

Brève introduction aux architectures à Qualité de Service de l'Internet

Chaput Emmanuel

2011-2012

Notes :

- 1 L'architecture *IntServ*
- 2 L'architecture *DiffServ*
- 3 Références bibliographiques

Notes :

L'architecture *IntServ*

- 1 L'architecture *IntServ*
 - Présentation générale
 - Les éléments de base
 - L'architecture
 - Les classes de service
 - Les paramètres de QoS
 - Une architecture réaliste ?
- 2 L'architecture *DiffServ*
- 3 Références bibliographiques

Notes :

L'architecture IntServ

- Groupe de travail de l'IETF [7]
- Deuxième moitié des années 90
- Constat : possibilité d'intégrer des services sur un réseau paquet
 - Audio
 - Vidéo
 - Temps-réel
 - Données
- Définition des interfaces et signalisation en ce sens

Notes :

Les objectifs INTSERV

- Garantir des paramètres de QoS par flot
 - Un flot = une communication
 - Garanties fermes de bout en bout
- Pour cela, définition et spécification
 - D'une architecture générale [2]
 - D'outils de spécification des paramètres de trafic (TS_{pec}) et des paramètres de QoS ($RSPEC$)
 - D'un protocole de signalisation ($RSVP$ [3])
 - De comportements associés aux classes de service proposées [14, 10]

Notes :

Les objectifs du groupe de travail

Trois grands axes

Définition des services

Définir et documenter un modèle de service fourni aux applications

Définition des interfaces

Expression des besoins applicatifs, des informations fournies aux routeurs et des contraintes sur le réseau local.

Validation des routeurs

Définir des tests (comportementaux) permettant de valider la capacité d'un routeur à supporter le modèle.

Notes :

Les éléments de base

Notion de flux de données

Ensemble des paquets entre une source (IP/port) et une destination.

Réservation de ressources

Réquisition des moyens nécessaires à la garantie d'un service voulu sur un flux déterminé.

Notes :

Les classes de trafic *IntServ*

Trois grandes classes de trafic

- Guaranteed delay [10]
 - Garantie de délai de bout en bout
 - Garantie de débit
 - Garantie de (non) pertes
- Controlled load [14]
 - Équivalent à un best-effort sur un réseau peu chargé
- Best effort
 - Traditionnel

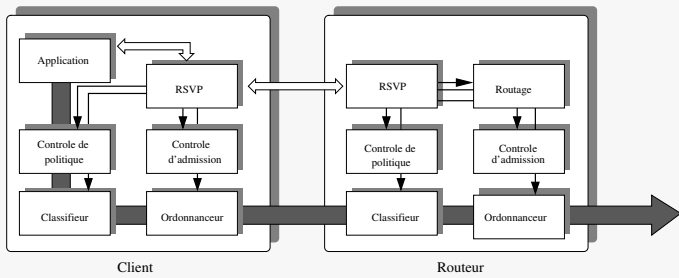
Notes :

L'architecture

- 1 L'architecture *IntServ*
 - Présentation générale
 - Les éléments de base
 - **L'architecture**
 - Les classes de service
 - Les paramètres de QoS
 - Une architecture réaliste ?

Notes :

L'architecture IntServ



Notes :

Les classes de service

- 1 L'architecture IntServ
 - Présentation générale
 - Les éléments de base
 - L'architecture
 - **Les classes de service**
 - Les paramètres de QoS
 - Une architecture réaliste ?

Notes :

La classe de trafic Guaranteed

- Ce qu'elle garanti aux applications [10]
 - Un délai borné et aucune perte due à la congestion pour les paquets conformes
- Pour cela, bien sûr, l'application s'engage à respecter son TSpec
 - Notamment en termes de taille de paquet (pas de fragmentation)
- Chaque routeur doit donc s'assurer qu'il introduira un délai borné
 - Défini par un modèle *token bucket*

Notes :

La classe de trafic *Controlled load*

- Ce qu'elle "garanti" aux applications [14]
 - Une grande proportion des paquets seront transmis (un taux d'erreur proche de celui des supports)
 - La plupart des paquets subiront un temps de transfert ne dépassant pas de beaucoup le temps le plus faible
- Pour cela, bien sûr, l'application s'engage à respecter son T_{Spec}
 - Les paquets hors profil seront traités différemment
 - Par exemple au même titre que le trafic "*best effort*"
- Les routeurs doivent donc accepter ou non les communications en fonction
 - Des T_{Psec} des nouvelles communications
 - Des ressources disponibles
 - Des T_{Spec} ou du comportement observé des communications en cours

Notes :

Les paramètres de QoS

- 1 L'architecture *IntServ*
 - Présentation générale
 - Les éléments de base
 - L'architecture
 - Les classes de service
 - Les paramètres de QoS
 - Une architecture réaliste ?

Notes :

Les spécifications *IntServ*

Deux ensembles de paramètres clairement identifiés

- La spécification des trafics (TSPEC)
 - Défini ce à quoi s'engage le client (l'application)
 - Outil commun à toutes les classes de trafic
- La spécification des besoins (RSPEC)
 - Défini ce que demande le client
 - Spécifiques à chaque classe de trafic

Notes :

La spécification des trafics

- Les caractéristiques de trafic (TSPEC) [11, 12]
 - Token bucket r, b
 - Débit crête p
 - Taille maximale de paquet M
 - Taille minimale de paquet m (tout paquet est traité comme si sa taille était au moins m)
- Les unités sont l'octet et la seconde
- Valeurs extrêmes très larges
- Outil commun à toutes les classes de trafic

Notes :

La spécification de la classe de trafic *Guaranteed*

- Les exigences de QoS sont exprimées au travers de deux paramètres
 - Un débit maximal $R \geq r$
 - Une différence de délai S entre le délai acceptable et celui fourni à un débit R
- Chaque routeur doit alors
 - Vérifier la disponibilité de ses ressources
 - Évaluer le retard (maximal) qu'il va induire
 - Une partie constante évaluant un pire cas (C)
 - Une partie fonction du débit (D)
 - Il s'agit en fait d'une marge d'erreur par rapport à un modèle fluide
- Le destinataire reçoit la somme des paramètres C et D et peut évaluer le délai maximal de bout en bout

Notes :

Une architecture réaliste ?

- ① L'architecture *IntServ*
 - Présentation générale
 - Les éléments de base
 - L'architecture
 - Les classes de service
 - Les paramètres de QoS
 - Une architecture réaliste ?

Notes :

Les difficultés de *IntServ*

- Traitement flot par flot
 - États transitoires (*soft state*)
 - Signalisation
 - Files d'attentes individuelles
- Traitement reposant sur les routeurs
 - Cœur du réseau
 - Difficultés de déploiement

Problème majeur

- Quelle capacité de passage à l'échelle ?

Notes :

L'architecture *DiffServ*

- 1 L'architecture *IntServ*
- 2 L'architecture *DiffServ*
 - Les principes
 - Les éléments de base
 - Les classes de trafic
 - Les domaines
 - L'allocation des ressources
 - Les fonctions
 - Bilan de *DiffServ*
- 3 Références bibliographiques

Notes :

Les principes

- 2 L'architecture *DiffServ*
 - Les principes
 - Les éléments de base
 - Les classes de trafic
 - Les domaines
 - L'allocation des ressources
 - Les fonctions
 - Bilan de *DiffServ*

Notes :

L'approche DiffServ

Groupe de travail de l'IETF [6] [9] [1]

Objectifs

- Architecture à QoS supportant les facteurs d'échelle
- Gestion de l'hétérogénéité
- Définir des blocs de base garantissant une architecture évolutive
- Contrainte : garder une architecture simple

Idées de base

- Différenciation des services
 - Qualité de service assurée sur la base de *classes de service*, pas de flux individuels
- Hiérarchisation des fonctions des routeurs
 - Pas de traitement lourd dans le cœur du réseau

Notes :

Les principes de DiffServ

Mise en œuvre des idées de base

- Les routeurs de frontière réalisent la classification (éventuellement complexe) et le conditionnement du trafic
- Les routeurs de cœur traitent un nombre limité de classes de trafic

Gestion de l'hétérogénéité

- Notion de domaine
 - Chaque domaine (réseau) peut appliquer les idées de base "indépendamment" de ses voisins
- Possibilité de reclassification
 - La classification peut être refaite à chaque entrée dans un nouveau domaine

Notes :

DiffServ et les facteurs d'échelle

Modération de l'impact des facteurs d'échelle.

Notion de classe de service

- Ensemble limité de besoins

Per Hop Behavior

- Comportement global par classe de service

Notes :

Les éléments de base

2 L'architecture DiffServ

- Les principes
- **Les éléments de base**
- Les classes de trafic
- Les domaines
- L'allocation des ressources
- Les fonctions
- Bilan de DiffServ

Notes :

Les classes de trafic

Constitution par marquage

- A l'entrée du domaine (réseau)
- Dans le champ DSCP (ex-TOS d'IP)

Traitement par classe

- Traitement commun au sein d'une classe
- Traitement simple et rapide
- Un DSCP implique un PHB

Notes :

Les per hop behaviors

Définition de quelques PHBS

- Expedited Forwarding
 - Ou premium
- Assured Forwarding
 - Plusieurs sous-classes
- Best Effort
 - Toujours présent (default PHB)

Notes :

Les classes de trafic

2 L'architecture DiffServ

- Les principes
- Les éléments de base
- **Les classes de trafic**
- Les domaines
- L'allocation des ressources
- Les fonctions
- Bilan de DiffServ

Notes :

La classe EF

La classe Expedited Forwarding [8] [4]

- Service garanti indépendamment du trafic non EF
 - Débit garanti
 - Délai, gigue, taux de perte faibles
- File d'attente spécifique
- Applications temporellement contraintes
- Intuitivement, le trafic EF doit toujours obtenir au moins le débit R prédéfini
- Concrètement, un paquet ne doit pas subir de retard supplémentaire à celui obtenu au débit R supérieur à une constante

Notes :

La classe AF

La classe Assured Forwarding

- Garantie d'un taux de perte maximal
- Plusieurs classes AF_i
 - Débit min et taux de perte max
 - Découpées en trois priorités de perte $AF_{i,j}$
- Décrite dans une RFC [5]

Notes :

Les domaines

2 L'architecture DiffServ

- Les principes
- Les éléments de base
- Les classes de trafic
- **Les domaines**
- L'allocation des ressources
- Les fonctions
- Bilan de DiffServ

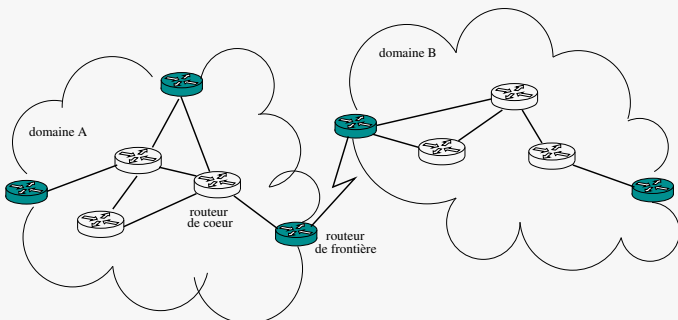
Notes :

Notion de domaine

- Un domaine
 - Un ensemble de routeurs administrés de façon cohérente
- À la frontière d'un domaine
 - Un autre domaine
 - Un client
- Conditions de traitement dans le domaine
 - Négociées dans un SLA [1, 13]
 - Qui intègre en particulier un TCA (*Traffic Conditioning Agreement*) définissant les règles de conditionnement à appliquer à une classe de trafic
 - Mise en place d'un contrat entre le domaine et le client ou un autre domaine

Notes :

Notion de domaine (2)



Notes :

Notion de domaine (3)

La notion de domaine permet

- De différencier les routeurs
 - Cœur
 - Frontière
- De hiérarchiser la gestion
 - ISP, AS, ...
- De mettre en place des PHB différents entre domaines
 - Par re-marquage
- De supporter le passage à l'échelle
- Un déploiement incrémental

Notes :

L'allocation des ressources

2 L'architecture DiffServ

- Les principes
- Les éléments de base
- Les classes de trafic
- Les domaines
- L'allocation des ressources
- Les fonctions
- Bilan de DiffServ

Notes :

L'allocation des ressources

- Comment allouer les ressources à une classe de trafic ?
 - Les contrats de trafic sont négociés entre un domaine et un client/domaine
 - Ils sont spécifiés au travers de SLA intégrant en particulier des TCA
 - L'architecture DiffServ ne spécifie aucun mécanisme d'allocation de ressource
- Différentes options
 - De proche en proche, par exemple à l'aide de RSVP
 - Par une entité supérieure, un *Bandwidth Broker*
 - Par le plan de gestion

Notes :

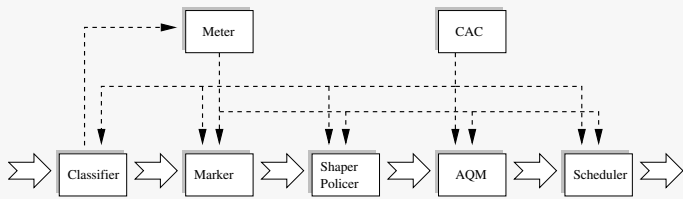
Les fonctions

2 L'architecture DiffServ

- Les principes
- Les éléments de base
- Les classes de trafic
- Les domaines
- L'allocation des ressources
- **Les fonctions**
- Bilan de DiffServ

Notes :

Les fonctions



Présence et localisation dépendantes

- De l'architecture
- De l'entité dans l'architecture

Notes :

Bilan de DiffServ

2 L'architecture DiffServ

- Les principes
- Les éléments de base
- Les classes de trafic
- Les domaines
- L'allocation des ressources
- Les fonctions
- **Bilan de DiffServ**

Notes :

Bilan de *DiffServ*

Les plus

- Plus "réaliste" qu'*IntServ*
- Granularité plus large
- Prise en compte de l'hétérogénéité

Les moins

- Quel plan de contrôle ?
 - Signalisation inter-domaine
 - Déploiement de politique au sein d'un domaine
 - COPS, RSVP, ...
- Quelle gestion des ressources ?
 - Distribuée (quelle pertinence ?)
 - Centralisée (*Bandwidth Broker*)
- Quelle garantie par flux ?

Notes :

Références bibliographiques

- [1] S. Blake, D. Black, M. Carlson, E. Davies, and Z. Wang January. RFC 2475 - an architecture for differentiated service. Informational, IETF, December 1998.
- [2] R. Braden, D. Clark, and S. Shenker. Integrated services in the internet architecture : an overview. Technical report, Internet Engineering Task Force, United States, 1994.
- [3] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. RFC 2205 : Resource reservation protocol (rsvp) – version 1 functional specification. Standards track, IETF, September 1997.
- [4] B. Davie, A. Charny, J.C.R. Bennet, K. Benson, J.Y. Le Boudec, W. Courtney, S. Davari, V. Firoiu, and D. Stiliadis. RFC 3246 : An Expedited Forwarding PHB (Per-Hop Behavior). RFC 3246 (Proposed Standard), March 2002.
- [5] J. Heinanen, F. Baker, W. Weiss, and J. Wroclawski.

Notes :

Références bibliographiques

- RFC 2597 : Assured forwarding phb group.
Standards track, IETF, June 1999.
- [6] IETF, <http://www.ietf.org/html.charters/OLD/diffserv-charter.html>. *Differentiated Services (diffserv) charter*.
 - [7] IETF, <http://www.ietf.org/html.charters/OLD/intserv-charter.html>. *Integated Services (intserv) charter*.
 - [8] V. Jacobson, K. Nichols, and K. Poduri. RFC 2598 : An expedited forwarding phb. Standards track, IETF, June 1999.
 - [9] K. Nichols, S. Blake, F. Baker, and D. Black. Definition of the differentiated services field (DS field) in the ipv4 and ipv6 headers. RFC 2474, Internet Engineering Task Force, December 1998.
 - [10] S. Shenker, C. Partridge, and R. Guerin. RFC 2212 : Specification of guaranteed quality of service. Standards track, IETF, 1997.

Notes :

- [11] S. Shenker and J. Wroclawski.
General Characterization Parameters for Integrated Service Network Elements.
RFC 2215 (Proposed Standard), September 1997.
- [12] S. Shenker and J. Wroclawski.
Network Element Service Specification Template.
RFC 2216 (Informational), September 1997.
- [13] A. Westerinen, J. Schnizlein, J. Strassner, M. Scherling, B. Quinn, S. Herzog, A. Huynh, M. Carlson, J. Perry, and S. Waldbusser.
RFC 3198 : Terminology for Policy-Based Management.
Technical Report 3198, Internet Engineering Task Force, November 2001.
- [14] J. Wroclawski.
RFC 2211 :Specification of the Controlled-Load Network Element Service.
Technical Report 2211, IETF, September 1997.

Notes :

Notes :

Notes :
