

Groupe de travail Réseau
Request for Comments : 3246
 RFC rendue obsolète : 2598
 Catégorie : En cours de normalisation

B. Davie, A. Charny, Cisco Systems, Inc.
 J.C.R. Bennett, Motorola
 K. Benson, Tellabs
 J.Y. Le Boudec, EPFL
 W. Courtney, TRW
 S. Davari, PMC-Sierra
 V. Firoiu, Nortel Networks
 D. Stiliadis, Lucent Technologies
 mars 2002

Traduction Claude Brière de L'Isle

Comportement par bond (PHB) de transmission accélérée

Statut de ce mémoire

Le présent document spécifie un protocole Internet en cours de normalisation pour la communauté de l'Internet, et appelle à des discussions et des suggestions pour son amélioration. Prière de se reporter à l'édition actuelle du STD 1 "Normes des protocoles officiels de l'Internet" pour connaître l'état de normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

Notice de copyright

Copyright (c) 2001 The Internet Society. Tous droits réservés.

Résumé

Le présent document définit un comportement pas bond (PHB, *per-hop behavior*) appelé Transmission accélérée (EF, *Expedited Forwarding*). Le PHB est un bloc de construction de base dans l'architecture des services différenciés. EF est destiné à fournir un bloc de construction pour les services à faible délai, faible gigue et faible perte en garantissant que l'agrégat EF est servi à un certain taux configuré. Le présent document rend obsolète la RFC 2598.

Table des Matières

1. Introduction.....	1
1.1 Relations avec la RFC 2598.....	2
2. Définition du PHB EF.....	2
2.1 Description intuitive de EF.....	2
2.2 Définition formelle du PHB EF.....	3
2.3 Figures de mérite.....	4
2.4 Délai et gigue.....	5
2.5 Perte.....	5
2.6 Désordre de microflux.....	5
2.7 Codet recommandé pour ce PHB.....	5
2.8 Mutabilité.....	6
2.9 Tunnelage.....	6
2.10 Interaction avec les autres PHB.....	6
3. Considérations pour la sécurité.....	6
4. Considérations pour l'IANA.....	6
5. Remerciements.....	6
6. Références.....	6
Appendice Exemples de mise en œuvre.....	7
Adresse des auteurs.....	8
Déclaration de droits de reproduction.....	8

1. Introduction

Les nœuds de réseau qui mettent en œuvre les améliorations de services différenciés à IP utilisent un codet dans l'en-tête IP pour choisir un comportement par bond (PHB, *per-hop behavior*) comme traitement de transmission spécifique pour ce paquet [3], [4]. Le présent mémoire décrit un PHB particulier appelé transmission accélérée (EF, *expedited forwarding*).

L'intention du PHB EF est de fournir un bloc de construction pour les service à faible perte, faible délai, et faible gigue. Les détails de la façon exacte dont construire de tels services sortent du domaine d'application de la présente spécification.

Les causes dominantes des délais dans les réseaux de paquets sont des délais de propagation fixes (par exemple, ceux qui résultent du délai de la vitesse de la lumière) sur les liaisons à grande distance, et des délais de mise en file d'attente dans les commutateurs et les routeurs. Comme les délais de propagation sont une propriété fixée de la topologie, les délais et la gigue sont minimisés lorsque les délais de mise en file d'attente sont minimisés. Dans ce contexte, la gigue est définie comme la variation entre le délai maximum et minimum. L'intention du PHB EF est de fournir un PHB dans lequel des paquets convenablement marqués rencontrent normalement des files d'attentes courtes ou vides. De plus, si les files d'attente restent courtes par rapport à l'espace de mémoire tampon disponible, la perte de paquet est aussi maintenue au minimum.

Pour s'assurer que les files d'attente rencontrées par les paquets EF sont normalement courtes, il est nécessaire de s'assurer que le taux de service des paquets EF sur une interface de sortie donnée excède leur taux d'arrivée à cette interface sur des intervalles longs et courts, indépendamment de la charge des autres trafics (non EF). La présente spécification définit un PHB dans lequel les paquets EF ont la garantie de recevoir un service au taux configuré ou mieux et donne un moyen pour quantifier la précision avec laquelle ce taux de service est fourni sur tout intervalle de temps. Il donne aussi un moyen pour quantifier le délai et la gigue maximum qu'un paquet peut rencontrer dans des conditions de fonctionnement encadrées.

Noter que le PHB EF ne définit que le comportement d'un seul nœud. La spécification du comportement d'une collection de nœuds est en dehors du domaine d'application du présent document. Une spécification de comportement par domaine (PDB, *Per-Domain Behavior*) [7] peut fournir de telles informations.

Lorsque un nœud conforme à DS revendique de mettre en œuvre le PHB EF, la mise en œuvre DOIT se conformer à la spécification donnée dans le présent document. Cependant, le PHB EF n'est pas une partie obligatoire de l'architecture des services différenciés – il N'EST PAS EXIGÉ d'un nœud qu'il mette en œuvre le PHB EF afin d'être considéré comme conforme à DS.

Dans le présent document, les mots clé "DOIT", "NE DOIT PAS", "EXIGÉ", "DEVRAIT" NE DEVRAIT PAS", "RECOMMANDÉ", "NON RECOMMANDÉ", "PEUT", et "FACULTATIF" doivent être interprétés comme décrit dans la RFC2119 [2]

1.1 Relations avec la RFC 2598

Le présent document remplace la RFC 2598 [1]. La principale différence est qu'il ajoute un formalisme mathématique pour donner une définition plus rigoureuse du comportement décrit. Les raisons complètes en sont données dans [6].

2. Définition du PHB EF

2.1 Description intuitive de EF

Intuitivement, la définition de EF est simple : le taux auquel le trafic EF est servi à une interface de sortie donnée devrait être au moins le taux configuré R , sur un intervalle convenablement défini, indépendamment de la charge offerte de trafic non EF à cette interface. Deux difficultés surviennent lorsque on essaye de formaliser cette intuition :

- il est difficile de définir l'échelle de temps appropriée pour mesurer R . En le mesurant sur une courte échelle de temps on peut introduire des erreurs d'échantillonnage ; sur une échelle de temps longue, on peut permettre une gigue excessive.
- le trafic EF ne peut clairement pas être servi au taux R si il n'y a pas de paquets EF qui attendent d'être servis, mais il peut être impossible de déterminer de l'extérieur si les paquets EF sont réellement en attente d'être servis par le programmeur de sortie. Par exemple, si un paquet EF est entré dans le routeur et n'est pas sorti, il peut attendre d'être servi, ou il peut simplement avoir subi un délai de traitement ou de transmission au sein du routeur.

La définition formelle ci-dessous tient compte de ces questions. Elle suppose que les paquets EF devraient idéalement être servis au taux R ou plus vite, et tendre vers la déviation du temps de départ réel de chaque paquet par rapport au temps de départ "idéale" de ce paquet. On définit le temps de départ d'un paquet comme le moment où le dernier bit de ce paquet quitte le nœud. Le temps de départ "idéale" de chaque paquet EF est calculé de façon itérative.

Dans le cas où un paquet EF arrive à un appareil lorsque le précédent paquet EF est déjà parti, le calcul du temps de départ idéal est simple. Servir le paquet devrait (idéalement) commencer aussitôt qu'il arrive, de sorte que le temps de départ idéal est simplement l'heure d'arrivée plus le temps idéal de transmission du paquet au taux R . Pour un paquet de longueur L_j , ce temps de transmission au taux configuré R est L_j/R . (Bien sûr, un paquet réel va normalement être transmis au débit de ligne une fois que sa transmission commence réellement, mais nous calculons ici le comportement cible idéal ; le service

idéal a lieu au taux R.)

Dans le cas où un paquet EF arrive à un appareil qui contient encore des paquets EF qui attendent d'être servis, le calcul de l'heure de départ idéale est plus compliqué. Deux cas sont à considérer. Si le départ précédant le (j-1^{ème}) est survenu après sa propre heure de départ idéale, le programmeur fonctionne alors "en retard". Dans ce cas, l'heure idéale pour commencer à servir le nouveau paquet est l'heure de départ idéale du précédent paquet (j-1^{ème}), ou l'heure d'arrivée du nouveau paquet, selon ce qui est le plus tardif, parce qu'on ne peut pas s'attendre à ce qu'un paquet commence à être servi avant qu'il arrive. Si le (j-1^{ème}) départ précédent est survenu avant sa propre heure de départ idéale, le programmeur fonctionne alors "en avance". Dans ce cas, le service du nouveau paquet devrait commencer à l'heure réelle de départ du paquet précédent.

Une fois qu'on connaît l'heure à laquelle le service du j^{ème} paquet devrait (idéalement) commencer, l'heure idéale de départ du j^{ème} paquet est L_j/R secondes plus tard. Donc, nous sommes capables d'exprimer l'heure de départ idéale du j^{ème} paquet en termes d'heure d'arrivée du j^{ème} paquet, l'heure réelle de départ du j-1^{ème} paquet, et l'heure idéale de départ du j-1^{ème} paquet. Les équations eq_1 et eq_2 du paragraphe 2.2 retracent cette relation.

Alors que la définition originale d'EF ne fournissait aucun moyen de garantir le délai d'un paquet EF individuel, cette propriété peut être souhaitée. Pour cette raison, les équations du paragraphe 2.2 comportent deux parties : un ensemble d'équations de "comportements agrégés" et un ensemble d'équations à "connaissance de l'identité du paquet". Les équations de comportements agrégés (eq_1 et eq_2) décrivent simplement les propriétés du service livré à l'agrégat EF par l'appareil. Les équations à "connaissance de l'identité du paquet" (eq_3 et eq_4) permettent de calculer les limites de délai d'un paquet individuel connaissant les conditions de fonctionnement de l'appareil. La signification de ces deux ensembles d'équations est exposée plus en détails au paragraphe 2.2. Noter que ces deux ensembles d'équations donnent deux façons de caractériser le comportement d'un seul appareil, et non deux différents modes de comportement.

2.2 Définition formelle du PHB EF

Un nœud qui prend en charge EF sur une interface I à un taux configura R DOIT satisfaire aux équations suivantes :

$$d_j \leq f_j + E_a \text{ pour tout } j > 0 \quad (\text{eq}_1)$$

où f_j est défini de façon itérative par $f_0 = 0$, $d_0 = 0$

$$f_j = \max(a_j, \min(d_{j-1}, f_{j-1})) + l_j/R, \text{ pour tout } j > 0 \quad (\text{eq}_2)$$

Dans cette définition :

- d_j est l'heure à laquelle le dernier bit du j^{ème} paquet EF à partir quitte réellement le nœud à partir de l'interface I.
- f_j est l'heure de départ cible par le j^{ème} paquet EF pour quitter I, l'heure "idéale" à laquelle ou avant laquelle le dernier bit de ce paquet devrait quitter le nœud.
- a_j est l'heure à laquelle le dernier bit du j^{ème} paquet EF destiné à la sortie I arrive réellement au nœud.
- l_j est la taille (en bits) du j^{ème} paquet EF qui part de I.
- l_j est mesuré sur le datagramme IP (en-tête IP plus charge utile) et n'inclut aucune redondance de couche inférieure (par exemple de couche MAC).
- R est le taux d'EF configuré à la sortie I (en bit/seconde).
- E_a est le terme d'erreur pour le traitement de l'agrégat EF. Noter que E_a représente le pire cas de déviation entre l'heure de départ réelle d'un paquet EF et l'heure de départ idéale du même paquet, c'est-à-dire que E_a donne une limite supérieure de $(d_j - f_j)$ pour tout j.
- d_0 et f_0 ne se réfèrent pas à un départ réel de paquet mais sont utilisés uniquement pour les besoins de la récurrence. L'origine des temps devrait être choisie de telle sorte qu'aucun paquet EF ne soit dans le système au temps 0.
- pour les définitions de a_j et d_j , le "dernier bit" du paquet inclut l'en-queue de couche 2 s'il est présent, parce que un paquet ne peut généralement pas être considéré comme disponible à la transmission tant qu'un tel en-queue n'a pas été reçu.

Un nœud conforme à EF DOIT être capable de se caractériser par la gamme de valeurs R possibles qu'il peut prendre en charge sur chacune de ses interfaces tout en se conformant à ces équations, et la valeur de E_a qu'il peut satisfaire sur chaque interface. R peut être le taux de ligne ou moins. E_a PEUT être spécifié comme la plus mauvaise valeur pour toutes les valeurs possibles de R ou PEUT être exprimé en fonction de R.

Noter aussi que, comme un nœud peut avoir plusieurs entrées et une programmation interne complexe, le j^{ème} paquet EF qui arrive au nœud destiné à une certaine interface ne peut pas être le j^{ème} paquet EF à partir de cette interface. C'est en ce sens que eq_1 et eq_2 n'ont pas connaissance de l'identité du paquet.

De plus, un nœud qui prend en charge EF sur une interface I à un taux configuré R DOIT satisfaire aux équations suivantes :

$$D_j \leq F_j + E_p \text{ pour tout } j > 0 \quad (\text{eq}_3)$$

où F_j est défini par itération par

$$F_0 = 0, D_0 = 0$$

$$F_j = \max(A_j, \min(D_{j-1}, F_{j-1})) + L_j/R, \text{ pour tout } j > 0 \quad (\text{eq}_4)$$

Dans cette définition :

- D_j est l'heure de départ réelle du paquet EF individuel qui est arrivé au nœud à destination de l'interface I au moment A_j , c'est-à-dire, soit un paquet qui est le $j^{\text{ème}}$ paquet EF destiné à I et arrive au nœud via toute entrée, D_j est le moment où le dernier bit de ce paquet individuel quitte réellement le nœud par l'interface I.
- F_j est l'heure de départ cible du paquet EF individuel qui est arrivé au nœud et est destiné à l'interface I au moment A_j .
- A_j est l'heure à laquelle le dernier bit du $j^{\text{ème}}$ paquet EF destiné à arriver à la sortie I arrive réellement au nœud.
- L_j est la taille (en bits) du $j^{\text{ème}}$ paquet EF arrivé au nœud et qui est destiné à la sortie I. L_j est mesuré sur le datagramme IP (en-tête IP plus charge utile) sans inclure de redondance de couche inférieure (par exemple, de couche MAC).
- R est le taux EF configuré à la sortie I (en bit/seconde).
- E_p est le terme d'erreur pour le traitement des paquets EF individuels. Noter que E_p représente le pire cas de déviation entre l'heure réelle de départ d'un paquet EF et l'heure de départ idéale de ce même paquet, c'est-à-dire que E_p donne une limite supérieure de $(D_j - F_j)$ pour tout j.
- D_0 et F_0 ne se réfèrent pas à un départ réel de paquet mais sont utilisés pour les seuls besoins de la récurrence. L'origine des temps devrait être choisie de sorte qu'aucun paquet EF ne soit dans le système au moment 0.
- pour les définitions de A_j et D_j , le "dernier bit" du paquet inclut l'en-queue de couche 2 s'il est présent, parce que un paquet ne peut généralement pas être considéré comme disponible à la transmission tant qu'un tel en-queue n'a pas été reçu.

C'est le fait que D_j et F_j se réfèrent aux heures de départ du $j^{\text{ème}}$ paquet qui arrive qui fait que eq_3 et eq_4 ont connaissance de l'identité du paquet. C'est la distinction critique entre les deux dernières et les deux premières équations.

Un nœud conforme à EF DEVRAIT être capable de se caractériser par la gamme de valeur R possibles qu'il peut prendre en charge sur chacune de ses interfaces tout en se conformant à ces équations, et par la valeur de E_p qui peut être satisfaite sur chaque interface. E_p PEUT être spécifié comme plus mauvaise valeur pour toutes les valeurs possibles de R ou PEUT être exprimé en fonction de R. Une valeur E_p de "indéfini" PEUT être spécifiée. Pour un exposé des situations dans lesquelles E_p peut être indéfini, voir l'Appendice et [6].

Pour les besoins des essais de conformité à ces équations, il peut être nécessaire de traiter les arrivées de paquet sur des interfaces différentes qui sont étroitement espacées dans le temps. Si deux paquets EF, ou plus, destinés à la même interface de sortie arrivent (d'entrées différentes) presque au même moment et si la différence entre leurs heures d'arrivée ne peut être mesurée, il est alors acceptable d'utiliser une méthode de départage aléatoire pour décider quel paquet est arrivé "premier".

2.3 Figures de mérite

E_a et E_p peuvent être vus comme des "figures de mérite" d'un appareil. Une plus petite valeur de E_a signifie que l'appareil sert l'agrégat EF plus en douceur au taux R sur des échelles de temps relativement courtes, tandis qu'une valeur supérieure de E_a implique un programmeur plus sporadique qui ne sert l'agrégat EF au taux R que lorsque il est mesuré sur des intervalles de temps plus longs. Un appareil avec un E_a plus grand peut "tomber au-delà" du taux de service R idéal de beaucoup plus qu'un appareil avec un E_a plus petit.

Une valeur inférieure de E_p implique une limite plus serrée sur le délai subi par un paquet individuel. Les facteurs qui pourraient conduire à un E_p supérieur pourraient inclure un grand nombre d'interfaces d'entrée (car un paquet EF pourrait arriver juste derrière un grand nombre de paquets EF arrivés sur d'autres interfaces) ou pourraient être dus à des particularités internes du programmeur (par exemple une programmation par flux au sein de l'agrégat EF).

On observe que les facteurs qui augmentent E_a tels que ceux qui sont notés ci-dessus augmentent aussi E_p , et que E_p est donc normalement supérieur ou égal à E_a . En résumé, E_a est une mesure de la déviation du service idéal de l'agrégat EF au taux R, alors que E_p mesure à la fois le service idéal et le traitement non FIFO des paquets au sein de l'agrégat.

Pour un exposé complémentaire sur ces questions, voir l'Appendice et [6].

2.4 Délai et gigue

Étant données une valeur connue de E_p et une connaissance des limites du trafic EF offertes à une interface de sortie données, additionnée sur toutes les interfaces d'entrée, il est possible de tracer les limites du délai et de la gigue que vont rencontrer le trafic EF qui quitte le nœud via cet interface. La limite de délai est

$$D = B/R + E_p \quad (\text{eq}_5)$$

où

- R est le taux de service EF configuré sur l'interface de sortie,
- la charge offerte totale de trafic EF destiné à l'interface de sortie, sommée sur toutes les interfaces de sortie, est limitée par un baquet de jetons de taux $r \leq R$ et de profondeur B .

Comme le délai minimum à travers l'appareil est évidemment d'au moins zéro, D donne aussi une limite sur la gigue. Pour donner une limite plus serrée sur la gigue, la valeur de E_p pour un appareil PEUT être spécifiée comme deux composants distincts tels que

$$E_p = E_{\text{fixe}} + E_{\text{variable}}$$

où E_{fixe} représente le délai minimum qui peut être subi par un paquet EF à travers le nœud.

2.5 Perte

Le PHB EF est destiné à constituer un bloc de construction pour les services à faible perte. Cependant, dans des conditions de charge de trafic RF suffisamment élevées (y compris de grosses salves inattendues provenant de nombreuses entrées à la fois) tout appareil avec des mémoires tampon de dimensions finies peut avoir besoin d'éliminer des paquets. Donc, il doit être possible d'établir si un appareil se conforme à la définition de EF même lorsque certains paquets sont perdus. Cela se fait en effectuant un essai de conformité "hors ligne" aux équations 1 à 4. Après avoir observé une séquence de paquets entrant et quittant le nœud, les paquets qui ne sont pas sortis sont supposés perdus et sont idéalement retirés du flux entrant. Les paquets restants constituent maintenant le flux d'arrivée (les a_j) et les paquets qui ont quitté le nœud constituent le flux de départ (les d_j). La conformité aux équations peut donc être vérifiée en ne considérant que les paquets qui ont réussi à passer à travers le nœud.

De plus, pour aider à satisfaire les objectifs de perte de EF, un nœud PEUT être caractérisé par la région de fonctionnement dans laquelle des pertes de EF dues à l'encombrement ne surviendront pas. Cela PEUT être spécifié, en utilisant un baquet de jetons de taux $r \leq R$ et une taille de salve de B , comme somme du trafic à travers toutes les entrées à une interface de sortie donnée qui peut être toléré sans pertes.

Dans le cas où des pertes surviennent quand même, la spécification de quels paquets seront perdus sort du domaine d'application du présent document. Cependant, il est exigé que les paquets qui ne sont pas éliminés DOIVENT se conformer aux équations du paragraphe 2.2.

2.6 Désordre de microflux

Les paquets qui appartiennent à un seul microflux au sein de l'agrégat EF qui passe à travers un appareil NE DEVRAIENT PAS subir de réarrangement en fonctionnement normal de l'appareil.

2.7 Codet recommandé pour ce PHB

Le codet 101110 est RECOMMANDÉ pour le PHB EF.

2.8 Mutabilité

Les paquets marqués pour PHB EF NE PEUVENT être remarqués à la frontière d'un domaine DS que par d'autres codets qui satisfont au PHB EF. Les paquets marqués pour les PHB EF NE DEVRAIENT PAS être déchus ou promus à un autre PHB par un domaine DS.

2.9 Tunnelage

Lorsque des paquets EF sont tunnelés, les paquets qui portent le tunnel DEVRAIENT être marqués comme EF. Un exposé complet sur les questions de tunnelage est présenté dans [5].

2.10 Interaction avec les autres PHB

D'autres PHB et groupes de PHB peuvent être déployés dans le même nœud ou domaine avec le PHB EF. Les équations du paragraphe 2.2 DOIVENT tenir pour un nœud indépendamment de la quantité de trafic non EF qui lui est offert.

Si le PHB EF est mis en œuvre par un mécanisme qui permet une préemption illimitée des autres trafics (par exemple, une file d'attente prioritaire), la mise en œuvre DOIT inclure un moyen pour limiter les dommages que le trafic EF pourrait infliger aux autres trafics (par exemple, un limiteur de taux de paquet de jetons). Le trafic qui excède cette limite DOIT être éliminé. Ce taux EF maximum, et le cas échéant, cette taille de salve, DOIT être réglable par un administrateur de réseau (utilisant tout mécanisme accepté par le nœud pour une configuration non volatile).

3. Considérations pour la sécurité

Pour se protéger contre les attaques de déni de service, la bordure d'un domaine DS DEVRAIT réguler strictement tous les paquets marqués EF à un taux négocié avec le domaine amont adjacent. Les paquets en excédant du taux négocié DEVRAIENT être éliminés. Si deux domaines adjacents n'ont pas négocié un taux EF, le domaine aval DEVRAIT utiliser le taux 0 (c'est-à-dire, éliminer tous les paquets marqués EF).

De plus, le conditionnement de trafic à l'entrée d'un domaine DS DOIT s'assurer que seuls les paquets qui ont des DSCP qui correspondent à un PHB EF lorsque ils entrent dans le domaine DS sont marqués avec un DSCP qui correspond à EF à l'intérieur du domaine DS. Un tel comportement est celui qui est exigé par l'architecture des services différenciés [4]. Il protège contre les attaques de déni de service et de vol de service qui exploitent les DSCP qui ne sont pas identifiés dans une spécification de conditionnement de trafic fournie à une interface d'entrée, mais qui se transpose dans EF à l'intérieur du domaine DS.

4. Considérations pour l'IANA

Le présent document alloue un codet, 101110, dans le pôle 1 de l'espace de codes défini par [3].

5. Remerciements

Le présent document est le résultat de la collaboration et de discussions avec un grand nombre de personnes. En particulier, Fred Baker, Angela Chiu, Chuck Kalmanek, et K. K. Ramakrishnan ont fait des contributions significatives à la nouvelle définition d'EF. John Wroclawski a fourni de nombreux commentaires utiles aux auteurs. Le présent document a beaucoup emprunté à la définition originale du PHB EF de Jacobson, Nichols, et Poduri. Il a été aussi largement influencé par le travail de l'équipe EFRESOLVE de Armitage, Casati, Crowcroft, Halpern, Kumar, et Schnizlein.

6. Références

- [1] V. Jacobson, K. Nichols, K. Poduri, "PHB Transmission accélérée", RFC2598, juin 1999. (*Obsolète, voir RFC3246*)
- [2] S. Bradner, "Mots clés à utiliser dans les RFC pour indiquer les niveaux d'exigence", RFC2119, BCP 14, mars 1997.

- [3] K. Nichols, S. Blake, F. Baker et D. Black, "Définition du champ Services différenciés (DS Field) dans les en-têtes IPv4 et IPv6", RFC2474, décembre 1998. (MàJ par RFC3168, RFC3260) (P.S.)
- [4] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang et W. Weiss, "Architecture pour services différenciés", RFC2475, décembre 1998. (MàJ par RFC3260)
- [5] D. Black, "Services différenciés et tunnels", RFC2983, octobre 2000. (Information)
- [6] A. Charny et autres, "Informations supplémentaires pour la nouvelle définition du comportement par bond de transmission accélérée", RFC3247, mars 2002. (Information)
- [7] K. Nichols, B. Carpenter, "Définition des comportements de services différenciés par domaine et règles de leur spécification", RFC3086, avril 2001. (Information)

Appendice Exemples de mise en œuvre

Le présent appendice ne fait pas partie de la spécification normative de EF. Cependant, il est inclus ici comme source possible d'informations utiles pour les développeurs.

Divers facteurs dans la mise en œuvre d'un nœud qui prend en charge EF vont influencer les valeurs de E_a et E_p . Ces facteurs sont exposés plus en détail dans [6], et incluent aussi bien les programmeurs de sortie que la conception interne d'un appareil.

Une file d'attente prioritaire est généralement considérée comme l'exemple canonique d'une mise en œuvre de EF. Un appareil à mémoire tampon de sortie "parfaite" (c'est à dire, qui livre les paquets immédiatement à la file d'attente de sortie appropriée) avec une file d'attente prioritaire pour le trafic EF présenterait à la fois un E_a et un E_p faibles. On notera que le principal facteur qui influence E_a sera l'incapacité à préempter un paquet non EF à la taille de la MTU qui a juste commencé sa transmission au moment où arrive un paquet EF à l'interface de sortie, plus tout délai supplémentaire qui pourrait être causé par des files d'attente non préemptables entre la file d'attente prioritaire et l'interface physique. E_p sera principalement influencé par le nombre d'interfaces.

Un autre exemple d'une mise en œuvre de EF est un programmeur à round robin pondéré. Une telle mise en œuvre ne sera normalement pas capable d'accepter des valeurs de R aussi élevées que la vitesse de la liaison, parce que le taux maximum auquel le trafic EF peut être servi en présence de concurrence de trafic sera affecté par le nombre d'autres files d'attente et par les pondérations qui leurs sont affectées. De plus, une telle mise en œuvre aura vraisemblablement une valeur de E_a supérieure à celle d'une mise en œuvre de file d'attente prioritaire, toutes choses égales par ailleurs, par suite du temps passé à servir les files d'attente non EF par le programmeur round robin.

Finalement, il est possible de mettre en œuvre des algorithmes de programmation hiérarchiques, qui fassent qu'un certain algorithme de programmation non FIFO fonctionne sur des sous flux au sein de l'agrégat EF, tandis que l'agrégat EF serait globalement servi avec une priorité élevée ou avec une grosse pondération par le programmeur de niveau supérieur. Un tel algorithme pourrait effectuer une programmation par entrée ou par microflux au sein de l'agrégat EF, par exemple. Parce que de tels algorithmes conduisent à un service non FIFO au sein de l'agrégat EF, la valeur de E_p pour de tels algorithmes peut être supérieure à celle d'autres mises en œuvre. Pour certains programmeurs de ce type, il peut être difficile de fournir une limite significative de E_p qui tienne pour tout schéma d'arrivée de trafic, et donc une valeur de "indéfini" peut être la plus appropriée.

On devrait noter qu'il est assez acceptable pour un domaine Diffserv de fournir plusieurs instances de EF. Chaque instance devrait être caractérisable par les équations du paragraphe 2.2 de la présente spécification. Les effets d'avoir plusieurs instances de EF sur les valeurs E_a et E_p de chaque instance va dépendre considérablement de la façon dont les multiples instances sont mises en œuvre. Par exemple, dans un programmeur à plusieurs niveaux de priorité, une instance de EF qui n'est pas à la priorité supérieure peut subir une attente sur des périodes relativement longues lorsque elle ne reçoit aucun service alors que des instances de EF de priorité supérieure sont servies. Il en résulterait des valeurs de E_a et E_p relativement élevées. À l'opposé, dans un programmeur de type WFQ, chaque instance de EF serait représentée par une file d'attente servie à un taux configuré et les valeurs de E_a et de E_p pourraient être similaires à celle d'une instance EF seule.

Adresse des auteurs

Bruce Davie	Anna Charny	Jon Bennett	Kent Benson
Cisco Systems, Inc.	Cisco Systems	Motorola	Tellabs Research Center
300 Apollo Drive	300 Apollo Drive	3 Highwood Drive East	3740 Edison Lake Parkway #101
Chelmsford, MA, 01824	Chelmsford, MA 01824	Tewksbury, MA 01876	Mishawaka, IN 46545
mél : bsd@cisco.com	mél : acharny@cisco.com	mél : jrb@motorola.com	mél : Kent.Benson@tellabs.com

Jean-Yves Le Boudec	Bill Courtney	Shahram Davari
ICA-EPFL, INN	TRW	PMC-Sierra Inc
Ecublens, CH-1015	Bldg. 201/3702	411 Legget Drive
Lausanne-EPFL, Suisse	One Space Park	Ottawa, ON K2K 3C9, Canada
mél : jean-yves.leboudec@epfl.ch	Redondo Beach, CA 90278	mél : shahram_davari@pmc-sierra.com
	mél : bill.courtney@trw.com	

Victor Firoiu	Dimitrios Stiliadis
Nortel Networks	Lucent Technologies
600 Tech Park	101 Crawfords Corner Road
Billerica, MA 01821	Holmdel, NJ 07733
mél : vfiroiu@nortelnetworks.com	mél : stiliadi@bell-labs.com

Déclaration de droits de reproduction

Copyright (C) The Internet Society (2001). Tous droits réservés.

Le présent document et ses traductions peuvent être copiés et fournis aux tiers, et les travaux dérivés qui les commentent ou les expliquent ou aident à leur mise en œuvre peuvent être préparés, copiés, publiés et distribués, en tout ou partie, sans restriction d'aucune sorte, pourvu que la déclaration de droits de reproduction ci-dessus et le présent et paragraphe soient inclus dans toutes ces copies et travaux dérivés. Cependant, le présent document lui-même ne peut être modifié d'aucune façon, en particulier en retirant la notice de droits de reproduction ou les références à la Internet Society ou aux autres organisations Internet, excepté autant qu'il est nécessaire pour les besoins du développement des normes Internet, auquel cas les procédures de droits de reproduction définies dans les procédures des normes Internet doivent être suivies, ou pour les besoins de la traduction dans d'autres langues que l'anglais.

Les permissions limitées accordées ci-dessus sont perpétuelles et ne seront pas révoquées par la Internet Society ou ses successeurs ou ayant droits.

Le présent document et les informations y contenues sont fournies sur une base "EN L'ÉTAT" et le contributeur, l'organisation qu'il ou elle représente ou qui le/la finance (s'il en est), la INTERNET SOCIETY et la INTERNET ENGINEERING TASK FORCE déclinent toutes garanties, exprimées ou implicites, y compris mais non limitées à toute garantie que l'utilisation des informations ci-encloses ne violent aucun droit ou aucune garantie implicite de commercialisation ou d'aptitude à un objet particulier.

Remerciement

Le financement de la fonction d'édition des RFC est actuellement fourni par l'Internet Society.