Network Working Group
Request for Comments: 2598
Category: Standards Track

V. Jacobson
K. Nichols
Cisco Systems
K. Poduri
Bay Networks

June 1999

# An Expedited Forwarding PHB

Status of this Memo



Copyright Notice



Abstract

The definition of PHBs (per-hop forwarding behaviors) is a critical part of the work of the Diffserv Working Group. This document describes a PHB called Expedited Forwarding. We show the generality of this PHB by noting that it can be produced by more than one mechanism and give an example of its use to produce at least one service, a Virtual Leased Line. A recommended codepoint for this PHB is given.

A pdf version of this document is available at ftp://ftp.ee.lbl.gov/papers/ef_phb.pdf

## 1. Introduction

Network nodes that implement the differentiated services enhancements to IP use a codepoint in the IP header to select a per-hop behavior (PHB) as the specific forwarding treatment for that packet [RFC2474, RFC2475]. This memo describes a particular PHB called expedited forwarding (EF). The EF PHB can be used to build a low loss, low latency, low jitter, assured bandwidth, end-to-end service through DS domains. Such a service appears to the endpoints like a point-to-point connection or a "virtual leased line". This service has also been described as Premium service [2BIT].

Loss, latency and jitter are all due to the queues traffic experiences while transiting the network. Therefore providing low loss, latency and jitter for some traffic aggregate means ensuring that the aggregate sees no (or very small) queues. Queues arise when (short-term) traffic arrival rate exceeds departure rate at some node. Thus a service that ensures no queues for some aggregate is equivalent to bounding rates such that, at every transit node, the aggregate's maximum arrival rate is less than that aggregate's minimum departure rate.

Creating such a service has two parts:

1) Configuring nodes so that the aggregate has a well-defined minimum departure rate. ("Well-defined" means independent of the dynamic state of the node. In particular, independent of the intensity of other traffic at the node.)

2) Conditioning the aggregate (via policing and shaping) so that its arrival rate at any node is always less than that node's configured minimum departure rate.

The EF PHB provides the first part of the service. The network boundary traffic conditioners described in [RFC2475] provide the second part.

The EF PHB is not a mandatory part of the Differentiated Services architecture, i.e., a node is not required to implement the EF PHB in order to be considered DS-compliant. However, when a DS-compliant node claims to implement the EF PHB, the implementation must conform to the specification given in this document.

The next sections describe the EF PHB in detail and give examples of how it might be implemented. The keywords "MUST", "MUST NOT", "REQUIRED", "SHOULD", "SHOULD NOT", and "MAY" that appear in this document are to be interpreted as described in [Bradner97].

## 2. Description of EF per-hop behavior

The EF PHB is defined as a forwarding treatment for a particular diffserv aggregate where the departure rate of the aggregate's packets from any diffserv node must equal or exceed a configurable rate. The EF traffic SHOULD receive this rate independent of the intensity of any other traffic attempting to transit the node. It SHOULD average at least the configured rate when measured over any time interval equal to or longer than the time it takes to send an output link MTU sized packet at the configured rate. (Behavior at time scales shorter than a packet time at the configured rate is

deliberately not specified.) The configured minimum rate MUST be settable by a network administrator (using whatever mechanism the node supports for non-volatile configuration).

If the EF PHB is implemented by a mechanism that allows unlimited preemption of other traffic (e.g., a priority queue), the implementation MUST include some means to limit the damage EF traffic could inflict on other traffic (e.g., a token bucket rate limiter). Traffic that exceeds this limit MUST be discarded. This maximum EF rate, and burst size if appropriate, MUST be settable by a network administrator (using whatever mechanism the node supports for non-volatile configuration). The minimum and maximum rates may be the same and configured by a single parameter.

The Appendix describes how this PHB can be used to construct end-to-end services.

## 2.2 Example Mechanisms to Implement the EF PHB

Several types of queue scheduling mechanisms may be employed to deliver the forwarding behavior described in section 2.1 and thus implement the EF PHB. A simple priority queue will give the appropriate behavior as long as there is no higher priority queue that could preempt the EF for more than a packet time at the configured rate. (This could be accomplished by having a rate policer such as a token bucket associated with each priority queue to bound how much the queue can starve other traffic.)

It's also possible to use a single queue in a group of queues serviced by a weighted round robin scheduler where the share of the output bandwidth assigned to the EF queue is equal to the configured rate. This could be implemented, for example, using one PHB of a Class Selector Compliant set of PHBs [RFC2474].

Another possible implementation is a CBQ [CBQ] scheduler that gives the EF queue priority up to the configured rate.

All of these mechanisms have the basic properties required for the EF PHB though different choices result in different ancillary behavior such as jitter seen by individual microflows. See Appendix A.3 for simulations that quantify some of these differences.

## 2.3 Recommended codepoint for this PHB

Codepoint 101110 is recommended for the EF PHB.

### 2.4 Mutability

Packets marked for EF PHB MAY be remarked at a DS domain boundary only to other codepoints that satisfy the EF PHB. Packets marked for EF PHBs SHOULD NOT be demoted or promoted to another PHB by a DS domain.

### 2.5 Tunneling

When EF packets are tunneled, the tunneling packets must be marked as EF.

### 2.6 Interaction with other PHBs

Other PHBs and PHB groups may be deployed in the same DS node or domain with the EF PHB as long as the requirement of section 2.1 is met.

## 3. Security Considerations

To protect itself against denial of service attacks, the edge of a DS domain MUST strictly police all EF marked packets to a rate negotiated with the adjacent upstream domain. (This rate must be <= the EF PHB configured rate.) Packets in excess of the negotiated rate MUST be dropped. If two adjacent domains have not negotiated an EF rate, the downstream domain MUST use 0 as the rate (i.e., drop all EF marked packets).

Since the end-to-end premium service constructed from the EF PHB requires that the upstream domain police and shape EF marked traffic to meet the rate negotiated with the downstream domain, the downstream domain's policer should never have to drop packets. Thus these drops SHOULD be noted (e.g., via SNMP traps) as possible security violations or serious misconfiguration. Similarly, since the aggregate EF traffic rate is constrained at every interior node, the EF queue should never overflow so if it does the drops SHOULD be noted as possible attacks or serious misconfiguration.

## 4. IANA Considerations

This document allocates one codepoint, 101110, in Pool 1 of the code space defined by [RFC2474].

## 5. References

[Bradner97] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

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

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

[2BIT] K. Nichols, V. Jacobson, and L. Zhang, "A Two-bit Differentiated Services Architecture for the Internet", Work in Progress, ftp://ftp.ee.lbl.gov/papers/dsarch.pdf

[CBQ] S. Floyd and V. Jacobson, "Link-sharing and Resource Management Models for Packet Networks", IEEE/ACM Transactions on Networking, Vol. 3 no. 4, pp. 365-386, August 1995.

[RFC2415] Poduri, K. and K. Nichols, "Simulation Studies of Increased Initial TCP Window Size", RFC 2415, September 1998.

[LCN] K. Nichols, "Improving Network Simulation with Feedback", Proceedings of LCN '98, October 1998.

## 6. Authors' Addresses

Van Jacobson
Cisco Systems, Inc
170 W. Tasman Drive
San Jose, CA 95134-1706

EMail: van@cisco.com

Kathleen Nichols
Cisco Systems, Inc
170 W. Tasman Drive
San Jose, CA 95134-1706

EMail: kmn@cisco.com

Kedarnath Poduri
Bay Networks, Inc.
4401 Great America Parkway
Santa Clara, CA 95052-8185

EMail: kpoduri@baynetworks.com

Appendix A: Example use of and experiences with the EF PHB

## A.1 Virtual Leased Line Service

A VLL Service, also known as Premium service [2BIT], is quantified by a peak bandwidth.

## A.2 Experiences with its use in ESNET

A prototype of the VLL service has been deployed on DOE's ESNet backbone. This uses weighted-round-robin queuing features of Cisco 75xx series routers to implement the EF PHB. The early tests have been very successful and work is in progress to make the service available on a routine production basis (see ftp://ftp.ee.lbl.gov/talks/vj-doeqos.pdf and ftp://ftp.ee.lbl.gov/talks/vj-i2qos-may98.pdf for details).

## A.3 Simulation Results

### A.3.1 Jitter variation

In section 2.2, we pointed out that a number of mechanisms might be used to implement the EF PHB. The simplest of these is a priority queue (PQ) where the arrival rate of the queue is strictly less than its service rate. As jitter comes from the queuing delay along the path, a feature of this implementation is that EF-marked microflows will see very little jitter at their subscribed rate since packets spend little time in queues. The EF PHB does not have an explicit jitter requirement but it is clear from the definition that the expected jitter in a packet stream that uses a service based on the EF PHB will be less with PQ than with best-effort delivery. We used simulation to explore how weighted round-robin (WRR) compares to PQ in jitter. We chose these two since they"re the best and worst cases, respectively, for jitter and we wanted to supply rough guidelines for EF implementers choosing to use WRR or similar mechanisms.

Our simulation model is implemented in a modified ns-2 described in [RFC2415] and [LCN]. We used the CBQ modules included with ns-2 as a basis to implement priority queuing and WRR. Our topology has six hops with decreasing bandwidth in the direction of a single 1.5 Mbps bottleneck link (see figure 6). Sources produce EF-marked packets at an average bit rate equal to their subscribed packet rate. Packets are produced with a variation of +-10% from the interpacket spacing at the subscribed packet rate. The individual source rates were picked aggregate to 30% of the bottleneck link or 450 Kbps. A mixture of FTPs and HTTPs is then used to fill the link. Individual EF packet sources produce either all 160 byte packets or all 1500 byte packets.

Though we present the statistics of flows with one size of packet, all of the experiments used a mixture of short and long packet EF sources so the EF queues had a mix of both packet lengths.

We defined jitter as the absolute value of the difference between the arrival times of two adjacent packets minus their departure times, |(aj-dj) - (ai-di)|. For the target flow of each experiment, we record the median and 90th percentile values of jitter (expressed as % of the subscribed EF rate) in a table. The pdf version of this document contains graphs of the jitter percentiles.

Our experiments compared the jitter of WRR and PQ implementations of the EF PHB. We assessed the effect of different choices of WRR queue weight and number of queues on jitter. For WRR, we define the service-to-arrival rate ratio as the service rate of the EF queue (or the queue"s minimum share of the output link) times the output link bandwidth divided by the peak arrival rate of EF-marked packets at the queue. Results will not be stable if the WRR weight is chosen to exactly balance arrival and departure rates thus we used a minimum service-to-arrival ratio of 1.03. In our simulations this means that the EF queue gets at least 31% of the output links. In WRR simulations we kept the link full with other traffic as described above, splitting the non-EF-marked traffic among the non-EF queues. (It should be clear from the experiment description that we are attempting to induce worst-case jitter and do not expect these settings or traffic to represent a "normal" operating point.)

Our first set of experiments uses the minimal service-to-arrival ratio of 1.06 and we vary the number of individual microflows composing the EF aggregate from 2 to 36. We compare these to a PQ implementation with 24 flows. First, we examine a microflow at a subscribed rate of 56 Kbps sending 1500 byte packets, then one at the same rate but sending 160 byte packets. Table 1 shows the 50th and 90th percentile jitter in percent of a packet time at the subscribed rate. Figure 1 plots the 1500 byte flows and figure 2 the 160 byte flows. Note that a packet-time for a 1500 byte packet at 56 Kbps is 214 ms, for a 160 byte packet 23 ms. The jitter for the large packets rarely exceeds half a subscribed rate packet-time, though most jitters for the small packets are at least one subscribed rate packet-time. Keep in mind that the EF aggregate is a mixture of small and large packets in all cases so short packets can wait for long packets in the EF queue. PQ gives a very low jitter.

Table 1: Variation in jitter with number of EF flows: Service/arrival ratio of 1.06 and subscription rate of 56 Kbps (all values given as % of subscribed rate)

| | 1500 byte pack. | | 160 byte packet | |
|---|---|---|---|---|
| # EF flows | 50th % | 90th % | 50th % | 90th % |
| PQ (24) | 1 | 5 | 17 | 43 |
| 2 | 11 | 47 | 96 | 513 |
| 4 | 12 | 35 | 100 | 278 |
| 8 | 10 | 25 | 96 | 126 |
| 24 | 18 | 47 | 96 | 143 |

Next we look at the effects of increasing the service-to-arrival ratio. This means that EF packets should remain enqueued for less time though the bandwidth available to the other queues remains the same. In this set of experiments the number of flows in the EF aggregate was fixed at eight and the total number of queues at five (four non-EF queues). Table 2 shows the results for 1500 and 160 byte flows. Figures 3 plots the 1500 byte results and figure 4 the 160 byte results. Performance gains leveled off at service-to-arrival ratios of 1.5. Note that the higher service-to-arrival ratios do not give the same performance as PQ, but now 90% of packets experience less than a subscribed packet-time of jitter even for the small packets.

Table 2: Variation in Jitter of EF flows: service/arrival ratio varies, 8 flow aggregate, 56 Kbps subscribed rate

| WRR | 1500 byte pack. | | 160 byte packet | |
|---|---|---|---|---|
| Ser/Arr | 50th % | 90th % | 50th % | 90th % |
| PQ | 1 | 3 | 17 | 43 |
| 1.03 | 14 | 27 | 100 | 178 |
| 1.30 | 7 | 21 | 65 | 113 |
| 1.50 | 5 | 13 | 57 | 104 |
| 1.70 | 5 | 13 | 57 | 100 |
| 2.00 | 5 | 13 | 57 | 104 |
| 3.00 | 5 | 13 | 57 | 100 |

Increasing the number of queues at the output interfaces can lead to more variability in the service time for EF packets so we carried out an experiment varying the number of queues at each output port. We fixed the number of flows in the aggregate to eight and used the minimal 1.03 service-to-arrival ratio. Results are shown in figure 5 and table 3. Figure 5 includes PQ with 8 flows as a baseline.

Table 3: Variation in Jitter with Number of Queues at Output Interface: Service-to-arrival ratio is 1.03, 8 flow aggregate

| # EF flows | 1500 byte packet 50th % | 90th % |
|---|---|---|
| PQ (8) | 1 | 3 |
| 2 | 7 | 21 |
| 4 | 7 | 21 |
| 6 | 8 | 22 |
| 8 | 10 | 23 |

It appears that most jitter for WRR is low and can be reduced by a proper choice of the EF queue's WRR share of the output link with respect to its subscribed rate. As noted, WRR is a worst case while PQ is the best case. Other possibilities include WFQ or CBQ with a fixed rate limit for the EF queue but giving it priority over other queues. We expect the latter to have performance nearly identical with PQ though future simulations are needed to verify this. We have not yet systematically explored effects of hop count, EF allocations other than 30% of the link bandwidth, or more complex topologies. The information in this section is not part of the EF PHB definition but provided simply as background to guide implementers.

### A.3.2 VLL service

We used simulation to see how well a VLL service built from the EF PHB behaved, that is, does it look like a `leased line' at the subscribed rate. In the simulations of the last section, none of the EF packets were dropped in the network and the target rate was always achieved for those CBR sources. However, we wanted to see if VLL really looks like a `wire' to a TCP using it. So we simulated long-lived FTPs using a VLL service. Table 4 gives the percentage of each link allocated to EF traffic (bandwidths are lower on the links with fewer EF microflows), the subscribed VLL rate, the average rate for the same type of sender-receiver pair connected by a full duplex dedicated link at the subscribed rate and the average of the VLL flows for each simulation (all sender-receiver pairs had the same value). Losses only occur when the input shaping buffer overflows but not in the network. The target rate is not achieved due to the well-known TCP behavior.

Table 4: Performance of FTPs using a VLL service

| % link to EF | Average delivered rate (Kbps) Subscribed | Dedicated | VLL |
|---|---|---|---|
| 20 | 100 | 90 | 90 |
| 40 | 150 | 143 | 143 |
| 60 | 225 | 213 | 215 |

## Full Copyright Statement



## Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.