take part in the admission control procedure, i.e. at the ingress to the first domain and the egress from the final domain. The border routers between operators within the CL-region would only have to do bulk accounting - they wouldn't do per microflow metering and policing, and they wouldn't take part in signal processing or hold per flow state [Briscoe]. [Re-feedback] explains how a downstream domain can police that its upstream domain does not 'cheat' by admitting traffic when the downstream path is congested. [Re-PCN] proposes how to achieve this with the help of another recently proposed extension to ECN, involving re-echoing ECN feedback [Re-ECN].

5.7.3. Preferential dropping of pre-emption marked packets

When the rate of real-time traffic in the specified class exceeds the maximum configured rate, then a router has to drop some packet(s) instead of forwarding them on the out-going link. Now when the egress gateway measures the Sustainable-Aggregate-Rate, neither dropped packets nor pre-emption marked packets contribute to it. Dropping non-pre-emption-marked packets therefore reduces the measured Sustainable-Aggregate-Rate below its true value. Thus a router should preferentially drop pre-emption marked packets.

Note that it is important that the operator doesn't set the configured-pre-emption-rate equal to the rate at which packets start being dropped (for the specified real-time service class). Otherwise the egress gateway may never see a pre-emption marked packet and so won't be triggered into the Pre-emption Alert state.

This optimisation is optional. When considering whether to use it an operator will consider issues such as whether the over-pre-emption is serious, and whether the particular routers can easily do this sort of selective drop.

5.7.4. Adaptive bandwidth for the Controlled Load service

The admission control mechanism described in this document assumes that each router has a fixed bandwidth allocated to CL flows. A possible extension is that the bandwidth is flexible, depending on the level of non-CL traffic. If a large share of the current load on a path is CL, then more CL traffic can be admitted. And if the greater share of the load is non-CL, then the admission threshold can be proportionately lower. The approach re-arranges sharing between classes to aim for economic efficiency, whatever the traffic load matrix. It also deals with unforeseen changes to capacity during failures better than configuring fixed engineered rates. Adaptive bandwidth allocation can be achieved by changing the admission marking behaviour, so that the probability of admission marking a packet would now depend on the number of queued non-CL packets as well as the size of the virtual queue. The adaptive bandwidth approach would be supplemented by placing limits on the adaptation to prevent starvation of the CL by other traffic classes and of other classes by CL traffic. [Songhurst] has more details of the adaptive bandwidth approach.

5.7.5. Controlled Load service with end-to-end Pre-Congestion Notification

It may be possible to extend the framework to parts of the network where there are only a low number of CL microflows, i.e. the aggregation assumption (Section 2.2) doesn't hold. In the extreme it may be possible to operate the framework end-to-end, i.e. between end hosts. One potential method is to send probe packets to test whether the network can support a prospective new CL microflow. The probe packets would be sent at the same traffic rate as expected for the actual microflow, but in order not to disturb existing CL traffic a router would always schedule probe packets behind CL ones (compare [Breslau00]); this implies they have a new DSCP. Otherwise the routers would treat probe packets identically to CL packets. In order to perform admission control quickly, in parts of the network where there are only a few CL microflows, the algorithm for Admission Marking described in [PCN] would need to "switch on" very rapidly, ie go from marking no packets to marking them all for only a minimal increase in the size of the virtual queue.

5.7.6. MPLS-TE

[ECN-MPLS] discusses how to extend the deployment model to MPLS, i.e. for admission control of microflows into a set of MPLS-TE aggregates (Multi-protocol label switching traffic engineering). It would require that the MPLS header could include the ECN field, which is not precluded by RFC3270. See [ECN-MPLS].

6. Relationship to other QoS mechanisms

6.1. IntServ Controlled Load

The CL mechanism delivers QoS similar to Integrated Services controlled load, but rather better. The reason the QoS is better is that the CL mechanism keeps the real queues empty, by driving admission control from a bulk virtual queue on each interface. The virtual queue [AVQ, vq] can detect a rise in load before the real queue builds. It is also more robust to route changes.

6.2. Integrated services operation over DiffServ

Our approach to end-to-end QoS is similar to that described in [RFC2998] for Integrated services operation over DiffServ networks. Like [RFC2998], an IntServ class (CL in our case) is achieved end-to-end, with a CL-region viewed as a single reservation hop in the total end-to-end path. Interior routers of the CL-region do not process flow signalling nor do they hold per flow state. Unlike [RFC2998] we do not require the end-to-end signalling mechanism to be RSVP, although it can be.

Bearing in mind these differences, we can describe our architecture in the terms of the options in [RFC2998]. The DiffServ network region is RSVP-aware, but awareness is confined to (what [RFC2998] calls) the "border routers" of the DiffServ region. We use explicit admission control into this region, with static provisioning within it. The ingress "border router" does per microflow policing and sets the DSCP and ECN fields to indicate the packets are CL ones (i.e. we use router marking rather than host marking).

6.3. Differentiated Services

The DiffServ architecture does not specify any way for devices outside the domain to dynamically reserve resources or receive indications of network resource availability. In practice, service providers rely on subscription-time Service Level Agreements (SLAs) that statically define the parameters of the traffic that will be accepted from a customer. The CL mechanism allows dynamic reservation of resources through the DiffServ domain and, with the potential extension mentioned in Section 5.7.2, it can span multiple domains without active policing mechanisms at the borders (unlike DiffServ). Therefore we do not use the traffic conditioning agreements (TCAs) of the (informational) DiffServ architecture [RFC2475].

An important benefit arises from the fact that the load is controlled dynamically rather than with traffic conditioning agreements (TCAs).

TCAs were originally introduced in the (informational) DiffServ architecture [RFC2475] as an alternative to reservation processing in the interior region in order to reduce the burden on interior routers. With TCAs, in practice service providers rely on subscription-time Service Level Agreements that statically define the parameters of the traffic that will be accepted from a customer. The problem arises because the TCA at the ingress must allow any destination address, if it is to remain scalable. But for longer topologies, the chances increase that traffic will focus on an interior resource, even though it is within contract at the ingress [Reid], e.g. all flows converge on the same egress gateway. Even though networks can be engineered to make such failures rare, when they occur all inelastic flows through the congested resource fail catastrophically.

[Johnson] compares admission control with a 'generously dimensioned' DiffServ network as ways to achieve QoS. The former is recommended.

6.4. ECN

The marking behaviour described in this document complies with the ECN aspects of the IP wire protocol RFC3168, but provides its own edge-to-edge feedback instead of the TCP aspects of RFC3168. All routers within the CL-region are upgraded with the admission marking and pre-emption marking of Pre-Congestion Notification, so the requirements of [Floyd] are met because the CL-region is an enclosed environment. The operator prevents traffic arriving at a router that doesn't understand CL by administrative configuration of the ring of gateways around the CL-region.

6.5. RTECN

Real-time ECN (RTECN) [RTECN, RTECN-usage] has a similar aim to this document (to achieve a low delay, jitter and loss service suitable for RT traffic) and a similar approach (per microflow admission control combined with an "early warning" of potential congestion through setting the CE codepoint). But it explores a different architecture without the aggregation assumption: host-to-host rather than edge-to-edge. We plan to document such a host-to-host framework in a parallel draft to this one, and to describe if and how [PCN] can work in this framework.

6.6. RMD

Resource Management in DiffServ (RMD) [RMD] is similar to this work, in that it pushes complex classification, traffic conditioning and admission control functions to the edge of a DiffServ domain and simplifies the operation of the interior routers. One of the RMD modes ("Congestion notification function based on probing") uses measurement-based admission control in a similar way to this document. The main difference is that in RMD probing plays a significant role in the admission control process. Other differences are that the admission control decision is taken on the egress gateway (rather than the ingress); 'admission marking' is encoded in a packet as a new DSCP (rather than in the ECN field), and that the NSIS protocols are used for signalling (rather than RSVP).

RMD also includes the concept of Severe Congestion handling. The pre-emption mechanism described in the CL architecture has similar objectives but relies on different mechanisms. The main difference is that the interior routers measure the data rate that causes an overload and mark packets according to this rate.

6.7. RSVP Aggregation over MPLS-TE

Multi-protocol label switching traffic engineering (MPLS-TE) allows scalable reservation of resources in the core for an aggregate of many microflows. To achieve end-to-end reservations, admission control and policing of microflows into the aggregate can be achieved using techniques such as RSVP Aggregation over MPLS TE Tunnels as per [AGGRE-TE]. However, in the case of inter-provider environments, these techniques require that admission control and policing be repeated at each trust boundary or that MPLS TE tunnels span multiple domains.

6.8. Other Network Admission Control Approaches

Link admission control (LAC) describes how admission control (AC) can be done on a single link and comprises, e.g., the calculation of effective bandwidths which may be the base for a parameter-based AC. In contrast, network AC (NAC) describes how AC can be done for a network and focuses on the locations from which data is gathered for the admission decision. Most approaches implement a link budget based NAC (LB NAC) where each link has a certain AC-budget. RSVP works according to that principle, but also the new concept admits additional flows as long as each link on the new flow's path still has resources available. The border-to-border budget based NAC (BBB NAC) pre-configures an AC budget for all border-to-border relationships (= CL-region-aggregates) and if this capacity budget is

exhausted, new flows are rejected. The TCA-based admission control which is associated with the DiffServ architecture implements an ingress budget based NAC (IB NAC). These basically different concepts have different flexibility and efficiency with regard to the use of link bandwidths [NAC-a,NAC-b]. They can be made resilient by choosing the budgets in such a way that the network will not be congested after rerouting due to a failure. The efficiency of the approaches is different with and without such resilient requirements.

7. Security Considerations

To protect against denial of service attacks, the ingress gateway of the CL-region needs to police all CL packets and drop packets in excess of the reservation. This is similar to operations with existing IntServ behaviour.

For pre-emption, it is considered acceptable from a security perspective that the ingress gateway can treat "emergency/military" CL flows preferentially compared with "ordinary" CL flows. However, in the rest of the CL-region they are not distinguished (nonetheless, our proposed technique does not preclude the use of different DSCPs at the packet level as well as different priorities at the flow level.). Keeping emergency traffic indistinguishable at the packet level minimises the opportunity for new security attacks. For example, if instead a mechanism used different DSCPs for "emergency/military" and "ordinary" packets, then an attacker could specifically target the former in the data plane (perhaps for DoS or for eavesdropping).

Further security aspects to be considered later.

8. Acknowledgements

The admission control mechanism evolved from the work led by Martin Karsten on the Guaranteed Stream Provider developed in the M3I project [GSPa, GSP-TR], which in turn was based on the theoretical work of Gibbens and Kelly [DCAC]. Kennedy Cheng, Gabriele Corliano, Carla Di Cairano-Gilfedder, Kashaf Khan, Peter Hovell, Arnaud Jacquet and June Tay (BT) helped develop and evaluate this approach.

Many thanks to those who have commented on this work at Transport Area Working Group meetings and on the mailing list, including: Ken

Carlberg, Ruediger Geib, Lars Westberg, David Black, Robert Hancock, Cornelia Kappler, Michael Menth.

9. Comments solicited

Comments and questions are encouraged and very welcome. They can be sent to the Transport Area Working Group's mailing list, tsvwg@ietf.org, and/or to the authors.

10. Changes from earlier versions of the draft

The main changes are:

From -00 to -01

The whole of the Pre-emption mechanism is added.

There are several modifications to the admission control mechanism.

From -01 to -02

The pre-congestion notification algorithms for admission marking and pre-emption marking are now described in [PCN].

There are new sub-sections in Section 4 on Failures, Admission of 'emergency / higher precedence' session, and Tunnelling; and a new sub-section in Section 5 on Mechanisms to deal with 'Flash crowds'.

From -02 to -03

Section 5 has been updated and expanded. It is now about the 'limitations' of the PCN mechanism, as described in the earlier sections, plus discussion of 'possible solutions' to those limitations.

The measurement of the Congestion-Level-Estimate now includes pre-emption marked packets as well as admission marked ones. Section 3.1.2 explains.

From -03 to -04

Detailed review by Michael Menth. In response, Abstract, Summary and Key benefits sections re-written. Numerous detailed comments on Sections 5 and following sections.

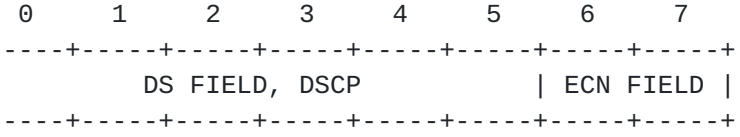
11. Appendices

11.1. Appendix A: Explicit Congestion Notification

This Appendix provides a brief summary of Explicit Congestion Notification (ECN).

[RFC3168] specifies the incorporation of ECN to TCP and IP, including ECN's use of two bits in the IP header. It specifies a method for indicating incipient congestion to end-hosts (e.g. as in RED, Random Early Detection), where the notification is through ECN marking packets rather than dropping them.

ECN uses two bits in the IP header of both IPv4 and IPv6 packets:



DSCP: differentiated services codepoint
ECN: Explicit Congestion Notification

Figure A.1: The Differentiated Services and ECN Fields in IP.

The two bits of the ECN field have four ECN codepoints, '00' to '11':

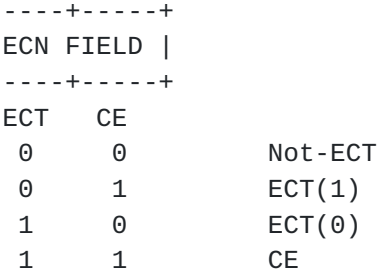


Figure A.2: The ECN Field in IP.

The not-ECT codepoint '00' indicates a packet that is not using ECN.

The CE codepoint '11' is set by a router to indicate congestion to the end hosts. The term 'CE packet' denotes a packet that has the CE codepoint set.

The ECN-Capable Transport (ECT) codepoints '10' and '01' (ECT(0) and ECT(1) respectively) are set by the data sender to indicate that the

end-points of the transport protocol are ECN-capable. Routers treat the ECT(0) and ECT(1) codepoints as equivalent. Senders are free to use either the ECT(0) or the ECT(1) codepoint to indicate ECT, on a packet-by-packet basis. The motivation for having two codepoints (the 'ECN nonce') is the desire to check two things: for the data sender to verify that network elements are not erasing the CE codepoint; and for the data sender to verify that data receivers are properly reporting to the sender the receipt of packets with the CE codepoint set.

ECN requires support from the transport protocol, in addition to the functionality given by the ECN field in the IP packet header. [RFC3168] addresses the addition of ECN Capability to TCP, specifying three new pieces of functionality: negotiation between the endpoints during connection setup to determine if they are both ECN-capable; an ECN-Echo (ECE) flag in the TCP header so that the data receiver can inform the data sender when a CE packet has been received; and a Congestion Window Reduced (CWR) flag in the TCP header so that the data sender can inform the data receiver that the congestion window has been reduced.

The transport layer (e.g.. TCP) must respond, in terms of congestion control, to a *single* CE packet as it would to a packet drop.

The advantage of setting the CE codepoint as an indication of congestion, instead of relying on packet drops, is that it allows the receiver(s) to receive the packet, thus avoiding the potential for excessive delays due to retransmissions after packet losses.

11.2. Appendix B: What is distributed measurement-based admission control?

This Appendix briefly explains what distributed measurement-based admission control is [Breslau99].

Traditional admission control algorithms for 'hard' real-time services (those providing a firm delay bound for example) guarantee QoS by using 'worst case analysis'. Each time a flow is admitted its traffic parameters are examined and the network re-calculates the remaining resources. When the network gets a new request it therefore knows for certain whether the prospective flow, with its particular parameters, should be admitted. However, parameter-based admission control algorithms result in under-utilisation when the traffic is bursty. Therefore 'soft' real time services - like Controlled Load - can use a more relaxed admission control algorithm.

This insight suggests measurement-based admission control (MBAC). The aim of MBAC is to provide a statistical service guarantee. The classic scenario for MBAC is where each router participates in hop-by-hop admission control, characterising existing traffic locally through measurements (instead of keeping an accurate track of traffic as it is admitted), in order to determine the current value of some parameter e.g. load. Note that for scalability the measurement is of the aggregate of the flows in the local system. The measured parameter(s) is then compared to the requirements of the prospective flow to see whether it should be admitted.

MBAC may also be performed centrally for a network, it which case it uses centralised measurements by a bandwidth broker.

We use distributed MBAC. "Distributed" means that the measurement is accumulated for the 'whole-path' using in-band signalling. In our case, this means that the measurement of existing traffic is for the same pair of ingress and egress gateways as the prospective microflow.

In fact our mechanism can be said to be distributed in three ways: all routers on the ingress-egress path affect the Congestion-Level-Estimate; the admission control decision is made just once on behalf of all the routers on the path across the CL-region; and the ingress and egress gateways cooperate to perform MBAC.

11.3. Appendix C: Calculating the Exponentially weighted moving average (EWMA)

At the egress gateway, for every CL packet arrival:

[EWMA-total-bits]n+1 = (w * bits-in-packet) + ((1-w) * [EWMA-total-bits]n)

[EWMA-M-bits]n+1 = (B * w * bits-in-packet) + ((1-w) * [EWMA-M-bits]n)

Then, per new flow arrival:

[Congestion-Level-Estimate]n+1 = [EWMA-M-bits]n+1 / [EWMA-total-bits]n+1

where

EWMA-total-bits is the total number of bits in CL packets, calculated as an exponentially weighted moving average (EWMA)

EWMA-M-bits is the total number of bits in CL packets that are Admission Marked or Pre-emption Marked, again calculated as an EWMA.

B is either 0 or 1:

B = 0 if the CL packet is not admission marked

B = 1 if the CL packet is admission marked

w is the exponential weighting factor.

Varying the value of the weight trades off between the smoothness and responsiveness of the Congestion-Level-Estimate. However, in general both can be achieved, given our original assumption of many CL microflows and remembering that the EWMA is calculated on the basis of aggregate traffic between the ingress and egress gateways. There will be a threshold inter-arrival time between packets of the same aggregate below which the egress will consider the estimate of the Congestion-Level-Estimate as too stale, and it will then trigger generation of probes by the ingress.

The first two per-packet algorithms can be simplified, if their only use will be where the result of one is divided by the result of the other in the third, per-flow algorithm.

[EWMA-total-bits]'n+1 = bits-in-packet + (w' * [EWMA- total-bits]n)

[EWMA-AM-bits]'n+1 = (B * bits-in-packet) + (w' * [EWMA-AM-bits]n)

where w' = (1-w)/w.

If w' is arranged to be a power of 2, these per packet algorithms can be implemented solely with a shift and an add.

There are alternative possibilities for smoothing out the congestion-level-estimate. For example [TEWMA] deals better with the issue of stale information when the traffic rate for

12. References

A later version will distinguish normative and informative references.

[AGGRE-TE] Francois Le Faucheur, Michael Dibiasio, Bruce Davie, Michael Davenport, Chris Christou, Jerry Ash, Bur Goode, 'Aggregation of RSVP Reservations over MPLS TE/DS-TE Tunnels', draft-ietf-tsvwg-rsvp-dste-03 (work in progress), June 2006

[ANSI.MLPP.Spec] American National Standards Institute, "Telecommunications- Integrated Services Digital Network (ISDN) - Multi-Level Precedence and Pre-emption (MLPP) Service Capability", ANSI T1.619-1992 (R1999), 1992.

[ANSI.MLPP.Supplement] American National Standards Institute, "MLPP Service Domain Cause Value Changes", ANSI ANSI T1.619a-1994 (R1999), 1990.

[AVQ] S. Kunniyur and R. Srikant "Analysis and Design of an Adaptive Virtual Queue (AVQ) Algorithm for Active Queue Management", In: Proc. ACM SIGCOMM'01, Computer Communication Review 31 (4) (October, 2001).

[Breslau99] L. Breslau, S. Jamin, S. Shenker "Measurement-based admission control: what is the research agenda?", In: Proc. Int'l Workshop on Quality of Service 1999.

[Breslau00] L. Breslau, E. Knightly, S. Shenker, I. Stoica, H. Zhang "Endpoint Admission Control: Architectural Issues and Performance", In: ACM SIGCOMM 2000

[Briscoe] Bob Briscoe and Steve Rudkin, "Commercial Models for IP Quality of Service Interconnect", BT Technology Journal, Vol 23 No 2, April 2005.

[DCAC] Richard J. Gibbens and Frank P. Kelly "Distributed connection acceptance control for a connectionless network", In: Proc. International Teletraffic Congress (ITC16), Edinburgh, pp. 941 952 (1999).

[ECN-MPLS] Bruce Davie, Bob Briscoe, June Tay, "Explicit Congestion Marking in MPLS", draft-davie-ecn-mpls-00.txt (work in progress), June 2006

[EMERG-RQTS] Carlberg, K. and R. Atkinson, "General Requirements for Emergency Telecommunication Service (ETS)", RFC 3689, February 2004.

[EMERG-TEL] Carlberg, K. and R. Atkinson, "IP Telephony Requirements for Emergency Telecommunication Service (ETS)", RFC 3690, February 2004.

[Floyd] S. Floyd, 'Specifying Alternate Semantics for the Explicit Congestion Notification (ECN) Field', draft-floyd-ecn-alternates-02.txt (work in progress), August 2005

[GSPa] Karsten (Ed.), Martin "GSP/ECN Technology & Experiments", Deliverable: 15.3 PtIII, M3I Eu Vth Framework Project IST-1999-11429, URL: http://www.m3i.org/ (February, 2002) (superseded by [GSP-TR])

[GSP-TR] Martin Karsten and Jens Schmitt, "Admission Control Based on Packet Marking and Feedback Signalling -- Mechanisms, Implementation and Experiments", TU-Darmstadt Technical Report TR-KOM-2002-03, URL: http://www.kom.e-technik.tu-darmstadt.de/publications/abstracts/KS02-5.html (May, 2002)

[ITU.MLPP.1990] International Telecommunications Union, "Multilevel Precedence and Pre-emption Service (MLPP)", ITU-T Recommendation I.255.3, 1990.

[Johnson] DM Johnson, 'QoS control versus generous dimensioning', BT Technology Journal, Vol 23 No 2, April 2005

[LoadBalancing-a] Ruediger Martin, Michael Menth, and Michael Hemmkeppler: "Accuracy and Dynamics of Hash-Based Load Balancing Algorithms for Multipath Internet Routing", IEEE Broadnets, San Jose, CA, USA, October 2006 http://www3.informatik.uni-wuerzburg.de/~menth/Publications/Menth06p.pdf

[LoadBalancing-b] Ruediger Martin, Michael Menth, and Michael Hemmkeppler: "Accuracy and Dynamics of Multi-Stage Load Balancing for Multipath Internet Routing", currently under submission http://www3.informatik.uni-wuerzburg.de/~menth/Publications/Menth07-Sub-6.pdf

[Low] S. Low, L. Andrew, B. Wydrowski, 'Understanding XCP: equilibrium and fairness', IEEE InfoCom 2005

[NAC-a] Michael Menth: "Efficient Admission Control and Routing in Resilient Communication Networks", PhD thesis, July 2004, http://opus.bibliothek.uni-wuerzburg.de/opus/volltexte/2004/994/pdf/Menth04.pdf

[NAC-b] Michael Menth, Stefan Kopf, Joachim Charzinski, and Karl Schrodi: "Resilient Network Admission Control", currently under submission. http://www3.informatik.uni-wuerzburg.de/~menth/Publications/Menth07-Sub-3.pdf

[PCN] B. Briscoe, P. Eardley, D. Songhurst, F. Le Faucheur, A. Charny, V. Liatsos, S. Dudley, J. Babiarz, K. Chan, G. Karagiannis, A. Bader, L. Westberg. 'Pre-Congestion Notification marking', draft-briscoe-tsvwg-cl-phb-02 (work in progress), June 2006.

[Re-ECN] Bob Briscoe, Arnaud Jacquet, Alessandro Salvatori, 'Re-ECN: Adding Accountability for Causing Congestion to TCP/IP', draft-briscoe-tsvwg-re-ecn-tcp-01 (work in progress), March 2006.

[Re-feedback] Bob Briscoe, Arnaud Jacquet, Carla Di Cairano-Gilfedder, Andrea Soppera, 'Re-feedback for Policing Congestion Response in an Inter-network', ACM SIGCOMM 2005, August 2005.

[Re-PCN] B. Briscoe, 'Emulating Border Flow Policing using Re-ECN on Bulk Data', draft-briscoe-tsvwg-re-ecn-border-cheat-00 (work in progress), February 2006.

[Reid] ABD Reid, 'Economics and scalability of QoS solutions', BT Technology Journal, Vol 23 No 2, April 2005

[RFC2211] J. Wroclawski, Specification of the Controlled-Load Network Element Service, September 1997

[RFC2309] Braden, B., et al., "Recommendations on Queue Management and Congestion Avoidance in the Internet", RFC 2309, April 1998.

[RFC2474] Nichols, K., Blake, S., Baker, F. and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers", RFC 2474, December 1998

[RFC2475] Blake, S., Black, D., Carlson, M., Davies, E., Wang, Z. and W. Weiss, 'A framework for Differentiated Services', RFC 2475, December 1998.

[RFC2597] Heinanen, J., Baker, F., Weiss, W. and J. Wrocklawski, "Assured Forwarding PHB Group", RFC 2597, June 1999.

[RFC2998] Bernet, Y., Yavatkar, R., Ford, P., Baker, F., Zhang, L., Speer, M., Braden, R., Davie, B., Wroclawski, J. and E. Felstaine, "A Framework for Integrated Services Operation Over DiffServ Networks", RFC 2998, November 2000.

[RFC3168] Ramakrishnan, K., Floyd, S. and D. Black "The Addition of Explicit Congestion Notification (ECN) to IP", RFC 3168, September 2001.

[RFC3246] B. Davie, A. Charny, J.C.R. Bennet, K. Benson, J.Y. Le Boudec, W. Courtney, S. Davari, V. Firoiu, D. Stiliadis, 'An Expedited Forwarding PHB (Per-Hop Behavior)', RFC 3246, March 2002.

[RFC3270] Le Faucheur, F., Wu, L., Davie, B., Davari, S., Vaananen, P., Krishnan, R., Cheval, P., and J. Heinanen, "Multi- Protocol Label Switching (MPLS) Support of Differentiated Services", RFC 3270, May 2002.

[RFC4542] F. Baker & J. Polk, "Implementing an Emergency Telecommunications Service for Real Time Services in the Internet Protocol Suite", RFC 4542, May 2006.

[RMD] Attila Bader, Lars Westberg, Georgios Karagiannis, Cornelia Kappler, Tom Phelan, 'RMD-QOSM - The Resource Management in DiffServ QoS model', draft-ietf-nsis-rmd-03 Work in Progress, June 2005.

[RSVP-PCN] Francois Le Faucheur, Anna Charny, Bob Briscoe, Philip Eardley, Joe Barbiaz, Kwok-Ho Chan, 'RSVP Extensions for Admission Control over DiffServ using Pre-Congestion Notification (PCN)', draft-lefaucheur-rsvp-ecn-01 (work in progress), June 2006.

[RSVP-PREEMPTION] Herzog, S., "Signaled Preemption Priority Policy Element", RFC 3181, October 2001.

[RSVP-EMERGENCY] Le Faucheur et al., RSVP Extensions for Emergency Services, draft-lefaucheur-emergency-rsvp-02.txt

[RTECN] Babiarz, J., Chan, K. and V. Firoiu, 'Congestion Notification Process for Real-Time Traffic', draft-babiarz-tsvwg-rtecn-04 Work in Progress, July 2005.

[RTECN-usage] Alexander, C., Ed., Babiarz, J. and J. Matthews, 'Admission Control Use Case for Real-time ECN', draft-alexander-rtecn-admission-control-use-case-00, Work in Progress, February 2005.

[Songhurst] David J. Songhurst, Philip Eardley, Bob Briscoe, Carla Di Cairano Gilfedder and June Tay, 'Guaranteed QoS Synthesis for Admission Control with Shared Capacity', BT Technical Report TR-CXR9-2006-001, Feb 2006, http://www.cs.ucl.ac.uk/staff/B.Briscoe/projects/ipe2eqos/gqs/papers/GQS_shared_tr.pdf

[vq] Costas Courcoubetis and Richard Weber "Buffer Overflow Asymptotics for a Switch Handling Many Traffic Sources" In: Journal Applied Probability 33 pp. 886--903 (1996).

Authors' Addresses

Bob Briscoe
BT Research
B54/77, Sirius House
Adastral Park
Martlesham Heath
Ipswich, Suffolk
IP5 3RE
United Kingdom
Email: bob.briscoe@bt.com

Dave Songhurst
BT Research
B54/69, Sirius House
Adastral Park
Martlesham Heath
Ipswich, Suffolk
IP5 3RE
United Kingdom
Email: dsonghurst@jungle.bt.co.uk

Philip Eardley
BT Research
B54/77, Sirius House
Adastral Park
Martlesham Heath
Ipswich, Suffolk
IP5 3RE
United Kingdom
Email: philip.eardley@bt.com

Francois Le Faucheur
Cisco Systems, Inc.
Village d'Entreprise Green Side - Batiment T3
400, Avenue de Roumanille
06410 Biot Sophia-Antipolis
France
Email: flefauch@cisco.com

Anna Charny
Cisco Systems
300 Apollo Drive
Chelmsford, MA 01824
USA
Email: acharny@cisco.com

Kwok Ho Chan
Nortel Networks
600 Technology Park Drive
Billerica, MA 01821
USA
Email: khchan@nortel.com

Jozef Z. Babiarz
Nortel Networks
3500 Carling Avenue
Ottawa, Ont K2H 8E9
Canada
Email: babiarz@nortel.com

Stephen Dudley
Nortel Networks
4001 E. Chapel Hill Nelson Highway
P.O. Box 13010, ms 570-01-0V8
Research Triangle Park, NC 27709
USA
Email: smdudley@nortel.com

Georgios Karagiannis
University of Twente
P.O. BOX 217
7500 AE Enschede,
The Netherlands
EMail: g.karagiannis@ewi.utwente.nl

Attila Báder
attila.bader@ericsson.com

Lars Westberg
Ericsson AB
SE-164 80 Stockholm
Sweden
EMail: Lars.Westberg@ericsson.com

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any Intellectual Property Rights or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; nor does it represent that it has made any independent effort to identify any such rights. Information on the procedures with respect to rights in RFC documents can be found in BCP 78 and BCP 79.

Copies of IPR disclosures made to the IETF Secretariat and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementers or users of this specification can be obtained from the IETF on-line IPR repository at http://www.ietf.org/ipr.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights that may cover technology that may be required to implement this standard. Please address the information to the IETF at ietf-ipr@ietf.org

Disclaimer of Validity

This document and the information contained herein are provided on an "AS IS" basis and THE CONTRIBUTOR, THE ORGANIZATION HE/SHE REPRESENTS OR IS SPONSORED BY (IF ANY), THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Copyright (C) The Internet Society (2006).

This document is subject to the rights, licenses and restrictions contained in BCP 78, and except as set forth therein, the authors retain all their rights.