Network Working Group P. Eardley, Ed.
Request for Comments: 5559 BT
Category: Informational June 2009

Pre-Congestion Notification (PCN) Architecture

Status of This Memo

Copyright Notice

Abstract

This document describes a general architecture for flow admission and termination based on pre-congestion information in order to protect the quality of service of established, inelastic flows within a single Diffserv domain.

Table of Contents

1. Introduction

1.1. Overview of PCN

The objective of Pre-Congestion Notification (PCN) is to protect the quality of service (QoS) of inelastic flows within a Diffserv domain in a simple, scalable, and robust fashion. Two mechanisms are used: admission control, to decide whether to admit or block a new flow request, and (in abnormal circumstances) flow termination, to decide whether to terminate some of the existing flows. To achieve this, the overall rate of PCN-traffic is metered on every link in the domain, and PCN packets are appropriately marked when certain configured rates are exceeded. These configured rates are below the rate of the link, thus providing notification to boundary nodes about overloads before any congestion occurs (hence, "Pre-Congestion Notification"). The level of marking allows boundary nodes to make decisions about whether to admit or terminate.

Within a PCN-domain, PCN-traffic is forwarded in a prioritised Diffserv traffic class. Every link in the PCN-domain is configured with two rates (PCN-threshold-rate and PCN-excess-rate). If the overall rate of PCN-traffic on a link exceeds a configured rate, then a PCN-interior-node marks PCN-packets appropriately. The PCN-egress-nodes use this information to make admission control and flow termination decisions. Flow admission control determines whether a new flow can be admitted without any impact, in normal circumstances, on the QoS of existing PCN-flows. However, in abnormal circumstances (for instance, a disaster affecting multiple nodes and causing traffic re-routes), the QoS on existing PCN-flows may degrade even though care was exercised when admitting those flows. The flow termination mechanism removes sufficient traffic in order to protect the QoS of the remaining PCN-flows. All PCN-boundary-nodes and PCN-interior-nodes are PCN-enabled and are trusted for correct PCN operation. PCN-ingress-nodes police arriving packets to check that they are part of an admitted PCN-flow that keeps within its agreed flowspec, and hence they maintain per-flow state. PCN-interior-nodes meter all PCN-traffic, and hence do not need to maintain any per-flow

state. Decisions about flow admission and termination are made for a particular pair of PCN-boundary-nodes, and hence PCN-egress-nodes must be able to identify which PCN-ingress-node sent each PCN-packet.

1.2. Example Use Case for PCN

This section outlines an end-to-end QoS scenario that uses the PCN mechanisms within one domain. The parts outside the PCN-domain are out of scope for PCN, but are included to help clarify how PCN could be used. Note that this section is only an example -- in particular, there are other possibilities (see Section 3) for how the PCN-boundary-nodes perform admission control and flow termination.

As a fundamental building block, each link of the PCN-domain operates the following. Please refer to [Eardley09] and Figure 1.

o A threshold meter and marker, which marks all PCN-packets if the rate of PCN-traffic is greater than a first configured rate, the PCN-threshold-rate. The admission control mechanism limits the PCN-traffic on each link to *roughly* its PCN-threshold-rate.

o An excess-traffic meter and marker, which marks a proportion of PCN-packets such that the amount marked equals the traffic rate in excess of a second configured rate, the PCN-excess-rate. The flow termination mechanism limits the PCN-traffic on each link to *roughly* its PCN-excess-rate.

Overall, the aim is to give an "early warning" of potential congestion before there is any significant build-up of PCN-packets in the queue on the link; we term this "Pre-Congestion Notification" by analogy with ECN (Explicit Congestion Notification, [RFC3168]). Note that the link only meters the bulk PCN-traffic (and not per flow).

```
                          ==   Metering &    ==
                          ==Marking behaviour==       ==PCN mechanisms==
                       ^
           Rate of     ^
      PCN-traffic on   |
     bottleneck link   |
                       |
                       |       Some pkts                  Terminate some
                       |  excess-traffic-marked           admitted flows
                       |           &                            &
                       |     Rest of pkts                Block new flows
                       |   threshold-marked
                       |
     PCN-excess-rate  -|------------------------------------------------
(=PCN-supportable-rate)|
                       |       All pkts                  Block new flows
                       |   threshold-marked
                       |
   PCN-threshold-rate -|------------------------------------------------
 (=PCN-admissible-rate)|
                       |        No pkts                  Admit new flows
                       |      PCN-marked
                       |
```

Figure 1: Example of how the PCN admission control and flow termination mechanisms operate as the rate of PCN-traffic increases.

The two forms of PCN-marking are indicated by setting the ECN and DSCP (Differentiated Services Codepoint [RFC2474]) fields to known values, which are configured for the domain. Thus, the PCN-egress-nodes can monitor the PCN-markings in order to measure the severity of pre-congestion. In addition, the PCN-ingress-nodes need to set the ECN and DSCP fields to that configured for an unmarked PCN-packet, and the PCN-egress-nodes need to revert to values appropriate outside the PCN-domain.

For admission control, we assume end-to-end RSVP (Resource Reservation Protocol) [RFC2205]) signalling in this example. The PCN-domain is a single RSVP hop. The PCN-domain operates Diffserv, and we assume that PCN-traffic is scheduled with the expedited forwarding (EF) per-hop behaviour [RFC3246]. Hence, the overall solution is in line with the "IntServ over Diffserv" framework defined in [RFC2998], as shown in Figure 2.

```
   ___    ___    _______________________________________    ____    ___
  |   |  |   |  | PCN-             PCN-            PCN- |  |    |  |   |
  |   |  |   |  |ingress         interior         egress|  |    |  |   |
  |   |  |   |  | -node           -nodes          -node |  |    |  |   |
  |   |  |   |  |-------+  +-------+  +-------+  +------|  |    |  |   |
  |   |  |   |  |       |  | PCN   |  | PCN   |  |      |  |    |  |   |
  |   |..|   |..|Ingress|..|meter &|..|meter &|..|Egress|..|    |..|   |
  |   |..|   |..|Policer|..|marker |..|marker |..|Meter |..|    |..|   |
  |   |  |   |  |-------+  +-------+  +-------+  +------|  |    |  |   |
  |   |  |   |  |  \                                 /  |  |    |  |   |
  |   |  |   |  |   \                               /   |  |    |  |   |
  |   |  |   |  |    \  PCN-feedback-information   /    |  |    |  |   |
  |   |  |   |  |     \  (for admission control)  /     |  |    |  |   |
  |   |  |   |  |      --<-----<----<----<-----<--      |  |    |  |   |
  |   |  |   |  |       PCN-feedback-information        |  |    |  |   |
  |   |  |   |  |        (for flow termination)         |  |    |  |   |
  |___|  |___|  |_______________________________________|  |____|  |___|

  Sx     Access               PCN-domain                   Access    Rx
  End    Network                                          Network   End
  Host                                                              Host
                  <---- signalling across PCN-domain--->
                (for admission control & flow termination)

  <-------------------end-to-end QoS signalling protocol--------------->
```

Figure 2: Example of possible overall QoS architecture.

A source wanting to start a new QoS flow sends an RSVP PATH message. Normal hop-by-hop IntServ [RFC1633] is used outside the PCN-domain (we assume successfully). The PATH message travels across the PCN-domain; the PCN-egress-node reads the PHOP (previous RSVP hop) object to discover the specific PCN-ingress-node for this flow. The RESV message travels back from the receiver, and triggers the PCN-egress-node to check what fraction of the PCN-traffic from the relevant PCN-ingress-node is currently being threshold-marked. It adds an object with this information onto the RESV message, and hence the PCN-ingress-node learns about the level of pre-congestion on the path. If this level is below some threshold, then the PCN-ingress-node admits the new flow into the PCN-domain. The RSVP message triggers the PCN-ingress-node to install two normal IntServ items: five-tuple information, so that it can subsequently identify data packets that are part of a previously admitted PCN-flow, and a traffic profile, so that it can police the flow to within its reservation. Similarly, the RSVP message triggers the PCN-egress-node to install five-tuple and PHOP information so that it can identify packets as part of a flow from a specific PCN-ingress-node.

The flow termination mechanism may happen when some abnormal circumstance causes a link to become so pre-congested that it excess-traffic-marks (and perhaps also drops) PCN-packets. In this example, when a PCN-egress-node observes such a packet, it then, with some probability, terminates this PCN-flow; the probability is configured low enough to avoid over termination and high enough to ensure rapid termination of enough flows. It also informs the relevant PCN-ingress-node so that it can block any further traffic on the terminated flow.

1.3. Applicability of PCN

Compared with alternative QoS mechanisms, PCN has certain advantages and disadvantages that will make it appropriate in particular scenarios. For example, compared with hop-by-hop IntServ [RFC1633], PCN only requires per-flow state at the PCN-ingress-nodes. Compared with the Diffserv architecture [RFC2475], an operator needs to be less accurate and/or conservative in its prediction of the traffic matrix. The Diffserv architecture's traffic-conditioning agreements are static and coarse; they are defined at subscription time and are used (for instance) to limit the total traffic at each ingress of the domain, regardless of the egress for the traffic. On the other hand, PCN firstly uses admission control based on measurements of the current conditions between the specific pair of PCN-boundary-nodes, and secondly, in case of a disaster, PCN protects the QoS of most flows by terminating a few selected ones.

PCN's admission control is a measurement-based mechanism. Hence, it assumes that the present is a reasonable prediction of the future: the network conditions are measured at the time of a new flow request, but the actual network performance must be acceptable during the call some time later. Hence, PCN is unsuitable in several circumstances:

- If the source adapts its bit rate dependent on the level of pre-congestion, because then the aggregate traffic might become unstable. The assumption in this document is that PCN-packets come from real-time applications generating inelastic traffic, such as the Controlled Load Service [RFC2211].

- If a potential bottleneck link has capacity for only a few flows, because then a new flow can move a link directly from no pre-congestion to being so overloaded that it has to drop packets. The assumption in this document is that this isn't a problem.

- If there is the danger of a "flash crowd", in which many admission requests arrive within the reaction time of PCN's admission mechanism, because then they all might get admitted and so

overload the network. The assumption in this document is that, if it is necessary, then flash crowds are limited in some fashion beyond the scope of this document, for instance by rate-limiting QoS requests.

The applicability of PCN is discussed further in Section 6.

1.4. Documents about PCN

The purpose of this document is to describe a general architecture for flow admission and termination based on (pre-)congestion information in order to protect the quality of service of flows within a Diffserv domain. This document describes the PCN architecture at a high level (Section 3) and in more detail (Section 4). It also defines some terminology, and provides considerations about operations, management, and security. Section 6 considers the applicability of PCN in more detail, covering its benefits, deployment scenarios, assumptions, and potential challenges. The Appendix covers some potential future work items.

Aspects of PCN are also documented elsewhere:

- Metering and marking: [Eardley09] standardises threshold metering and marking and excess-traffic metering and marking. A PCN-packet may be marked, depending on the metering results.

- Encoding: the "baseline" encoding is described in [Moncaster09-1], which standardises two PCN encoding states (PCN-marked and not PCN-marked), whilst (experimental) extensions to the baseline encoding can provide three encoding states (threshold-marked, excess-traffic-marked, or not PCN-marked), for instance, see [Moncaster09-2]. (There may be further encoding states as suggested in [Westberg08].) Section 3.6 considers the backwards compatibility of PCN encoding with ECN.

- PCN-boundary-node behaviour: how the PCN-boundary-nodes convert the PCN-markings into decisions about flow admission and flow termination, as described in Informational documents such as [Taylor09] and [Charny07-2]. The concept is that the standardised metering and marking by PCN-nodes allows several possible PCN-boundary-node behaviours. A number of possibilities are outlined in this document; detailed descriptions and comparisons are in [Charny07-1] and [Menth09-2].

- Signalling between PCN-boundary-nodes: signalling is needed to transport PCN-feedback-information between the PCN-boundary-nodes (in the example above, this is the fraction of traffic, between the pair of PCN-boundary-nodes, that is PCN-marked). The exact

details vary for different PCN-boundary-node behaviours, and so should be described in those documents. It may require an extension to the signalling protocol -- standardisation is out of scope of the PCN WG.

- The interface by which the PCN-boundary-nodes learn identification information about the admitted flows: the exact requirements vary for different PCN-boundary-node behaviours and for different signalling protocols, and so should be described in those documents. They will be similar to those described in the example above -- a PCN-ingress-node needs to be able to identify that a packet is part of a previously admitted flow (typically from its five-tuple) and each PCN-boundary-node needs to be able to identify the other PCN-boundary-node for the flow.

2. Terminology

- PCN-domain: a PCN-capable domain; a contiguous set of PCN-enabled nodes that perform Diffserv scheduling [RFC2474]; the complete set of PCN-nodes that in principle can, through PCN-marking packets, influence decisions about flow admission and termination for the PCN-domain; includes the PCN-egress-nodes, which measure these PCN-marks, and the PCN-ingress-nodes.

- PCN-boundary-node: a PCN-node that connects one PCN-domain to a node either in another PCN-domain or in a non-PCN-domain.

- PCN-interior-node: a node in a PCN-domain that is not a PCN-boundary-node.

- PCN-node: a PCN-boundary-node or a PCN-interior-node.

- PCN-egress-node: a PCN-boundary-node in its role in handling traffic as it leaves a PCN-domain.

- PCN-ingress-node: a PCN-boundary-node in its role in handling traffic as it enters a PCN-domain.

- PCN-traffic, PCN-packets, PCN-BA: a PCN-domain carries traffic of different Diffserv behaviour aggregates (BAs) [RFC2474]. The PCN-BA uses the PCN mechanisms to carry PCN-traffic, and the corresponding packets are PCN-packets. The same network will carry traffic of other Diffserv BAs. The PCN-BA is distinguished by a combination of the Diffserv codepoint (DSCP) and ECN fields.

- PCN-flow: the unit of PCN-traffic that the PCN-boundary-node admits (or terminates); the unit could be a single microflow (as defined in [RFC2474]) or some identifiable collection of microflows.

- Pre-congestion: a condition of a link within a PCN-domain such that the PCN-node performs PCN-marking, in order to provide an "early warning" of potential congestion before there is any significant build-up of PCN-packets in the real queue. (Hence, by analogy with ECN, we call our mechanism Pre-Congestion Notification.)

- PCN-marking: the process of setting the header in a PCN-packet based on defined rules, in reaction to pre-congestion; either threshold-marking or excess-traffic-marking. Such a packet is then called PCN-marked.

- Threshold-metering: a metering behaviour that, if the PCN-traffic exceeds the PCN-threshold-rate, indicates that all PCN-traffic is to be threshold-marked.

- PCN-threshold-rate: the reference rate of a threshold-meter, which is configured for each link in the PCN-domain and which is lower than the PCN-excess-rate.

- Threshold-marking: the setting of the header in a PCN-packet to a specific encoding, based on indications from the threshold-meter. Such a packet is then called threshold-marked.

- Excess-traffic-metering: a metering behaviour that, if the PCN-traffic exceeds the PCN-excess-rate, indicates that the amount of PCN-traffic to be excess-traffic-marked is equal to the amount in excess of the PCN-excess-rate.

- PCN-excess-rate: the reference rate of an excess-traffic-meter, which is a configured for each link in the PCN-domain and which is higher than the PCN-threshold-rate.

- Excess-traffic-marking: the setting of the header in a PCN-packet to a specific encoding, based on indications from the excess-traffic-meter. Such a packet is then called excess-traffic-marked.

- PCN-colouring: the process of setting the header in a PCN-packet by a PCN-boundary-node; performed by a PCN-ingress-node so that PCN-nodes can easily identify PCN-packets; performed by a PCN-egress-node so that the header is appropriate for nodes beyond the PCN-domain.

- Ingress-egress-aggregate: The collection of PCN-packets from all PCN-flows that travel in one direction between a specific pair of PCN-boundary-nodes.

- PCN-feedback-information: information signalled by a PCN-egress-node to a PCN-ingress-node (or a central control node), which is needed for the flow admission and flow termination mechanisms.

- PCN-admissible-rate: the rate of PCN-traffic on a link up to which PCN admission control should accept new PCN-flows.

- PCN-supportable-rate: the rate of PCN-traffic on a link down to which PCN flow termination should, if necessary, terminate already admitted PCN-flows.

3. High-Level Functional Architecture

The high-level approach is to split functionality between:

- PCN-interior-nodes "inside" the PCN-domain, which monitor their own state of pre-congestion and mark PCN-packets as appropriate. They are not flow-aware, nor are they aware of ingress-egress-aggregates. The functionality is also done by PCN-ingress-nodes for their outgoing interfaces (ie, those "inside" the PCN-domain).

- PCN-boundary-nodes at the edge of the PCN-domain, which control admission of new PCN-flows and termination of existing PCN-flows, based on information from PCN-interior-nodes. This information is in the form of the PCN-marked data packets (which are intercepted by the PCN-egress-nodes) and is not in signalling messages. Generally, PCN-ingress-nodes are flow-aware.

The aim of this split is to keep the bulk of the network simple, scalable, and robust, whilst confining policy, application-level, and security interactions to the edge of the PCN-domain. For example, the lack of flow awareness means that the PCN-interior-nodes don't care about the flow information associated with PCN-packets, nor do the PCN-boundary-nodes care about which PCN-interior-nodes its ingress-egress-aggregates traverse.

In order to generate information about the current state of the PCN-domain, each PCN-node PCN-marks packets if it is "pre-congested". Exactly when a PCN-node decides if it is "pre-congested" (the algorithm) and exactly how packets are "PCN-marked" (the encoding) will be defined in separate Standards Track documents, but at a high level it is as follows:

- the algorithms: a PCN-node meters the amount of PCN-traffic on each one of its outgoing (or incoming) links. The measurement is made as an aggregate of all PCN-packets, not per flow. There are two algorithms: one for threshold-metering and one for excess-traffic-metering. The meters trigger PCN-marking as necessary.

- the encoding(s): a PCN-node PCN-marks a PCN-packet by modifying a combination of the DSCP and ECN fields. In the "baseline" encoding [Moncaster09-1], the ECN field is set to 11 and the DSCP is not altered. Extension encodings may be defined that, at most, use a second DSCP (eg, as in [Moncaster09-2]) and/or set the ECN field to values other than 11 (eg, as in [Menth08-2]).

In a PCN-domain, the operator may have two or three encoding states available. The baseline encoding provides two encoding states (not PCN-marked and PCN-marked), whilst extended encodings can provide three encoding states (not PCN-marked, threshold-marked, and excess-traffic-marked).

An operator may choose to deploy either admission control or flow termination or both. Although designed to work together, they are independent mechanisms, and the use of one does not require or prevent the use of the other. Three encoding states naturally allows both flow admission and flow termination. If there are only two encoding states, then there are several options -- see Section 3.3.

The PCN-boundary-nodes monitor the PCN-marked packets in order to extract information about the current state of the PCN-domain. Based on this monitoring, a distributed decision is made about whether to admit a prospective new flow or terminate existing flow(s). Sections 4.4 and 4.5 mention various possibilities for how the functionality could be distributed.

PCN-metering and PCN-marking need to be configured on all (potentially pre-congested) links in the PCN-domain to ensure that the PCN mechanisms protect all links. The actual functionality can be configured on the outgoing or incoming interfaces of PCN-nodes -- or one algorithm could be configured on the outgoing interface and the other on the incoming interface. The important point is that a consistent choice is made across the PCN-domain to ensure that the PCN mechanisms protect all links. See [Eardley09] for further discussion.

The objective of threshold-marking, as triggered by the threshold-metering algorithm, is to threshold-mark all PCN-packets whenever the bit rate of PCN-packets is greater than some configured rate, the PCN-threshold-rate. The objective of excess-traffic-metering, as triggered by the excess-traffic-marking algorithm, is to excess-

traffic-mark PCN-packets at a rate equal to the difference between the bit rate of PCN-packets and some configured rate, the PCN-excess-rate. Note that this description reflects the overall intent of the algorithms rather than their instantaneous behaviour, since the rate measured at a particular moment depends on the detailed algorithm, its implementation, and the traffic's variance as well as its rate (eg, marking may well continue after a recent overload, even after the instantaneous rate has dropped). The algorithms are specified in [Eardley09].

Admission and termination approaches are detailed and compared in [Charny07-1] and [Menth09-2]. The discussion below is just a brief summary. Sections 3.1 and 3.2 assume there are three encoding states available, whilst Section 3.3 assumes there are two encoding states available.

From the perspective of the outside world, a PCN-domain essentially looks like a Diffserv domain, but without the Diffserv architecture's traffic-conditioning agreements. PCN-traffic is either transported across it transparently or policed at the PCN-ingress-node (ie, dropped or carried at a lower QoS). One difference is that PCN-traffic has better QoS guarantees than normal Diffserv traffic because the PCN mechanisms better protect the QoS of admitted flows. Another difference may occur in the rare circumstance when there is a failure: on the one hand, some PCN-flows may get terminated but, on the other hand, other flows will get their QoS restored. Non-PCN-traffic is treated transparently, ie, the PCN-domain is a normal Diffserv domain.

3.1. Flow Admission

The objective of PCN's flow admission control mechanism is to limit the PCN-traffic on each link in the PCN-domain to *roughly* its PCN-admissible-rate by admitting or blocking prospective new flows, in order to protect the QoS of existing PCN-flows. With three encoding states available, the PCN-threshold-rate is configured by the operator as equal to the PCN-admissible-rate on each link. It is set lower than the traffic rate at which the link becomes congested and the node drops packets.

Exactly how the admission control decision is made will be defined separately in Informational documents. This document describes two approaches (others might be possible):

o The PCN-egress-node measures (possibly as a moving average) the fraction of the PCN-traffic that is threshold-marked. The fraction is measured for a specific ingress-egress-aggregate. If the fraction is below a threshold value, then the new flow is

admitted; if the fraction is above the threshold value, then it is blocked. The fraction could be measured as an EWMA (exponentially weighted moving average), which has sometimes been called the "congestion level estimate".

- The PCN-egress-node monitors PCN-traffic and if it receives one (or several) threshold-marked packets, then the new flow is blocked; otherwise, it is admitted. One possibility may be to react to the marking state of an initial flow-setup packet (eg, RSVP PATH). Another is that after one (or several) threshold-marks, all flows are blocked until after a specific period of no congestion.

Note that the admission control decision is made for a particular pair of PCN-boundary-nodes. So it is quite possible for a new flow to be admitted between one pair of PCN-boundary-nodes, whilst at the same time another admission request is blocked between a different pair of PCN-boundary-nodes.

3.2. Flow Termination

The objective of PCN's flow termination mechanism is to limit the PCN-traffic on each link to *roughly* its PCN-supportable-rate, by terminating some existing PCN-flows, in order to protect the QoS of the remaining PCN-flows. With three encoding states available, the PCN-excess-rate is configured by the operator as equal to the PCN-supportable-rate on each link. It may be set lower than the traffic rate at which the link becomes congested and at which the node drops packets.

Exactly how the flow termination decision is made will be defined separately in Informational documents. This document describes several approaches (others might be possible):

- In one approach, the PCN-egress-node measures the rate of PCN-traffic that is not excess-traffic-marked, which is the amount of PCN-traffic that can actually be supported, and communicates this to the PCN-ingress-node. Also, the PCN-ingress-node measures the rate of PCN-traffic that is destined for this specific PCN-egress-node. The difference represents the excess amount that should be terminated.

- Another approach instead measures the rate of excess-traffic-marked traffic and terminates this amount of traffic. This terminates less traffic than the previous approach, if some nodes are dropping PCN-traffic.

- Another approach monitors PCN-packets and terminates some of the PCN-flows that have an excess-traffic-marked packet. (If all such flows were terminated, far too much traffic would be terminated, so a random selection needs to be made from those with an excess-traffic-marked packet [Menth08-1].)

Since flow termination is designed for "abnormal" circumstances, it is quite likely that some PCN-nodes are congested and, hence, that packets are being dropped and/or significantly queued. The flow termination mechanism must accommodate this.

Note also that the termination control decision is made for a particular pair of PCN-boundary-nodes. So it is quite possible for PCN-flows to be terminated between one pair of PCN-boundary-nodes, whilst at the same time none are terminated between a different pair of PCN-boundary-nodes.

3.3. Flow Admission and/or Flow Termination When There Are Only Two PCN Encoding States

If a PCN-domain has only two encoding states available (PCN-marked and not PCN-marked), ie, it is using the baseline encoding [Moncaster09-1], then an operator has three options (others might be possible):

- admission control only: PCN-marking means threshold-marking, ie, only the threshold-metering algorithm triggers PCN-marking. Only PCN admission control is available.

- flow termination only: PCN-marking means excess-traffic-marking, ie, only the excess-traffic-metering algorithm triggers PCN-marking. Only PCN termination control is available.

- both admission control and flow termination: only the excess-traffic-metering algorithm triggers PCN-marking; however, the configured rate (PCN-excess-rate) is set equal to the PCN-admissible-rate, as shown in Figure 3. [Charny07-2] describes how both admission control and flow termination can be triggered in this case and also gives some pros and cons of this approach. The main downside is that admission control is less accurate.

```
                          ==   Metering &    ==
                          ==Marking behaviour==       ==PCN mechanisms==
                       ^
           Rate of     ^
      PCN-traffic on   |
     bottleneck link   |                                  Terminate some
                       |                                  admitted flows
                       |                                         &
                       |                                 Block new flows
                       |
                       |       Some pkts
   U*PCN-excess-rate  -|  excess-traffic-marked        -----------------
(=PCN-supportable-rate)|
                       |                                 Block new flows
                       |
                       |
     PCN-excess-rate  -|------------------------------------------------
 (=PCN-admissible-rate)|
                       |         No pkts                 Admit new flows
                       |       PCN-marked
                       |
```

Figure 3: Schematic of how the PCN admission control and flow termination mechanisms operate as the rate of PCN-traffic increases, for a PCN-domain with two encoding states and using the approach of [Charny07-2]. Note: U is a global parameter for all links in the PCN-domain.

3.4. Information Transport

The transport of pre-congestion information from a PCN-node to a PCN-egress-node is through PCN-markings in data packet headers, ie, "in-band"; no signalling protocol messaging is needed. Signalling is needed to transport PCN-feedback-information -- for example, to convey the fraction of PCN-marked traffic from a PCN-egress-node to the relevant PCN-ingress-node. Exactly what information needs to be transported will be described in future documents about possible boundary mechanisms. The signalling could be done by an extension of RSVP or NSIS (Next Steps in Signalling), for instance; [Lefaucheur06] describes the extensions needed for RSVP.

3.5. PCN-Traffic

The following are some high-level points about how PCN works:

- o There needs to be a way for a PCN-node to distinguish PCN-traffic from other traffic. This is through a combination of the DSCP field and/or ECN field.

- It is not advised to have competing-non-PCN-traffic but, if there is such traffic, there needs to be a mechanism to limit it. "Competing-non-PCN-traffic" means traffic that shares a link with PCN-traffic and competes for its forwarding bandwidth. Hence, more competing-non-PCN-traffic results in poorer QoS for PCN. Further, the unpredictable amount of competing-non-PCN-traffic makes the PCN mechanisms less accurate and so reduces PCN's ability to protect the QoS of admitted PCN-flows.

- Two examples of such competing-non-PCN-traffic are:

  1. traffic that is priority scheduled over PCN (perhaps a particular application or an operator's control messages);

  2. traffic that is scheduled at the same priority as PCN (for example, if the Voice-Admit codepoint is used for PCN-traffic [Moncaster09-1] and there is non-PCN, voice-admit traffic in the PCN-domain).

- If there is such competing-non-PCN-traffic, then PCN's mechanisms should take account of it, in order to improve the accuracy of the decision about whether to admit (or terminate) a PCN-flow. For example, one mechanism is that such competing-non-PCN-traffic contributes to the PCN-meters (ie, is metered by the threshold-marking and excess-traffic-marking algorithms).

- There will be other non-PCN-traffic that doesn't compete for the same forwarding bandwidth as PCN-traffic, because it is forwarded at lower priority. Hence, it shouldn't contribute to the PCN-meters. Examples are best-effort and assured-forwarding traffic. However, a PCN-node should dedicate some capacity to lower-priority traffic so that it isn't starved.

- This document assumes that the PCN mechanisms are applied to a single behaviour aggregate in the PCN-domain. However, it would also be possible to apply them independently to more than one behaviour aggregate, which are distinguished by DSCP.

3.6. Backwards Compatibility

PCN specifies semantics for the ECN field that differ from the default semantics of [RFC3168]. A particular PCN encoding scheme needs to describe how it meets the guidelines of BCP 124 [RFC4774] for specifying alternative semantics for the ECN field. In summary, the approach is to:

- use a DSCP to allow PCN-nodes to distinguish PCN-traffic that uses the alternative ECN semantics;

- define these semantics for use within a controlled region, the PCN-domain;

- take appropriate action if ECN-capable, non-PCN-traffic arrives at a PCN-ingress-node with the DSCP used by PCN.

For the baseline encoding [Moncaster09-1], the "appropriate action" is to block ECN-capable traffic that uses the same DSCP as PCN from entering the PCN-domain directly. "Blocking" means it is dropped or downgraded to a lower-priority behaviour aggregate, or alternatively such traffic may be tunnelled through the PCN-domain. The reason that "appropriate action" is needed is that the PCN-egress-node clears the ECN field to 00.

Extended encoding schemes may need to take different "appropriate action".

4. Detailed Functional Architecture

This section is intended to provide a systematic summary of the new functional architecture in the PCN-domain. First, it describes functions needed at the three specific types of PCN-node; these are data plane functions and are in addition to the normal router functions for PCN-nodes. Then, it describes the further functionality needed for both flow admission control and flow termination; these are signalling and decision-making functions, and there are various possibilities for where the functions are physically located. The section is split into:

1. functions needed at PCN-interior-nodes

2. functions needed at PCN-ingress-nodes

3. functions needed at PCN-egress-nodes

4. other functions needed for flow admission control

5. other functions needed for flow termination control

Note: Probing is covered in the Appendix.

The section then discusses some other detailed topics:

1. addressing

2. tunnelling

3. fault handling

4.1. PCN-Interior-Node Functions

Each link of the PCN-domain is configured with the following functionality:

- Behaviour aggregate classification - determine whether or not an incoming packet is a PCN-packet.

- PCN-meter - measure the "amount of PCN-traffic". The measurement is made on the overall PCN-traffic, not per flow. Algorithms determine whether to indicate to the PCN-marking functionality that packets should be PCN-marked.

- PCN-mark - as triggered by indications from the PCN-meter functionality; if necessary, PCN-mark packets with the appropriate encoding.

- Drop - if the queue overflows, then naturally packets are dropped. In addition, the link may be configured with a maximum rate for PCN-traffic (below the physical link rate), above which PCN-packets are dropped.

The functions are defined in [Eardley09] and the baseline encoding in [Moncaster09-1] (extended encodings are to be defined in other documents).

```
                                       +---------+   Result
                                    +->|Threshold|-------+
                                    |  |  Meter  |       |
                                    |  +---------+       V
         +----------+   +- - - - -+  |                +------+
         |   BA     |   |         |  |                |      |    Marked
Packet =>|Classifier|==>| Dropper |==?===============>|Marker|==> Packet
Stream   |          |   |         |  |                |      |    Stream
         +----------+   +- - - - -+  |                +------+
                                    |  +---------+       ^
                                    |  | Excess  |       |
                                    +->| Traffic |-------+
                                       |  Meter  |   Result
                                       +---------+
```

Figure 4: Schematic of PCN-interior-node functionality.

4.2. PCN-Ingress-Node Functions

Each ingress link of the PCN-domain is configured with the following functionality:

- Packet classification - determine whether an incoming packet is part of a previously admitted flow by using a filter spec (eg, DSCP, source and destination addresses, port numbers, and protocol).

- Police - police, by dropping any packets received with a DSCP indicating PCN transport that do not belong to an admitted flow. (A prospective PCN-flow that is rejected could be blocked or admitted into a lower-priority behaviour aggregate.) Similarly, police packets that are part of a previously admitted flow, to check that the flow keeps to the agreed rate or flowspec (eg, see [RFC1633] for a microflow and its NSIS equivalent).

- PCN-colour - set the DSCP and ECN fields appropriately for the PCN-domain, for example, as in [Moncaster09-1].

- Meter - some approaches to flow termination require the PCN-ingress-node to measure the (aggregate) rate of PCN-traffic towards a particular PCN-egress-node.

The first two are policing functions, needed to make sure that PCN-packets admitted into the PCN-domain belong to a flow that has been admitted and to ensure that the flow keeps to the flowspec agreed (eg, doesn't exceed an agreed maximum rate and is inelastic traffic). Installing the filter spec will typically be done by the signalling protocol, as will re-installing the filter, for example, after a re-route that changes the PCN-ingress-node (see [Briscoe06] for an example using RSVP). PCN-colouring allows the rest of the PCN-domain to recognise PCN-packets.

4.3. PCN-Egress-Node Functions

Each egress link of the PCN-domain is configured with the following functionality:

- Packet classify - determine which PCN-ingress-node a PCN-packet has come from.

- Meter - "measure PCN-traffic" or "monitor PCN-marks".

- PCN-colour - for PCN-packets, set the DSCP and ECN fields to the appropriate values for use outside the PCN-domain.

The metering functionality, of course, depends on whether it is targeted at admission control or flow termination. Alternatives involve the PCN-egress-node "measuring", as an aggregate (ie, not per flow), all PCN-packets from a particular PCN-ingress-node, or "monitoring" the PCN-traffic and reacting to one (or several) PCN-

marked packets. For PCN-colouring, [Moncaster09-1] specifies that the PCN-egress-node resets the ECN field to 00; other encodings may define different behaviour.

4.4. Admission Control Functions

As well as the functions covered above, other specific admission control functions need to be performed (others might be possible):

- Make decision about admission - based on the output of the PCN-egress-node's meter function. In the case where it "measures PCN-traffic", the measured traffic on the ingress-egress-aggregate is compared with some reference level. In the case where it "monitors PCN-marks", the decision is based on whether or not one (or several) packets are PCN-marked (eg, the RSVP PATH message). In either case, the admission decision also takes account of policy and application-layer requirements [RFC2753].

- Communicate decision about admission - signal the decision to the node making the admission control request (which may be outside the PCN-domain) and to the policer (PCN-ingress-node function) for enforcement of the decision.

There are various possibilities for how the functionality could be distributed (we assume the operator will configure which is used):

- The decision is made at the PCN-egress-node and the decision (admit or block) is signalled to the PCN-ingress-node.

- The decision is recommended by the PCN-egress-node (admit or block), but the decision is definitively made by the PCN-ingress-node. The rationale is that the PCN-egress-node naturally has the necessary information about the amount of PCN-marks on the ingress-egress-aggregate, whereas the PCN-ingress-node is the policy enforcement point [RFC2753] that polices incoming traffic to ensure it is part of an admitted PCN-flow.

- The decision is made at the PCN-ingress-node, which requires that the PCN-egress-node signals PCN-feedback-information to the PCN-ingress-node. For example, it could signal the current fraction of PCN-traffic that is PCN-marked.

- The decision is made at a centralised node (see Appendix).

Note: Admission control functionality is not performed by normal PCN-interior-nodes.

4.5. Flow Termination Functions

As well as the functions covered above, other specific termination control functions need to be performed (others might be possible):

- PCN-meter at PCN-egress-node - similarly to flow admission, there are two types of possibilities: to "measure PCN-traffic" on the ingress-egress-aggregate, or to "monitor PCN-marks" and react to one (or several) PCN-marks.

- (if required) PCN-meter at PCN-ingress-node - make "measurements of PCN-traffic" being sent towards a particular PCN-egress-node; again, this is done for the ingress-egress-aggregate and not per flow.

- (if required) Communicate PCN-feedback-information to the node that makes the flow termination decision - for example, as in [Briscoe06], communicate the PCN-egress-node's measurements to the PCN-ingress-node.

- Make decision about flow termination - use the information from the PCN-meter(s) to decide which PCN-flow or PCN-flows to terminate. The decision takes account of policy and application-layer requirements [RFC2753].

- Communicate decision about flow termination - signal the decision to the node that is able to terminate the flow (which may be outside the PCN-domain) and to the policer (PCN-ingress-node function) for enforcement of the decision.

There are various possibilities for how the functionality could be distributed, similar to those discussed above in Section 4.4.

Note: Flow termination functionality is not performed by normal PCN-interior-nodes.

4.6. Addressing

PCN-nodes may need to know the address of other PCN-nodes. Note that PCN-interior-nodes don't need to know the address of other PCN-nodes (except their next-hop neighbours for routing purposes).

At a minimum, the PCN-egress-node needs to know the address of the PCN-ingress-node associated with a flow so that the PCN-ingress-node can be informed of the admission decision (and any flow termination decision) and enforce it through policing. There are various

possibilities for how the PCN-egress-node can do this, ie, associate the received packet to the correct ingress-egress-aggregate. It is not the intention of this document to mandate a particular mechanism.

- The addressing information can be gathered from signalling -- for example, through the regular processing of an RSVP PATH message, as the PCN-ingress-node is the previous RSVP hop (PHOP) ([Lefaucheur06]). Another option is that the PCN-ingress-node could signal its address to the PCN-egress-node.

- Always tunnel PCN-traffic across the PCN-domain. Then the PCN-ingress-node's address is simply the source address of the outer packet header. The PCN-ingress-node needs to learn the address of the PCN-egress-node, either by manual configuration or by one of the automated tunnel endpoint discovery mechanisms (such as signalling or probing over the data route, interrogating routing, or using a centralised broker).

4.7. Tunnelling

Tunnels may originate and/or terminate within a PCN-domain (eg, IP over IP, IP over MPLS). It is important that the PCN-marking of any packet can potentially influence PCN's flow admission control and termination -- it shouldn't matter whether the packet happens to be tunnelled at the PCN-node that PCN-marks the packet, or indeed whether it's decapsulated or encapsulated by a subsequent PCN-node. This suggests that the "uniform conceptual model" described in [RFC2983] should be re-applied in the PCN context. In line with both this and the approach of [RFC4303] and [Briscoe09], the following rule is applied if encapsulation is done within the PCN-domain:

- Any PCN-marking is copied into the outer header.

Note: A tunnel will not provide this behaviour if it complies with [RFC3168] tunnelling in either mode, but it will if it complies with [RFC4301] IPsec tunnelling.

Similarly, in line with the "uniform conceptual model" of [RFC2983], with the "full-functionality option" of [RFC3168], and with [RFC4301], the following rule is applied if decapsulation is done within the PCN-domain:

- If the outer header's marking state is more severe, then it is copied onto the inner header.

Note that the order of increasing severity is: not PCN-marked, threshold-marked, and excess-traffic-marked.

An operator may wish to tunnel PCN-traffic from PCN-ingress-nodes to PCN-egress-nodes. The PCN-marks shouldn't be visible outside the PCN-domain, which can be achieved by the PCN-egress-node doing the PCN-colouring function (Section 4.3) after all the other (PCN and tunnelling) functions. The potential reasons for doing such tunnelling are: the PCN-egress-node then automatically knows the address of the relevant PCN-ingress-node for a flow, and, even if ECMP (Equal Cost Multi-Path) is running, all PCN-packets on a particular ingress-egress-aggregate follow the same path (for more on ECMP, see Section 6.4). But such tunnelling also has drawbacks, for example, the additional overhead in terms of bandwidth and processing as well as the cost of setting up a mesh of tunnels between PCN-boundary-nodes (there is an N^2 scaling issue).

Potential issues arise for a "partially PCN-capable tunnel", ie, where only one tunnel endpoint is in the PCN-domain:

1. The tunnel originates outside a PCN-domain and ends inside it. If the packet arrives at the tunnel ingress with the same encoding as used within the PCN-domain to indicate PCN-marking, then this could lead the PCN-egress-node to falsely measure pre-congestion.

2. The tunnel originates inside a PCN-domain and ends outside it. If the packet arrives at the tunnel ingress already PCN-marked, then it will still have the same encoding when it's decapsulated, which could potentially confuse nodes beyond the tunnel egress.

In line with the solution for partially capable Diffserv tunnels in [RFC2983], the following rules are applied:

- For case (1), the tunnel egress node clears any PCN-marking on the inner header. This rule is applied before the "copy on decapsulation" rule above.

- For case (2), the tunnel ingress node clears any PCN-marking on the inner header. This rule is applied after the "copy on encapsulation" rule above.

Note that the above implies that one has to know, or determine, the characteristics of the other end of the tunnel as part of establishing it.

Tunnelling constraints were a major factor in the choice of the baseline encoding. As explained in [Moncaster09-1], with current tunnelling endpoints, only the 11 codepoint of the ECN field survives decapsulation, and hence the baseline encoding only uses the 11 codepoint to indicate PCN-marking. Extended encoding schemes need to

explain their interactions with (or assumptions about) tunnelling. A lengthy discussion of all the issues associated with layered encapsulation of congestion notification (for ECN as well as PCN) is in [Briscoe09].

4.8. Fault Handling

If a PCN-interior-node (or one of its links) fails, then lower-layer protection mechanisms or the regular IP routing protocol will eventually re-route around 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 that new flows will not be admitted until enough existing flows have departed. Re-routing may result in heavy (pre-)congestion, which will cause the flow termination mechanism to kick in.

If a PCN-boundary-node fails, then we would like the regular QoS signalling protocol to be responsible for taking appropriate action. As an example, [Briscoe09] considers what happens if RSVP is the QoS signalling protocol.

5. Operations and Management

This section considers operations and management issues, under the FCAPS headings: Faults, Configuration, Accounting, Performance, and Security. Provisioning is discussed with performance.

5.1. Fault Operations and Management

Fault Operations and Management is about preventing faults, telling the management system (or manual operator) that the system has recovered (or not) from a failure, and about maintaining information to aid fault diagnosis.

Admission blocking and, particularly, flow termination mechanisms should rarely be needed in practice. It would be unfortunate if they didn't work after an option had been accidentally disabled. Therefore, it will be necessary to regularly test that the live system works as intended (devising a meaningful test is left as an exercise for the operator).

Section 4 describes how the PCN architecture has been designed to ensure admitted flows continue gracefully after recovering automatically from link or node failures. The need to record and monitor re-routing events affecting signalling is unchanged by the

addition of PCN to a Diffserv domain. Similarly, re-routing events within the PCN-domain will be recorded and monitored just as they would be without PCN.

PCN-marking does make it possible to record "near-misses". For instance, at the PCN-egress-node a "reporting threshold" could be set to monitor how often -- and for how long -- the system comes close to triggering flow blocking without actually doing so. Similarly, bursts of flow termination marking could be recorded even if they are not sufficiently sustained to trigger flow termination. Such statistics could be correlated with per-queue counts of marking volume (Section 5.2) to upgrade resources in danger of causing service degradation or to trigger manual tracing of intermittent incipient errors that would otherwise have gone unnoticed.

Finally, of course, many faults are caused by failings in the management process ("human error"): a wrongly configured address in a node, a wrong address given in a signalling protocol, a wrongly configured parameter in a queueing algorithm, a node set into a different mode from other nodes, and so on. Generally, a clean design with few configurable options ensures this class of faults can be traced more easily and prevented more often. Sound management practice at run-time also helps. For instance, a management system should be used that constrains configuration changes within system rules (eg, preventing an option setting inconsistent with other nodes), configuration options should be recorded in an offline database, and regular automatic consistency checks between live systems and the database should be performed. PCN adds nothing specific to this class of problems.

5.2. Configuration Operations and Management

Threshold-metering and -marking and excess-traffic-metering and -marking are standardised in [Eardley09]. However, more diversity in PCN-boundary-node behaviours is expected, in order to interface with diverse industry architectures. It may be possible to have different PCN-boundary-node behaviours for different ingress-egress-aggregates within the same PCN-domain.

PCN-metering behaviour is enabled on either the egress or the ingress interfaces of PCN-nodes. A consistent choice must be made across the PCN-domain to ensure that the PCN mechanisms protect all links.

PCN configuration control variables fall into the following categories:

- system options (enabling or disabling behaviours)

- parameters (setting levels, addresses, etc.)

One possibility is that all configurable variables sit within an SNMP (Simple Network Management Protocol) management framework [RFC3411], being structured within a defined management information base (MIB) on each node, and being remotely readable and settable via a suitably secure management protocol (such as SNMPv3).

Some configuration options and parameters have to be set once to "globally" control the whole PCN-domain. Where possible, these are identified below. This may affect operational complexity and the chances of interoperability problems between equipment from different vendors.

It may be possible for an operator to configure some PCN-interior-nodes so that they don't run the PCN mechanisms, if it knows that these links will never become (pre-)congested.

5.2.1. System Options

On PCN-interior-nodes there will be very few system options:

- Whether two PCN-markings (threshold-marked and excess-traffic-marked) are enabled or only one. Typically, all nodes throughout a PCN-domain will be configured the same in this respect. However, exceptions could be made. For example, if most PCN-nodes used both markings but some legacy hardware was incapable of running two algorithms, an operator might be willing to configure these legacy nodes solely for excess-traffic-marking to enable flow termination as a back-stop. It would be sensible to place such nodes where they could be provisioned with a greater leeway over expected traffic levels.

- In the case where only one PCN-marking is enabled, all nodes must be configured to generate PCN-marks from the same meter (ie, either the threshold meter or the excess-traffic meter).

PCN-boundary-nodes (ingress and egress) will have more system options:

- Which of admission and flow termination are enabled. If any PCN-interior-node is configured to generate a marking, all PCN-boundary-nodes must be able to interpret that marking (which

includes understanding, in a PCN-domain that uses only one type of PCN-marking, whether they are generated by PCN-interior-nodes' threshold meters or their excess-traffic meters). Therefore, all PCN-boundary-nodes must be configured the same in this respect.

- Where flow admission and termination decisions are made: at PCN-ingress-nodes or at PCN-egress-nodes (or at a centralised node, see Appendix). Theoretically, this configuration choice could be negotiated for each pair of PCN-boundary-nodes, but we cannot imagine why such complexity would be required, except perhaps in future inter-domain scenarios.

- How PCN-markings are translated into admission control and flow termination decisions (see Sections 3.1 and 3.2).

PCN-egress-nodes will have further system options:

- How the mapping should be established between each packet and its aggregate (eg, by MPLS label and by IP packet filter spec) and how to take account of ECMP.

- If an equipment vendor provides a choice, there may be options for selecting which smoothing algorithm to use for measurements.

5.2.2. Parameters

Like any Diffserv domain, every node within a PCN-domain will need to be configured with the DSCP(s) used to identify PCN-packets. On each interior link, the main configuration parameters are the PCN-threshold-rate and PCN-excess-rate. A larger PCN-threshold-rate enables more PCN-traffic to be admitted on a link, hence improving capacity utilisation. A PCN-excess-rate set further above the PCN-threshold-rate allows greater increases in traffic (whether due to natural fluctuations or some unexpected event) before any flows are terminated, ie, minimises the chances of unnecessarily triggering the termination mechanism. For instance, an operator may want to design their network so that it can cope with a failure of any single PCN-node without terminating any flows.

Setting these rates on the first deployment of PCN will be very similar to the traditional process for sizing an admission-controlled network, depending on: the operator's requirements for minimising flow blocking (grade of service), the expected PCN-traffic load on each link and its statistical characteristics (the traffic matrix), contingency for re-routing the PCN-traffic matrix in the event of single or multiple failures, and the expected load from other classes relative to link capacities [Menth09-1]. But, once a domain is in operation, a PCN design goal is to be able to determine growth in

these configured rates much more simply, by monitoring PCN-marking rates from actual rather than expected traffic (see Section 5.4 on Performance and Provisioning).

Operators may also wish to configure a rate greater than the PCN-excess-rate that is the absolute maximum rate that a link allows for PCN-traffic. This may simply be the physical link rate, but some operators may wish to configure a logical limit to prevent starvation of other traffic classes during any brief period after PCN-traffic exceeds the PCN-excess-rate but before flow termination brings it back below this rate.

Threshold-metering requires a threshold token bucket depth to be configured, excess-traffic-metering requires a value for the MTU (maximum size of a PCN-packet on the link), and both require setting a maximum size of their token buckets. It is preferable to have rules that set defaults for these parameters but to then allow operators to change them -- for instance, if average traffic characteristics change over time.

The PCN-egress-node may allow configuration of:

o how it smooths metering of PCN-markings (eg, EWMA parameters)

Whichever node makes admission and flow termination decisions will contain algorithms for converting PCN-marking levels into admission or flow termination decisions. These will also require configurable parameters, for instance:

o An admission control algorithm that is based on the fraction of marked packets will at least require a marking threshold setting above which it denies admission to new flows.

o Flow termination algorithms will probably require a parameter to delay termination of any flows until it is more certain that an anomalous event is not transient.

o A parameter to control the trade-off between how quickly excess flows are terminated and over-termination.

One particular approach [Charny07-2] would require a global parameter to be defined on all PCN-nodes, but would only need one PCN-marking rate to be configured on each link. The global parameter is a scaling factor between admission and termination (the rate of PCN-traffic on a link up to which flows are admitted vs. the rate above which flows are terminated). [Charny07-2] discusses in full the impact of this particular approach on the operation of PCN.

5.3. Accounting Operations and Management

Accounting is only done at trust boundaries so it is out of scope of this document, which is confined to intra-domain issues. Use of PCN internal to a domain makes no difference to the flow signalling events crossing trust boundaries outside the PCN-domain, which are typically used for accounting.

5.4. Performance and Provisioning Operations and Management

Monitoring of performance factors measurable from *outside* the PCN-domain will be no different with PCN than with any other packet-based, flow admission control system, both at the flow level (blocking probability, etc.) and the packet level (jitter [RFC3393], [Y.1541], loss rate [RFC4656], mean opinion score [P.800], etc.). The difference is that PCN is intentionally designed to indicate *internally* which exact resource(s) are the cause of performance problems and by how much.

Even better, PCN indicates which resources will probably cause problems if they are not upgraded soon. This can be achieved by the management system monitoring the total amount (in bytes) of PCN-marking generated by each queue over a period. Given possible long provisioning lead times, pre-congestion volume is the best metric to reveal whether sufficient persistent demand has occurred to warrant an upgrade because, even before utilisation becomes problematic, the statistical variability of traffic will cause occasional bursts of pre-congestion. This "early warning system" decouples the process of adding customers from the provisioning process. This should cut the time to add a customer when compared against admission control that is provided over native Diffserv [RFC2998] because it saves having to verify the capacity-planning process before adding each customer.

Alternatively, before triggering an upgrade, the long-term pre-congestion volume on each link can be used to balance traffic load across the PCN-domain by adjusting the link weights of the routing system. When an upgrade to a link's configured PCN-rates is required, it may also be necessary to upgrade the physical capacity available to other classes. However, there will usually be sufficient physical capacity for the upgrade to go ahead as a simple configuration change. Alternatively, [Songhurst06] describes an adaptive rather than preconfigured system, where the configured PCN-threshold-rate is replaced with a high and low water mark and the marking algorithm automatically optimises how physical capacity is shared, using the relative loads from PCN and other traffic classes.

All the above processes require just three extra counters associated with each PCN queue: threshold-markings, excess-traffic-markings, and drops. Every time a PCN-packet is marked or dropped, its size in bytes should be added to the appropriate counter. Then the management system can read the counters at any time and subtract a previous reading to establish the incremental volume of each type of (pre-)congestion. Readings should be taken frequently so that anomalous events (eg, re-routes) can be distinguished from regular fluctuating demand, if required.

5.5. Security Operations and Management

Security Operations and Management is about using secure operational practices as well as being able to track security breaches or near-misses at run-time. PCN adds few specifics to the general good practice required in this field [RFC4778]. The correct functions of the system should be monitored (Section 5.4) in multiple independent ways and correlated to detect possible security breaches. Persistent (pre-)congestion marking should raise an alarm (both on the node doing the marking and on the PCN-egress-node metering it). Similarly, persistently poor external QoS metrics (such as jitter or mean opinion score) should raise an alarm. The following are examples of symptoms that may be the result of innocent faults, rather than attacks; however, until diagnosed, they should be logged and should trigger a security alarm:

- Anomalous patterns of non-conforming incoming signals and packets rejected at the PCN-ingress-nodes (eg, packets already marked PCN-capable or traffic persistently starving token bucket policers).

- PCN-capable packets arriving at a PCN-egress-node with no associated state for mapping them to a valid ingress-egress-aggregate.

- A PCN-ingress-node receiving feedback signals that are about the pre-congestion level on a non-existent aggregate or that are inconsistent with other signals (eg, unexpected sequence numbers, inconsistent addressing, conflicting reports of the pre-congestion level, etc.).

- Pre-congestion marking arriving at a PCN-egress-node with (pre-)congestion markings focused on particular flows, rather than randomly distributed throughout the aggregate.

6. Applicability of PCN

6.1. Benefits

The key benefits of the PCN mechanisms are that they are simple, scalable, and robust, because:

- Per-flow state is only required at the PCN-ingress-nodes ("stateless core"). This is required for policing purposes (to prevent non-admitted PCN-traffic from entering the PCN-domain) and so on. It is not generally required that other network entities are aware of individual flows (although they may be in particular deployment scenarios).

- Admission control is resilient: with PCN, QoS is decoupled from the routing system. Hence, in general, admitted flows can survive capacity, routing, or topology changes without additional signalling. The PCN-admissible-rate on each link can be chosen to be small enough that admitted traffic can still be carried after a re-routing in most failure cases [Menth09-1]. 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 [Iyer03].

- The PCN-metering behaviours only operate on the overall PCN-traffic on the link, not per flow.

- The information of these measurements is signalled to the PCN-egress-nodes by the PCN-marks in the packet headers, ie, "in-band". No additional signalling protocol is required for transporting the PCN-marks. Therefore, no secure binding is required between data packets and separate congestion messages.

- The PCN-egress-nodes make separate measurements, operating on the aggregate PCN-traffic from each PCN-ingress-node, ie, not per flow. Similarly, signalling by the PCN-egress-node of PCN-feedback-information (which is used for flow admission and termination decisions) is at the granularity of the ingress-egress-aggregate. An alternative approach is that the PCN-egress-nodes monitor the PCN-traffic and signal PCN-feedback-information (which is used for flow admission and termination decisions) at the granularity of one (or a few) PCN-marks.

- The admitted PCN-load is controlled dynamically. Therefore, it adapts as the traffic matrix changes. It also adapts 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 PCN-traffic matrix.

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

- Flow termination can also enable an operator to be less conservative when deploying network capacity. It is an alternative to running links at low utilisation in order to protect against link or node failures. This is especially the case with SRLGs (shared risk link groups), which are links that share a resource, such as a fibre, whose failure affects all links in that group [RFC4216]). Fully protecting traffic against a single SRLG failure requires low utilisation (~10%) of the link bandwidth on some links before failure [Charny08].

- The PCN-supportable-rate may be set below the maximum rate that PCN-traffic can be transmitted on a link in order to trigger the termination of some PCN-flows before loss (or excessive delay) of PCN-packets occurs, or to keep the maximum PCN-load on a link below a level configured by the operator.

- Provisioning of the network is decoupled from the process of adding new customers. By contrast, with the Diffserv architecture [RFC2475], operators rely on subscription-time Service Level Agreements, which statically define the parameters of the traffic that will be accepted from a customer. This way, the operator has to verify that provision is sufficient each time a new customer is added to check that the Service Level Agreement can be fulfilled. A PCN-domain doesn't need such traffic conditioning.

6.2. Deployment Scenarios

Operators of networks will want to use the PCN mechanisms in various arrangements depending, for instance, on how they are performing admission control outside the PCN-domain (users after all are concerned about QoS end-to-end), what their particular goals and assumptions are, how many PCN encoding states are available, and so on.

A PCN-domain may have three encoding states (or pedantically, an operator may choose to use up three encoding states for PCN): not PCN-marked, threshold-marked, and excess-traffic-marked. This way, both PCN admission control and flow termination can be supported. As

illustrated in Figure 1, admission control accepts new flows until the PCN-traffic rate on the bottleneck link rises above the PCN-threshold-rate, whilst, if necessary, the flow termination mechanism terminates flows down to the PCN-excess-rate on the bottleneck link.

On the other hand, a PCN-domain may have two encoding states (as in [Moncaster09-1]) (or pedantically, an operator may choose to use up two encoding states for PCN): not PCN-marked and PCN-marked. This way, there are three possibilities, as discussed in the following paragraphs (see also Section 3.3).

First, an operator could just use PCN's admission control, solving heavy congestion (caused by re-routing) by "just waiting" -- as sessions end, PCN-traffic naturally reduces; meanwhile, the admission control mechanism will prevent admission of new flows that use the affected links. So, the PCN-domain will naturally return to normal operation, but with reduced capacity. The drawback of this approach would be that, until sufficient sessions have ended to relieve the congestion, all PCN-flows as well as lower-priority services will be adversely affected.

Second, an operator could just rely on statically provisioned capacity per PCN-ingress-node (regardless of the PCN-egress-node of a flow) for admission control, as is typical in the hose model of the Diffserv architecture [Kumar01]. Such traffic-conditioning agreements can lead to focused overload: many flows happen to focus on a particular link and then all flows through the congested link fail catastrophically. PCN's flow termination mechanism could then be used to counteract such a problem.

Third, both admission control and flow termination can be triggered from the single type of PCN-marking; the main downside here is that admission control is less accurate [Charny07-2]. This possibility is illustrated in Figure 3.

Within the PCN-domain, there is some flexibility about how the decision-making functionality is distributed. These possibilities are outlined in Section 4.4 and are also discussed elsewhere, such as in [Menth09-2].

The flow admission and termination decisions need to be enforced through per-flow policing by the PCN-ingress-nodes. If there are several PCN-domains on the end-to-end path, then each needs to police at its PCN-ingress-nodes. One exception is if the operator runs both the access network (not a PCN-domain) and the core network (a PCN-domain); per-flow policing could be devolved to the access network

and not be done at the PCN-ingress-node. Note that, to aid readability, the rest of this document assumes that policing is done by the PCN-ingress-nodes.

PCN admission control has to fit with the overall approach to admission control. For instance, [Briscoe06] describes the case where RSVP signalling runs end-to-end. The PCN-domain is a single RSVP hop, ie, only the PCN-boundary-nodes process RSVP messages, with RSVP messages processed on each hop outside the PCN-domain, as in IntServ over Diffserv [RFC2998]. It would also be possible for the RSVP signalling to be originated and/or terminated by proxies, with application-layer signalling between the end user and the proxy (eg, SIP signalling with a home hub). A similar example would use NSIS (Next Steps in Signalling) [RFC3726] instead of RSVP.

It is possible that a user wants its inelastic traffic to use the PCN mechanisms but also react to ECN markings outside the PCN-domain [Sarker08]. Two possible ways to do this are to tunnel all PCN-packets across the PCN-domain, so that the ECN marks are carried transparently across the PCN-domain, or to use an encoding like [Moncaster09-2]. Tunnelling is discussed further in Section 4.7.

Some further possible deployment models are outlined in the Appendix.

6.3. Assumptions and Constraints on Scope

The scope of this document is restricted by the following assumptions:

1. These components are deployed in a single Diffserv domain, within which all PCN-nodes are PCN-enabled and are trusted for truthful PCN-marking and transport.

2. All flows handled by these mechanisms are inelastic and constrained to a known peak rate through policing or shaping.

3. The number of PCN-flows across any potential bottleneck link is sufficiently large that stateless, statistical mechanisms can be effective. To put it another way, the aggregate bit rate of PCN-traffic across any potential bottleneck link needs to be sufficiently large, relative to the maximum additional bit rate added by one flow. This is the basic assumption of measurement-based admission control.

4. PCN-flows may have different precedence, but the applicability of the PCN mechanisms for emergency use (911, GETS (Government Telecommunications Service), WPS (Wireless Priority Service), MLPP (Multilevel Precedence and Premption), etc.) is out of scope.

6.3.1. Assumption 1: Trust and Support of PCN - Controlled Environment

It is assumed that the PCN-domain is a controlled environment, ie, all the nodes in a PCN-domain run PCN and are trusted. There are several reasons for this assumption:

- The PCN-domain has to be encircled by a ring of PCN-boundary-nodes; otherwise, traffic could enter a PCN-BA without being subject to admission control, which would potentially degrade the QoS of existing PCN-flows.

- Similarly, a PCN-boundary-node has to trust that all the PCN-nodes mark PCN-traffic consistently. A node not performing PCN-marking wouldn't be able to send an alert when it suffered pre-congestion, which potentially would lead to too many PCN-flows being admitted (or too few being terminated). Worse, a rogue node could perform various attacks, as discussed in Section 7.

One way of assuring the above two points are in effect is to have the entire PCN-domain run by a single operator. Another way is to have several operators that trust each other in their handling of PCN-traffic.

Note: All PCN-nodes need to be trustworthy. However, if it is known that an interface cannot become pre-congested, then it is not strictly necessary for it to be capable of PCN-marking, but this must be known even in unusual circumstances, eg, after the failure of some links.

6.3.2. Assumption 2: Real-Time Applications

It is assumed that any variation of source bit rate is independent of the level of pre-congestion. We assume that PCN-packets come from real-time applications generating inelastic traffic, ie, sending packets at the rate the codec produces them, regardless of the availability of capacity [RFC4594]. Examples of such real-time applications include voice and video requiring low delay, jitter, and packet loss, the Controlled Load Service [RFC2211], and the Telephony service class [RFC4594]. This assumption is to help focus the effort where it looks like PCN would be most useful, ie, the sorts of

applications where per-flow QoS is a known requirement. In other words, we focus on PCN providing a benefit to inelastic traffic (PCN may or may not provide a benefit to other types of traffic).

As a consequence, it is assumed that PCN-metering and PCN-marking is being applied to traffic scheduled with an expedited forwarding per-hop behaviour [RFC3246] or with a per-hop behaviour with similar characteristics.

6.3.3. Assumption 3: Many Flows and Additional Load

It is assumed that there are many PCN-flows on any bottleneck link in the PCN-domain (or, to put it another way, the aggregate bit rate of PCN-traffic across any potential bottleneck link is sufficiently large, relative to the maximum additional bit rate added by one PCN-flow). Measurement-based admission control assumes that the present is a reasonable prediction of the future: the network conditions are measured at the time of a new flow request, but the actual network performance must be acceptable during the call some time later. One issue is that if there are only a few variable rate flows, then the aggregate traffic level may vary a lot, perhaps enough to cause some packets to get dropped. If there are many flows, then the aggregate traffic level should be statistically smoothed. How many flows is enough depends on a number of factors, such as the variation in each flow's rate, the total rate of PCN-traffic, and the size of the "safety margin" between the traffic level at which we start admission-marking and at which packets are dropped or significantly delayed.

No explicit assumptions are made about how many PCN-flows are in each ingress-egress-aggregate. Performance-evaluation work may clarify whether it is necessary to make any additional assumptions on aggregation at the ingress-egress-aggregate level.

6.3.4. Assumption 4: Emergency Use Out of Scope

PCN-flows may have different precedence, but the applicability of the PCN mechanisms for emergency use (911, GETS, WPS, MLPP, etc.) is out of scope for this document.

6.4. Challenges

Prior work on PCN and similar mechanisms has led to a number of considerations about PCN's design goals (things PCN should be good at) and some issues that have been hard to solve in a fully satisfactory manner. Taken as a whole, PCN represents a list of

trade-offs (it is unlikely that they can all be 100% achieved) and perhaps a list of evaluation criteria to help an operator (or the IETF) decide between options.

The following are open issues. They are mainly taken from [Briscoe06], which also describes some possible solutions. Note that some may be considered unimportant in general or in specific deployment scenarios, or by some operators.

Note: Potential solutions are out of scope for this document.

o ECMP (Equal Cost Multi-Path) Routing: The level of pre-congestion is measured on a specific ingress-egress-aggregate. However, if the PCN-domain runs ECMP, then traffic on this ingress-egress-aggregate may follow several different paths -- some of the paths could be pre-congested whilst others are not. There are three potential problems:

1. over-admission: a new flow is admitted (because the pre-congestion level measured by the PCN-egress-node is sufficiently diluted by unmarked packets from non-congested paths that a new flow is admitted), but its packets travel through a pre-congested PCN-node.

2. under-admission: a new flow is blocked (because the pre-congestion level measured by the PCN-egress-node is sufficiently increased by PCN-marked packets from pre-congested paths that a new flow is blocked), but its packets travel along an uncongested path.

3. ineffective termination: a flow is terminated but its path doesn't travel through the (pre-)congested router(s). Since flow termination is a "last resort", which protects the network should over-admission occur, this problem is probably more important to solve than the other two.

o ECMP and Signalling: It is possible that, in a PCN-domain running ECMP, the signalling packets (eg, RSVP, NSIS) follow a different path than the data packets, which could matter if the signalling packets are used as probes. Whether this is an issue depends on which fields the ECMP algorithm uses; if the ECMP algorithm is restricted to the source and destination IP addresses, then it will not be an issue. ECMP and signalling interactions are a specific instance of a general issue for non-traditional routing combined with resource management along a path [Hancock02].

- Tunnelling: There are scenarios where tunnelling makes it difficult to determine the path in the PCN-domain. The problem, its impact, and the potential solutions are similar to those for ECMP.

- Scenarios with only one tunnel endpoint in the PCN-domain: Such scenarios may make it harder for the PCN-egress-node to gather from the signalling messages (eg, RSVP, NSIS) the identity of the PCN-ingress-node.

- Bi-Directional Sessions: Many applications have bi-directional sessions -- hence, there are two microflows that should be admitted (or terminated) as a pair -- for instance, a bi-directional voice call only makes sense if microflows in both directions are admitted. However, the PCN mechanisms concern admission and termination of a single flow, and coordination of the decision for both flows is a matter for the signalling protocol and out of scope for PCN. One possible example would use SIP pre-conditions. However, there are others.

- Global Coordination: PCN makes its admission decision based on PCN-markings on a particular ingress-egress-aggregate. Decisions about flows through a different ingress-egress-aggregate are made independently. However, one can imagine network topologies and traffic matrices where, from a global perspective, it would be better to make a coordinated decision across all the ingress-egress-aggregates for the whole PCN-domain. For example, to block (or even terminate) flows on one ingress-egress-aggregate so that more important flows through a different ingress-egress-aggregate could be admitted. The problem may well be relatively insignificant.

- Aggregate Traffic Characteristics: Even when the number of flows is stable, the traffic level through the PCN-domain will vary because the sources vary their traffic rates. PCN works best when there is not too much variability in the total traffic level at a PCN-node's interface (ie, in the aggregate traffic from all sources). Too much variation means that a node may (at one moment) not be doing any PCN-marking and then (at another moment) drop packets because it is overloaded. This makes it hard to tune the admission control scheme to stop admitting new flows at the right time. Therefore, the problem is more likely with fewer, burstier flows.

- Flash crowds and Speed of Reaction: PCN is a measurement-based mechanism and so there is an inherent delay between packet marking by PCN-interior-nodes and any admission control reaction at PCN-boundary-nodes. For example, if a big burst of admission requests

potentially occurs in a very short space of time (eg, prompted by a televote), they could all get admitted before enough PCN-marks are seen to block new flows. In other words, any additional load offered within the reaction time of the mechanism must not move the PCN-domain directly from a no congestion state to overload. This "vulnerability period" may have an impact at the signalling level, for instance, QoS requests should be rate-limited to bound the number of requests able to arrive within the vulnerability period.

- Silent at Start: After a successful admission request, the source may wait some time before sending data (eg, waiting for the called party to answer). Then the risk is that, in some circumstances, PCN's measurements underestimate what the pre-congestion level will be when the source does start sending data.

7. Security Considerations

Security considerations essentially come from the Trust Assumption Section 6.3.1, ie, that all PCN-nodes are PCN-enabled and are trusted for truthful PCN-metering and PCN-marking. PCN splits functionality between PCN-interior-nodes and PCN-boundary-nodes, and the security considerations are somewhat different for each, mainly because PCN-boundary-nodes are flow-aware and PCN-interior-nodes are not.

- Because PCN-boundary-nodes are flow-aware, they are trusted to use that awareness correctly. The degree of trust required depends on the kinds of decisions they have to make and the kinds of information they need to make them. There is nothing specific to PCN.

- The PCN-ingress-nodes police packets to ensure a PCN-flow sticks within its agreed limit, and to ensure that only PCN-flows that have been admitted contribute PCN-traffic into the PCN-domain. The policer must drop (or perhaps downgrade to a different DSCP) any PCN-packets received that are outside this remit. This is similar to the existing IntServ behaviour. Between them, the PCN-boundary-nodes must encircle the PCN-domain; otherwise, PCN-packets could enter the PCN-domain without being subject to admission control, which would potentially destroy the QoS of existing flows.

- PCN-interior-nodes are not flow-aware. This prevents some security attacks where an attacker targets specific flows in the data plane -- for instance, for DoS or eavesdropping.

- The PCN-boundary-nodes rely on correct PCN-marking by the PCN-interior-nodes. For instance, a rogue PCN-interior-node could PCN-mark all packets so that no flows were admitted. Another possibility is that it doesn't PCN-mark any packets, even when it is pre-congested. More subtly, the rogue PCN-interior-node could perform these attacks selectively on particular flows, or it could PCN-mark the correct fraction overall but carefully choose which flows it marked.

- The PCN-boundary-nodes should be able to deal with DoS attacks and state exhaustion attacks based on fast changes in per-flow signalling.

- The signalling between the PCN-boundary-nodes must be protected from attacks. For example, the recipient needs to validate that the message is indeed from the node that claims to have sent it. Possible measures include digest authentication and protection against replay and man-in-the-middle attacks. For the RSVP protocol specifically, hop-by-hop authentication is in [RFC2747], and [Behringer09] may also be useful.

Operational security advice is given in Section 5.5.

8. Conclusions

This document describes a general architecture for flow admission and termination based on pre-congestion information, in order to protect the quality of service of established, inelastic flows within a single Diffserv domain. The main topic is the functional architecture. This document also mentions other topics like the assumptions and open issues associated with the PCN architecture.

9. Acknowledgements

This document is a revised version of an earlier individual working draft authored by: P. Eardley, J. Babiarz, K. Chan, A. Charny, R. Geib, G. Karagiannis, M. Menth, and T. Tsou. They are therefore contributors to this document.

Thanks to those who have made comments on this document: Lachlan Andrew, Joe Babiarz, Fred Baker, David Black, Steven Blake, Ron Bonica, Scott Bradner, Bob Briscoe, Ross Callon, Jason Canon, Ken Carlberg, Anna Charny, Joachim Charzinski, Andras Csaszar, Francis Dupont, Lars Eggert, Pasi Eronen, Adrian Farrel, Ruediger Geib, Wei Gengyu, Robert Hancock, Fortune Huang, Christian Hublet, Cullen Jennings, Ingemar Johansson, Georgios Karagiannis, Hein Mekkes, Michael Menth, Toby Moncaster, Dimitri Papadimitriou, Dan Romascanu, Daisuke Satoh, Ben Strulo, Tom Taylor, Hannes Tschofenig, Tina Tsou,

David Ward, Lars Westberg, Magnus Westerlund, and Delei Yu. Thanks to Bob Briscoe who extensively revised the Operations and Management section.

This document is the result of discussions in the PCN WG and forerunner activity in the TSVWG. A number of previous drafts were presented to TSVWG; their authors were: B. Briscoe, P. Eardley, D. Songhurst, F. Le Faucheur, A. Charny, J. Babiarz, K. Chan, S. Dudley, G. Karagiannis, A. Bader, L. Westberg, J. Zhang, V. Liatsos, X-G. Liu, and A. Bhargava.

The admission control mechanism evolved from the work led by Martin Karsten on the Guaranteed Stream Provider developed in the M3I project [Karsten02] [M3I], which in turn was based on the theoretical work of Gibbens and Kelly [Gibbens99].

10. References

10.1. Normative References

[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.

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

10.2. Informative References

[RFC1633] Braden, B., Clark, D., and S. Shenker, "Integrated Services in the Internet Architecture: an Overview", RFC 1633, June 1994.

[RFC2205] Braden, B., Zhang, L., Berson, S., Herzog, S., and S. Jamin, "Resource ReSerVation Protocol (RSVP) -- Version 1 Functional Specification", RFC 2205, September 1997.

[RFC2211] Wroclawski, J., "Specification of the Controlled-Load Network Element Service", RFC 2211, September 1997.

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

[RFC2747] Baker, F., Lindell, B., and M. Talwar, "RSVP Cryptographic Authentication", RFC 2747, January 2000.

[RFC2753] Yavatkar, R., Pendarakis, D., and R. Guerin, "A Framework for Policy-based Admission Control", RFC 2753, January 2000.

[RFC2983] Black, D., "Differentiated Services and Tunnels", RFC 2983, October 2000.

[RFC2998] Bernet, Y., Ford, P., Yavatkar, R., 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.

[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.

[RFC3393] Demichelis, C. and P. Chimento, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)", RFC 3393, November 2002.

[RFC3411] Harrington, D., Presuhn, R., and B. Wijnen, "An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks", STD 62, RFC 3411, December 2002.

[RFC3726] Brunner, M., "Requirements for Signaling Protocols", RFC 3726, April 2004.

[RFC4216] Zhang, R. and J. Vasseur, "MPLS Inter-Autonomous System (AS) Traffic Engineering (TE) Requirements", RFC 4216, November 2005.

[RFC4301] Kent, S. and K. Seo, "Security Architecture for the Internet Protocol", RFC 4301, December 2005.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, December 2005.

[RFC4594] Babiarz, J., Chan, K., and F. Baker, "Configuration Guidelines for DiffServ Service Classes", RFC 4594, August 2006.

[RFC4656] Shalunov, S., Teitelbaum, B., Karp, A., Boote, J., and M. Zekauskas, "A One-way Active Measurement Protocol (OWAMP)", RFC 4656, September 2006.

[RFC4774] Floyd, S., "Specifying Alternate Semantics for the Explicit Congestion Notification (ECN) Field", BCP 124, RFC 4774, November 2006.

[RFC4778] Kaeo, M., "Operational Security Current Practices in Internet Service Provider Environments", RFC 4778, January 2007.

[RFC5129] Davie, B., Briscoe, B., and J. Tay, "Explicit Congestion Marking in MPLS", RFC 5129, January 2008.

[RFC5462] Andersson, L. and R. Asati, "Multiprotocol Label Switching (MPLS) Label Stack Entry: "EXP" Field Renamed to "Traffic Class" Field", RFC 5462, February 2009.

[P.800] "Methods for subjective determination of transmission quality", ITU-T Recommendation P.800, August 1996.

[Y.1541] "Network Performance Objectives for IP-based Services", ITU-T Recommendation Y.1541, February 2006.

[Babiarz06] Babiarz, J., Chan, K., Karagiannis, G., and P. Eardley, "SIP Controlled Admission and Preemption", Work in Progress, October 2006.

[Behringer09] Behringer, M. and F. Le Faucheur, "Applicability of Keying Methods for RSVP Security", Work in Progress, March 2009.

[Briscoe06] Briscoe, B., Eardley, P., Songhurst, D., Le Faucheur, F., Charny, A., Babiarz, J., Chan, K., Dudley, S., Karagiannis, G., Bader, A., and L. Westberg, "An edge-to-edge Deployment Model for Pre-Congestion Notification: Admission Control over a Diffserv Region", Work in Progress, October 2006.

[Briscoe08] Briscoe, B., "Emulating Border Flow Policing using Re-PCN on Bulk Data", Work in Progress, September 2008.

[Briscoe09] Briscoe, B., "Tunnelling of Explicit Congestion Notification", Work in Progress, March 2009.

[Bryant08] Bryant, S., Davie, B., Martini, L., and E. Rosen, "Pseudowire Congestion Control Framework", Work in Progress, May 2008.

[Charny07-1] Charny, A., Babiarz, J., Menth, M., and X. Zhang, "Comparison of Proposed PCN Approaches", Work in Progress, November 2007.

[Charny07-2] Charny, A., Zhang, X., Le Faucheur, F., and V. Liatsos, "Pre-Congestion Notification Using Single Marking for Admission and Termination", Work in Progress, November 2007.

[Charny07-3] Charny, A., "Email to PCN WG mailing list", November 2007, <http://www1.ietf.org/mail-archive/web/pcn/current/msg00871.html>.

[Charny08] Charny, A., "Email to PCN WG mailing list", March 2008, <http://www1.ietf.org/mail-archive/web/pcn/current/msg01359.html>.

[Eardley07] Eardley, P., "Email to PCN WG mailing list", October 2007, <http://www1.ietf.org/mail-archive/web/pcn/current/msg00831.html>.

[Eardley09] Eardley, P., "Metering and marking behaviour of PCN-nodes", Work in Progress, May 2009.

[Gibbens99] Gibbens, R. and F. Kelly, "Distributed connection acceptance control for a connectionless network", Proceedings International Teletraffic Congress (ITC16), Edinburgh, pp. 941-952, 1999.

[Hancock02] Hancock, R. and E. Hepworth, "Slide 14 of 'NSIS: An Outline Framework for QoS Signalling'", May 2002, <http://www-nrc.nokia.com/sua/nsis/interim/nsis-framework-outline.ppt>.

[Iyer03] Iyer, S., Bhattacharyya, S., Taft, N., and C. Diot, "An approach to alleviate link overload as observed on an IP backbone", IEEE INFOCOM, 2003, <http://www.ieee-infocom.org/2003/papers/10_04.pdf>.

[Karsten02] Karsten, M. and J. Schmitt, "Admission Control Based on Packet Marking and Feedback Signalling -- Mechanisms, Implementation and Experiments", TU-Darmstadt Technical Report TR-KOM-2002-03, May 2002, <http://www.kom.e-technik.tu-darmstadt.de/publications/abstracts/KS02-5.html>.

[Kumar01] Kumar, A., Rastogi, R., Silberschatz, A., and B. Yener, "Algorithms for Provisioning Virtual Private Networks in the Hose Model", Proceedings ACM SIGCOMM (ITC16), , 2001.

[Lefaucheur06] Le Faucheur, F., Charny, A., Briscoe, B., Eardley, P., Babiarz, J., and K. Chan, "RSVP Extensions for Admission Control over Diffserv using Pre-congestion Notification (PCN)", Work in Progress, June 2006.

[M3I] "M3I - Market Managed Multiservice Internet", <http://www.m3iproject.org/>.

[Menth08-1] Menth, M., Lehrieder, F., Eardley, P., Charny, A., and J. Babiarz, "Edge-Assisted Marked Flow Termination", Work in Progress, February 2008.

[Menth08-2] Menth, M., Babiarz, J., Moncaster, T., and B. Briscoe, "PCN Encoding for Packet-Specific Dual Marking (PSDM)", Work in Progress, July 2008.

[Menth09-1] Menth, M. and M. Hartmann, "Threshold Configuration and Routing Optimization for PCN-Based Resilient Admission Control", Computer Networks, 2009, <http://dx.doi.org/10.1016/j.comnet.2009.01.013>.

[Menth09-2] Menth, M., Lehrieder, F., Briscoe, B., Eardley, P., Moncaster, T., Babiarz, J., Chan, K., Charny, A., Karagiannis, G., Zhang, X., Taylor, T., Satoh, D., and R. Geib, "A Survey of PCN-Based Admission Control and Flow Termination", IEEE Communications Surveys and Tutorials, <http://www3.informatik.uni-wuerzburg.de/staff/menth/Publications/papers/Menth08-PCN-Overview.pdf>.

[Moncaster09-1] Moncaster, T., Briscoe, B., and M. Menth, "Baseline Encoding and Transport of Pre-Congestion Information", Work in Progress, May 2009.

[Moncaster09-2] Moncaster, T., Briscoe, B., and M. Menth, "A PCN encoding using 2 DSCPs to provide 3 or more states", Work in Progress, April 2009.

[Sarker08] Sarker, Z. and I. Johansson, "Usecases and Benefits of end to end ECN support in PCN Domains", Work in Progress, November 2008.

[Songhurst06] Songhurst, DJ., Eardley, P., Briscoe, B., Di Cairano Gilfedder, C., and J. Tay, "Guaranteed QoS Synthesis for Admission Control with Shared Capacity", BT Technical Report TR-CXR9-2006-001, Feburary 2006, <http://www.cs.ucl.ac.uk/staff/B.Briscoe/projects/ipe2eqos/gqs/papers/GQS_shared_tr.pdf>.

[Taylor09] Charny, A., Huang, F., Menth, M., and T. Taylor, "PCN Boundary Node Behaviour for the Controlled Load (CL) Mode of Operation", Work in Progress, March 2009.

[Tsou08] Tsou, T., Huang, F., and T. Taylor, "Applicability Statement for the Use of Pre-Congestion Notification in a Resource-Controlled Network", Work in Progress, November 2008.

[Westberg08] Westberg, L., Bhargava, A., Bader, A., Karagiannis, G., and H. Mekkes, "LC-PCN: The Load Control PCN Solution", Work in Progress, November 2008.

Appendix A. Possible Future Work Items

This section mentions some topics that are outside the PCN WG's current charter but that have been mentioned as areas of interest. They might be work items for the PCN WG after a future re-chartering, some other IETF WG, another standards body, or an operator-specific usage that is not standardised.

Note: It should be crystal clear that this section discusses possibilities only.

The first set of possibilities relate to the restrictions described in Section 6.3:

- A single PCN-domain encompasses several autonomous systems that do not trust each other. A possible solution is a mechanism like re-PCN [Briscoe08].

- Not all the nodes run PCN. For example, the PCN-domain is a multi-site enterprise network. The sites are connected by a VPN tunnel; although PCN doesn't operate inside the tunnel, the PCN mechanisms still work properly because of the good QoS on the virtual link (the tunnel). Another example is that PCN is deployed on the general Internet (ie, widely but not universally deployed).

- Applying the PCN mechanisms to other types of traffic, ie, beyond inelastic traffic -- for instance, applying the PCN mechanisms to traffic scheduled with the Assured Forwarding per-hop behaviour. One example could be flow-rate adaptation by elastic applications that adapt according to the pre-congestion information.

- The aggregation assumption doesn't hold, because the link capacity is too low. Measurement-based admission control is less accurate, with a greater risk of over-admission for instance.

- The applicability of PCN mechanisms for emergency use (911, GETS, WPS, MLPP, etc.).

Other possibilities include:

- Probing. This is discussed in Appendix A.1 below.

- The PCN-domain extends to the end users. This scenario is described in [Babiarz06]. The end users need to be trusted to do their own policing. If there is sufficient traffic, then the aggregation assumption may hold. A variant is that the PCN-domain extends out as far as the LAN edge switch.

- Indicating pre-congestion through signalling messages rather than in-band (in the form of PCN-marked packets).

- The decision-making functionality is at a centralised node rather than at the PCN-boundary-nodes. This requires that the PCN-egress-node signals PCN-feedback-information to the centralised node, and that the centralised node signals to the PCN-ingress-node the decision about admission (or termination). Such possibility may need the centralised node and the PCN-boundary-nodes to be configured with each other's addresses. The centralised case is described further in [Tsou08].

- Signalling extensions for specific protocols (eg, RSVP and NSIS) -- for example, the details of how the signalling protocol installs the flowspec at the PCN-ingress-node for an admitted PCN-flow, and how the signalling protocol carries the PCN-feedback-information. Perhaps also for other functions such as for coping with failure of a PCN-boundary-node ([Briscoe06] considers what happens if RSVP is the QoS signalling protocol) and for establishing a tunnel across the PCN-domain if it is necessary to carry ECN marks transparently.

- Policing by the PCN-ingress-node may not be needed if the PCN-domain can trust that the upstream network has already policed the traffic on its behalf.

- PCN for Pseudowire. PCN may be used as a congestion avoidance mechanism for edge-to-edge pseudowire emulations [Bryant08].

- PCN for MPLS. [RFC3270] defines how to support the Diffserv architecture in MPLS (Multiprotocol Label Switching) networks. [RFC5129] describes how to add PCN for admission control of microflows into a set of MPLS aggregates. PCN-marking is done in MPLS's EXP field (which [RFC5462] re-names the Class of Service (CoS) field).

- PCN for Ethernet. Similarly, it may be possible to extend PCN into Ethernet networks, where PCN-marking is done in the Ethernet header. Note: Specific consideration of this extension is outside of the IETF's remit.

A.1. Probing

A.1.1. Introduction

Probing is a potential mechanism to assist admission control.

PCN's admission control, as described so far, is essentially a reactive mechanism where the PCN-egress-node monitors the pre-congestion level for traffic from each PCN-ingress-node; if the level rises, then it blocks new flows on that ingress-egress-aggregate. However, it's possible that an ingress-egress-aggregate carries no traffic, and so the PCN-egress-node can't make an admission decision using the usual method described earlier.

One approach is to be "optimistic" and simply admit the new flow. However, it's possible to envisage a scenario where the traffic levels on other ingress-egress-aggregates are already so high that they're blocking new PCN-flows, and admitting a new flow onto this "empty" ingress-egress-aggregate adds extra traffic onto a link that is already pre-congested. This may 'tip the balance' so that PCN's flow termination mechanism is activated or some packets are dropped. This risk could be lessened by configuring, on each link, a sufficient 'safety margin' above the PCN-threshold-rate.

An alternative approach is to make PCN a more proactive mechanism. The PCN-ingress-node explicitly determines, before admitting the prospective new flow, whether the ingress-egress-aggregate can support it. This can be seen as a "pessimistic" approach, in contrast to the "optimism" of the approach above. It involves probing: a PCN-ingress-node generates and sends probe packets in order to test the pre-congestion level that the flow would experience.

One possibility is that a probe packet is just a dummy data packet, generated by the PCN-ingress-node and addressed to the PCN-egress-node.

A.1.2. Probing Functions

The probing functions are:

- o Make the decision that probing is needed. As described above, this is when the ingress-egress-aggregate (or the ECMP path -- see Section 6.4) carries no PCN-traffic. An alternative is to always probe, ie, probe before admitting any PCN-flow.

- (if required) Communicate the request that probing is needed; the PCN-egress-node signals to the PCN-ingress-node that probing is needed.

- (if required) Generate probe traffic; the PCN-ingress-node generates the probe traffic. The appropriate number (or rate) of probe packets will depend on the PCN-metering algorithm; for example, an excess-traffic-metering algorithm triggers fewer PCN-marks than a threshold-metering algorithm, and so will need more probe packets.

- Forward probe packets; as far as PCN-interior-nodes are concerned, probe packets are handled the same as (ordinary data) PCN-packets in terms of routing, scheduling, and PCN-marking.

- Consume probe packets; the PCN-egress-node consumes probe packets to ensure that they don't travel beyond the PCN-domain.

A.1.3. Discussion of Rationale for Probing, Its Downsides and Open Issues

It is an unresolved question whether probing is really needed, but two viewpoints have been put forward as to why it is useful. The first is perhaps the most obvious: there is no PCN-traffic on the ingress-egress-aggregate. The second assumes that multipath routing (eg, ECMP) is running in the PCN-domain. We now consider each in turn.

The first viewpoint assumes the following:

- There is no PCN-traffic on the ingress-egress-aggregate (so a normal admission decision cannot be made).

- Simply admitting the new flow has a significant risk of leading to overload: packets dropped or flows terminated.

On the former bullet, [Eardley07] suggests that, during the future busy hour of a national network with about 100 PCN-boundary-nodes, there are likely to be significant numbers of aggregates with very few flows under nearly all circumstances.

The latter bullet could occur if new flows start on many of the empty ingress-egress-aggregates, which together overload a link in the PCN-domain. To be a problem, this would probably have to happen in a short time period (flash crowd) because, after the reaction time of the system, other (non-empty) ingress-egress-aggregates that pass through the link will measure pre-congestion and so block new flows. Also, flows naturally end anyway.

The downsides of probing for this viewpoint are:

- Probing adds delay to the admission control process.

- Sufficient probing traffic has to be generated to test the pre-congestion level of the ingress-egress-aggregate. But the probing traffic itself may cause pre-congestion, causing other PCN-flows to be blocked or even terminated -- and, in the flash crowd scenario, there will be probing on many ingress-egress-aggregates.

The second viewpoint applies in the case where there is multipath routing (eg, ECMP) in the PCN-domain. Note that ECMP is often used on core networks. There are two possibilities:

(1) If admission control is based on measurements of the ingress-egress-aggregate, then the viewpoint that probing is useful assumes:

  * There's a significant chance that the traffic is unevenly balanced across the ECMP paths and, hence, there's a significant risk of admitting a flow that should be blocked (because it follows an ECMP path that is pre-congested) or of blocking a flow that should be admitted.

  Note: [Charny07-3] suggests unbalanced traffic is quite possible, even with quite a large number of flows on a PCN-link (eg, 1000), when Assumption 3 (aggregation) is likely to be satisfied.

(2) If admission control is based on measurements of pre-congestion on specific ECMP paths, then the viewpoint that probing is useful assumes:

  * There is no PCN-traffic on the ECMP path on which to base an admission decision.

  * Simply admitting the new flow has a significant risk of leading to overload.

  * The PCN-egress-node can match a packet to an ECMP path.

  Note: This is similar to the first viewpoint and so, similarly, could occur in a flash crowd if a new flow starts more or less simultaneously on many of the empty ECMP paths. Because there are several ECMP paths between each pair of PCN-boundary-nodes, it's presumably more likely that an ECMP path is "empty" than an ingress-egress-aggregate is. To constrain the number of ECMP paths, a few tunnels could be set up between each pair of PCN-

boundary-nodes. Tunnelling also solves the issue in the point immediately above (which is otherwise hard to solve because an ECMP routing decision is made independently on each node).

The downsides of probing for this viewpoint are:

- o Probing adds delay to the admission control process.

- o Sufficient probing traffic has to be generated to test the pre-congestion level of the ECMP path. But there's the risk that the probing traffic itself may cause pre-congestion, causing other PCN-flows to be blocked or even terminated.

- o The PCN-egress-node needs to consume the probe packets to ensure they don't travel beyond the PCN-domain, since they might confuse the destination end node. This is non-trivial, since probe packets are addressed to the destination end node in order to test the relevant ECMP path (ie, they are not addressed to the PCN-egress-node, unlike the first viewpoint above).

The open issues associated with these viewpoints include:

- o What rate and pattern of probe packets does the PCN-ingress-node need to generate so that there's enough traffic to make the admission decision?

- o What difficulty does the delay (whilst probing is done), and possible packet drops, cause applications?

- o Can the delay be alleviated by automatically and periodically probing on the ingress-egress-aggregate? Or does this add too much overhead?

- o Are there other ways of dealing with the flash crowd scenario? For instance, by limiting the rate at which new flows are admitted, or perhaps by a PCN-egress-node blocking new flows on its empty ingress-egress-aggregates when its non-empty ones are pre-congested.

- o (Second viewpoint only) How does the PCN-egress-node disambiguate probe packets from data packets (so it can consume the former)? The PCN-egress-node must match the characteristic setting of particular bits in the probe packet's header or body, but these bits must not be used by any PCN-interior-node's ECMP algorithm. In the general case, this isn't possible, but it should be possible for a typical ECMP algorithm (which examines the source and destination IP addresses and port numbers, the protocol ID, and the DSCP).

Author's Address

Philip Eardley (editor)
BT
B54/77, Sirius House Adastral Park Martlesham Heath
Ipswich, Suffolk IP5 3RE
United Kingdom

EMail: philip.eardley@bt.com