

RFC 2474 : Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers

Stéphane Bortzmeyer
<stephane+blog@bortzmeyer.org>

Première rédaction de cet article le 13 février 2008. Dernière mise à jour le 18 février 2008

Date de publication du RFC : Décembre 1998

<http://www.bortzmeyer.org/2474.html>

La question du **traitement différencié** des paquets IP occupe beaucoup de monde depuis les débuts de l'Internet. En gros, il s'agit de savoir si les routeurs et autres équipements intermédiaires vont traiter tous les paquets IP de la même manière, où bien si certains auront la priorité, ou en tout cas un traitement particulier selon certains critères. Si la question a un aspect politique, lié à la revendication de neutralité du réseau, pour savoir qui va décider quels paquets seront sacrifiés, notre RFC ne couvre que l'aspect technique : il change complètement un des champs du paquet IP, pour tenir compte de l'échec qu'avait été ce champ "*Type Of Service*".

Avant de mettre en œuvre des traitements différenciés, encore faut-il savoir à quels paquets les appliquer. Dans le RFC original sur IPv4, le RFC 791¹, un champ du paquet, **TOS**, "*Type Of Service*" était réservé au marquage du paquet. Il permettait d'indiquer la priorité souhaitée et les critères les plus importants pour ce paquet (débit, coût, latence, etc). La norme Posix prévoit des moyens pour les applications de définir ces champs (un programme comme `echoping` <<http://echoping.sourceforge.net/>> les met en œuvre avec ses options `-p` et `-P`). Voici un exemple en C de définition du champ TOS avec `setsockopt` :

```
result = setsockopt(mysocket, IPPROTO_IP,
                   IP_TOS, IPTOS_THROUGHPUT, sizeof(int));
/* IPTOS_THROUGHPUT vaut 0x08 et était censé maximiser le débit. Cf. RFC 791,
   section 3.1 */
```

1. Pour voir le RFC de numéro NNN, <https://www.ietf.org/rfc/rfcNNN.txt>, par exemple <https://www.ietf.org/rfc/rfc791.txt>

Pour voir les capacités des différents systèmes d'exploitation à fixer ces valeurs, on peut regarder mon article sur la fixation d'options IP depuis un programme <<http://www.bortzmeyer.org/ip-header-set.html>>.

Ce mécanisme n'a eu aucun succès, peu d'applications fixent ce champ et peu de routeurs le traitaient comme spécifié. Il y a plusieurs raisons à cet échec, et toutes ne sont pas techniques, loin de là. Par exemple, comment empêcher une application de toujours demander la plus forte priorité? (La question est discutée dans la section 7.1 de notre RFC.)

Parmi les raisons techniques, il y avait le fait que le mécanisme du RFC 791 était trop rigide. Par exemple, les critères exprimables étaient en nombre limité et trop vagues. Une nouvelle architecture pour la qualité de service, "*Differentiated Services*", connue sous l'abréviation de "*Diffserv*" a été définie dans le RFC 2475. Ses conséquences pratiques sur le format des paquets font l'objet de notre RFC 2474, qui met donc à jour le RFC 791 et qui s'applique également à son équivalent IPv6, le RFC 2460, pour la définition du champ "*Traffic class*" (équivalent IPv6 du champ TOS).

Le champ TOS de IPv4 change donc de nom. Il devient DS ("*Differentiated Services*") et la sémantique n'est plus fixée par le RFC. Un TOS de 0×08 voulait toujours dire « Je cherche à maximiser le débit » alors qu'un DS de 0×08 aura une signification donnée par les routeurs du domaine administratif où le paquet passe. Aux frontières de ces domaines, par exemple pour passer d'un opérateur à un autre, il faudra changer le champ (section 2 du RFC) ou bien utiliser des valeurs attribuées par un accord entre opérateurs (Diffserv définit un mécanisme, pas une politique). La section 1 de notre RFC prévient que, si le "*forwarding*" de chaque paquet est un problème bien connu, l'attribution et la gestion de politiques de qualité de service est un problème bien plus ouvert. Neuf ans après la sortie du RFC, il ne semble pas en voie de résolution. La même section 1 rappelle que des politiques simples et appliquées statiquement sont peut être la meilleure solution.

La section 3 décrit le nouveau champ DS. Voici l'ancien champ ToS, sur huit bits, du RFC 791 :

```

      0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
|           | D | T | R | 0 | 0 |
| PRECEDENCE | | | | | | |
+---+---+---+---+---+---+---+

```

Et voici le nouveau champ DS ("*Differentiated Services*") :

```

      0   1   2   3   4   5   6   7
+---+---+---+---+---+---+---+---+
|           | DSCP | CU |
+---+---+---+---+---+---+---+

```

DSCP: differentiated services codepoint
CU: currently unused

(La partie réservée a été remplie ultérieurement par l'ECN du RFC 3168.) Le champ DSCP contient un **index** dans une table des PHB ("*per-host behavior*") qui sont les traitements que le routeur applique. Des exemples de PHB sont « fait passer ce paquet devant tout le monde dans la queue » ou bien « n'hésite pas à jeter ce paquet si le réseau est congestionné ». Le PHB par défaut (section 4.1) étant le traditionnel « faire de son mieux » des routeurs IP. Certains RFC définissent des PHB comme le RFC 2597 sur le "*Assured Forwarding*".

On voit sur les schémas que, contrairement à l'ancien TOS, le DSCP n'a pas de structure interne. 64 valeurs différentes sont possibles, et les sections 4 et 5 discutent de leur allocation, la section 6 couvrant d'éventuelles standardisations <<https://www.iana.org/assignments/dscp-registry>> de certaines valeurs.

Si on regarde ce qui passe sur le réseau avec tshark (option -v), on verra :

```
Internet Protocol, Src Addr: 172.19.1.1 (172.19.1.1), Dst Addr: 172.19.1.2 (172.19.1.2)
...
  Differentiated Services Field: 0x08 (DSCP 0x02: Unknown DSCP; ECN: 0x00)
    0000 10.. = Differentiated Services Codepoint: Unknown (0x02)
    .... ..0. = ECN-Capable Transport (ECT): 0
    .... ...0 = ECN-CE: 0
```

On voit que le 0x08 mis dans le champ équivaut à un DSCP de 0x02, qui ne correspond pas à une valeur connue de manière standard (d'où le "*Unknown DSCP*").

L'informatique est un monde très conservateur et, neuf ans après la sortie de ce RFC, on continue couramment à utiliser des mots comme TOS, normalement abandonnés (cf. par exemple la bogue echoping #1892544 <https://sourceforge.net/tracker/index.php?func=detail&aid=1892544&group_id=4581&atid=104581>). La norme Posix continue à définir uniquement des constantes pour l'ancien comportement (comme IP.TOS ou IPTOS.THROUGHPUT, utilisés plus haut). Il n'est pas facile de changer une API si utilisée !

Sur Linux, on peut aussi changer la valeur du champ DSCP avec Netfilter et sa commande iptables. Cela se fait en modifiant la table mangle (qui permet de modifier un paquet, pas seulement de l'accepter ou de le refuser), avec la pseudo-chaîne de destination DSCP. Par exemple :

```
# iptables -t mangle -A OUTPUT -p tcp --dport 22 -j DSCP --set-dscp 0x02
```

va changer le DSCP de tous les paquets à destination du port 22. Pour les raisons expliquées plus haut, tcpdump va afficher ce DSCP de 0x2 en 0x8 :

```
17:50:35.477598 IP (tos 0x8, ttl 64, id 1040, offset 0, flags [DF], proto TCP (6), length 60)
  192.168.2.1.36067 > 10.22.58.30.22: Flags [S], cksum 0xe4b2 (correct), seq 3255303173, win 5840, options [ms
```

Pour mettre en œuvre le traitement différencié sur un routeur Cisco, on peut consulter "*Implementing Quality of Service Policies with DSCP*" <<http://www.cisco.com/warp/public/105/dscpvalues.html>>.

Aujourd'hui, DSCP ne semble pas avoir eu beaucoup plus de succès que son prédécesseur. Il est difficile de mesurer le taux de déploiement car tous sont internes à une organisation. Certains RFC comme le RFC 4504 (section 2.12) ont rendu obligatoire l'utilisation de DSCP mais ils n'ont pas forcément été respectés. Si une entreprise qui déploie de la VoIP utilise DSCP en interne pour permettre aux coups de téléphone de passer même lorsqu'un gros transfert de fichiers est en cours, tout en simplifiant la configuration de ses routeurs, cela ne se voit pas à l'extérieur.