

Qualité de service

dans

les réseaux IP

Qualité de service

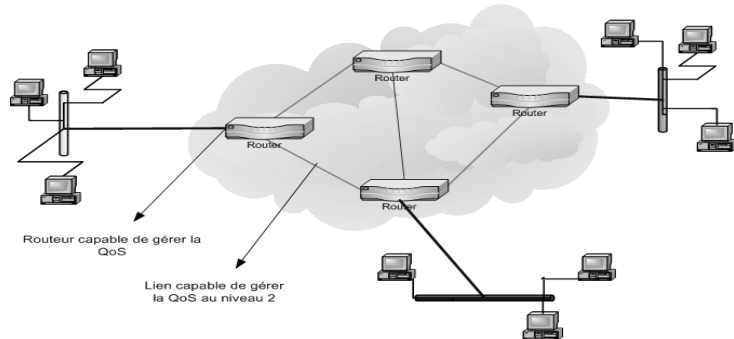
dans les Réseaux IP

Mécanismes de contrôle de trafic: à l'échelle d'un flux

- ◆ Routage avec QoS
 - ◆ Comment choisir un chemin avec des contraintes de QoS
- ◆ Signalisation de la QoS
 - ◆ Comment indiquer les exigences de QoS d'un flux
- ◆ Solutions standardisées
 - ◆ Services intégrés (IntServ)
 - ◆ Services différenciés (DiffServ)

La QoS à travers un réseau

- ◆ Trois mécanismes nécessaires pour garantir de la QoS bout en bout dans un réseau
 - Mécanismes de contrôle de trafic à l'intérieur des routeurs
 - Mécanismes de sélection de chemin répondant aux contraintes QoS
 - Mécanismes d'établissement du flux sur le chemin choisi



Routage intégrant la QoS

- ◆ Contrainte Classique: Trouver le chemin le plus court (nombre de saut, contraintes arbitraires...)
- ◆ L'établissement d'un flux à travers un réseau avec une certaine QoS exige:
 - Modification des protocoles de routage
 - ◆ Les algorithmes de routage doivent prendre en charge les caractéristiques des liens:
 - Bande passante
 - Délai
 - Variabilité de délai
 - Fiabilité
 - Coût
 - Poids administratif
 - ◆ Les routeurs doivent être capables de sélectionner le chemin qui satisfait ces contraintes

Caractéristiques et contraintes

♦ Au niveau du routeur

- Trois types de contraintes de QoS
 - ♦ Additives: trouver le chemin minimisant une somme (ex. délai, nombre de sauts)
 - ♦ Multiplicatives: trouver le chemin minimisant un produit (ex. taux de perte)
 - ♦ Concave: trouver le chemin ayant des caractéristiques supérieures à une valeur donnée (bande passante, classe)

Gestion des contraintes

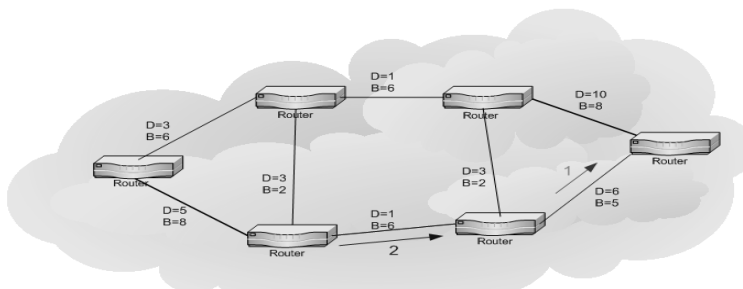
- ♦ Les modifications des protocoles de routage doivent prendre en compte:
 - La distribution de l'information des caractéristiques des liens:
 - ♦ Comment informer l'ensemble des routeurs de l'état de tous les liens du réseau?
 - ♦ Les routeurs doivent être informés à tout moment de l'état des liens
 - A quelle fréquence doivent être envoyées les mises à jours?
 - Le choix du chemin
 - ♦ Prendre en compte les contraintes
 - ♦ Comment trouver le chemin qui satisfait ces contraintes

Distribution des information

- ◆ Choix des protocoles de routage pour la distribution des informations sur l'état des liens distants
- ◆ Protocoles à vecteur de distance (RIP)
 - Diffusion de la table de routage aux voisins
 - ◆ Impossible d'intégrer des informations sur des liens distants
 - Unique contrainte pour le choix du meilleur chemin (nombre de sauts)
 - ◆ Difficile de prendre en compte des caractéristiques de QoS
- ◆ Protocoles de routage à état de liaison (OSPF, IS-IS, IGRP)
 - Connaissance de la topologie du réseau et l'états des liens
 - ◆ Facilité du choix du meilleur chemin avec QoS
 - Distribution des LSP avec les même contraintes que pour la QoS
 - ◆ Les informations sur la charge des liens distant facilement intégrable
 - Fréquence de mises à jours bien définie
 - ◆ Mises à jours provoquées par des modification

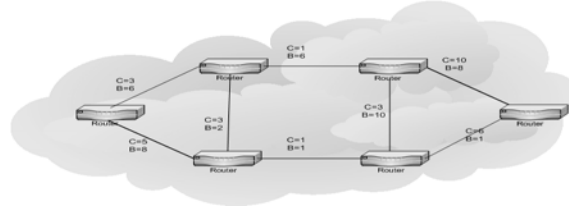
Fréquences des distributions

- ◆ La distribution de l'état du réseau:
 - ne doit pas être diffusée à tout moment
 - ◆ Pour éviter les risques de surcharge du réseau
 - doit être assez fréquente
 - ◆ pour que les routeurs soit informés des changements d'états



Calcul du meilleur chemin avec contraintes

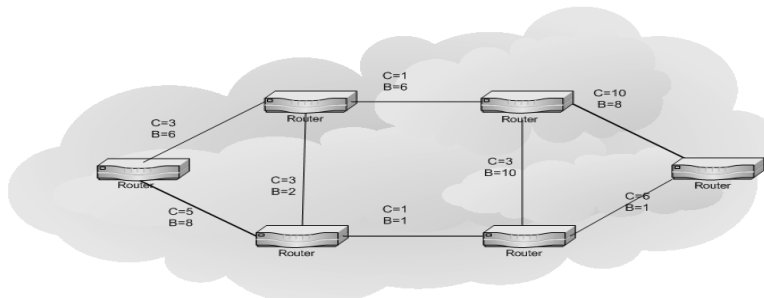
- ◆ Problème: Quelle type de contraintes
- ◆ Contrainte unique additive ou multiplicative
 - L'utilisation d'un algorithme classique suffit pour calculer le meilleur chemin (Dijkstra, Ford...)



- ◆ Pour plus d'une contrainte additive/multiplicative
 - Problème très complexe
 - ◆ Besoin de connaître et d'évaluer tous les chemins
 - ◆ Très chère en matière de CPU pour garantir de la QoS
 - Pour deux ou trois contraintes
 - ◆ Possibilité d'implémentation d'heuristiques
 - ◆ résolution du problème en un temps et avec un coût raisonnables

Calcul du meilleur chemin avec contraintes

- ◆ Pour des contraintes concaves
 - faciles à supporter au niveau d'un routeur
 - ◆ Consiste à trouver par exemple le chemin qui a au moins 3 Mbps de disponible
 - ◆ On applique un des algorithmes classiques
 - On ne considère que les chemins ayant plus de 3 Mbps de disponible



Solutions d'aujourd'hui

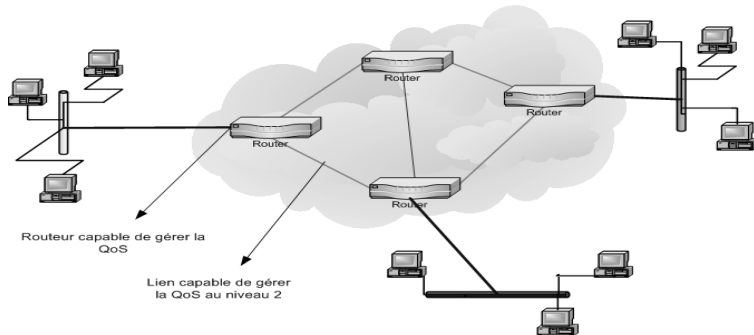
- ◆ Les solutions présentes sur le marché sont peu nombreuses
 - Seuls des contraintes sur la bande passante et le délai sont prises en charge
 - Les contraintes sur la variabilité de délai (gigue) sont moins importantes
- ◆ Obligation de travailler au niveau flux et non pas paquet par paquet
- ◆ Les solutions basées sur le routage par le source sont considérées comme les meilleures
 - La source sélectionne le chemin à travers lequel elle souhaite établir son flux
 - ◆ Les nœuds intermédiaires effectuent un contrôle d'admission
 - ◆ Pas de garantie d'établissement
- ◆ Il existe plusieurs solutions prototype

Mécanismes de contrôle de trafic: à l'échelle d'un flux

- ◆ Routage avec QoS
 - ◆ Comment choisir un chemin avec des contraintes de QoS
- ◆ Signalisation de la QoS
 - ◆ Comment indiquer les exigences de QoS d'un flux
- ◆ Solutions standardisées
 - ◆ Services intégrés (IntServ)
 - ◆ Services différenciés (DiffServ)

La QoS à travers un réseau

- ◆ Trois mécanismes nécessaires pour garantir de la QoS bout en bout dans un réseau
 - Mécanismes de contrôle de trafic à l'intérieur des routeurs
 - Mécanismes de sélection de chemin répondant aux contraintes QoS
 - Mécanismes d'établissement du flux sur le chemin choisi



Techniques d'établissement de flux

- ◆ Pour informer les nœuds du besoin en matière de QoS:
 - Une source a besoin d'un mécanisme de signalisation
 - ◆ Établir, maintenir et libérer un flux ou une connexion
- ◆ Il existe deux façons pour mettre en place un mécanisme d'établissement de flux (ou signalisation)
- ◆ Signalisation dans la bande (In-band)
 - Permet d'établir un flux en utilisant des paquets du même format que ceux des données
 - ◆ Exemple : connexion TCP
- ◆ Signalisation hors bande (out-of-band)
 - Un protocole propre à la signalisation est mis en place
 - Objectif: établir, maintenir et libérer les flux selon le besoin
 - Exemple: code SS7 de la téléphonie classique, canal D pour RNIS

Pourquoi pas le canal D ?

♦ Avantages:

- Un bon mécanisme de réservation pour la commutation de circuit
- Gestion native de QoS (garantie de délai, de débit, de gigue...)
- Tous les paquets du flux de données suivent le même chemin (circuit)

♦ Inconvénients:

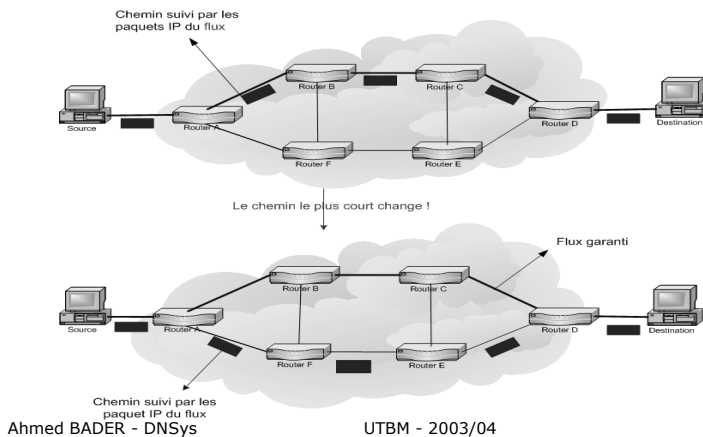
- Mémorisation d'informations concernant toutes les communications sur tous les nœuds intermédiaires
- Pas ou peu de tolérance aux pannes
- N'est pas la meilleure solution pour un réseau à commutation de paquets comme les réseaux IP

RSVP

- ♦ RSVP: Resource Reservation protocol (RFC 2205)
- ♦ Protocole de signalisation
- ♦ Informer le réseau des besoins des applications
- ♦ Conçu pour demander des réservations IP
 - Supporter l'établissement des flux dans les réseaux IP
 - ♦ Différents types de flux
 - Flux au niveau 4
 - Flux MPLS
- ♦ Vu comme le protocole de signalisation de IntServ
- ♦ Fonctionne en IP unicast et en Multicast
 - Supporte les changements de routes
 - Peut gérer les groupes dynamiques

Impact de changement de routes

- ❑ Les paquets sont routés un par un
- ❑ Le flux est établi par une réservation
- ❑ Les routes peuvent changer à tout moment
- ❑ Est-ce la réservation qui doit suivre la route prise par les paquets?
- ❑ Ou bien, les paquets qui doivent rester sur le chemin où sont réservées les ressources?

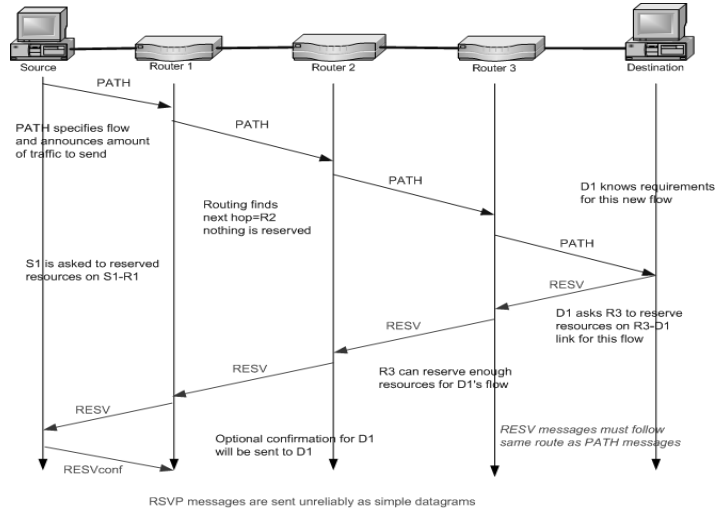


17

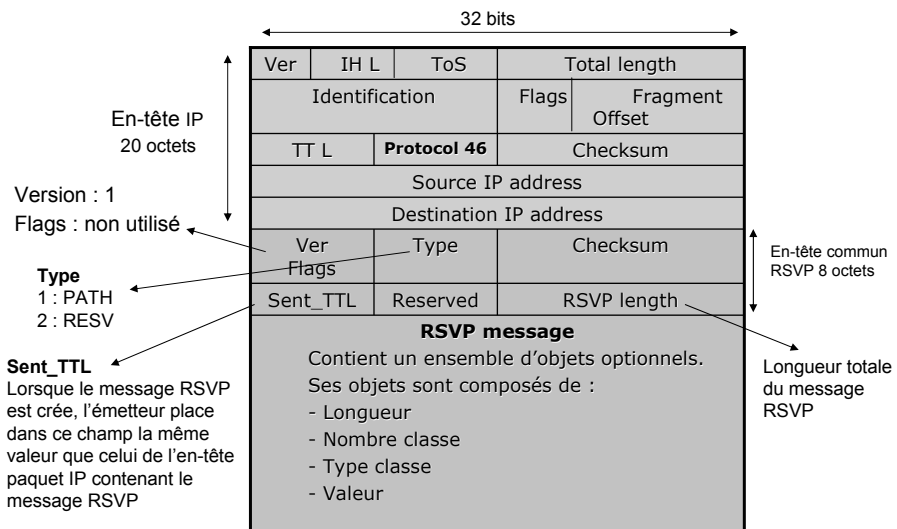
Les changements de routes avec RSVP

- ◆ Principe:
 - Un flux établi avec RSVP dure un certain temps
 - Pour se maintenir, un flux doit se rafraîchir régulièrement
 - RSVP ne prend en charge que des flux unidirectionnels
- ◆ Comportement des systèmes terminaux
 - Notion de contexte applicatif
 - Rafraîchi par des messages RSVP (resv)
- ◆ Comportement des nœuds RSVP
 - Chaque routeur RSVP traversé par un flux RSVP mémorise un état de ce flux
 - Après un certain délai L de non réception des mises à jours, l'état est détruit
 - Libération immédiate de la réservation
 - Messages teardown (démolition)

Etablissement d'un flux avec RSVP



Format du paquet RSVP



Information sur le flux

- ◆ Les routeurs RSVP traversés par un flux RSVP maintiennent un état de ce flux
 - Identification de la source, de la destination et du flux...
- ◆ La maintenance de ces informations peut s'effectuer de deux façons:
 - Etat dur (Hard-State)
 - ◆ Consiste à créer l'état du flux à son établissement et de l'effacer à sa libération
 - ◆ Utilisée dans la commutation de circuits (RTC, RNIS)
 - ◆ Ne supporte pas les changements de routes
 - ◆ Si une panne survient, toutes les réservations et états sont perdus.
 - Etat souple (Soft-State)
 - ◆ Doit être rafraîchi régulièrement
 - Un temporisateur est associé à chaque flux
 - Si le temporisateur expire, le routeur considère que le flux est rompu
 - ◆ Les messages PATH et RESV sert également à réinitialiser le temporisateur
 - ◆ Si une panne survient, les états sont automatiquement réinitialisés
 - ◆ Si la route change
 - De nouvelles réservations et états sont effectués sur un nouveau chemin
 - Les anciennes réservations et états sont rompus

Etablissement de flux RSVP en détail

◆ Objectifs:

- Que contient les messages RSVP ?
 - ◆ Encapsulation dans un paquet IP
- Comment sont gérés les messages PATH et RESV ?
 - ◆ Le chemin inverse suivi par les messages RESV
- Quelles informations d'état maintenues par les routeurs?

Messages PATH

- ◆ Émis « périodiquement » par la source
- ◆ Intercepté et interprété par les routeurs
- ◆ Suit le même chemin que les données
- ◆ Le message PATH Transporte
 - le **Tspec** de la source au moment de son émission
 - les **ADspec** des routeurs traversés
 - ◆ caractérisation des retards dus aux routeurs
 - ◆ mis à jour par tous les routeurs RSVP
- ◆ Le récepteur choisit la bande passante à réserver en fonction du **Tspec** et **ADspec** reçu

Messages RESV

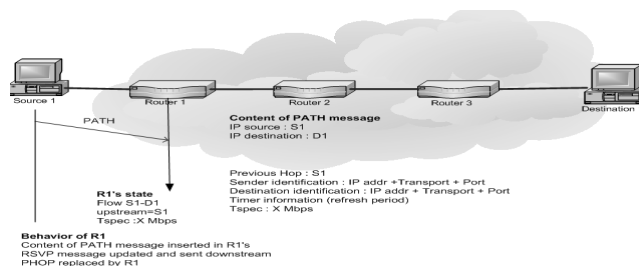
- ◆ Émis par le(s) destinataire(s)
- ◆ Demande une réservation en bande passante
- ◆ Prend le chemin inverse des messages PATH
- ◆ En multicast, les messages RESV sont fusionnés
- ◆ Transporte le descripteur de flot:
 - Tspec de la source
 - FilterSpec: pour la classification
 - Rspec: pour l'ordonnancement
- ◆ Résultats
 - les routeurs connaissent les flots
 - les traitements ne sont plus banalisés

Description de flot

- ◆ FilterSpec: identifier la (les) source(s)
 - IPv4: Adresse source et numéro de port
 - IPv6: Adresse source et flow ID
- ◆ Session: identifier la (les) destination(s)
 - Adresse de destination, protocole ID et numéro de port
- ◆ FlowSpec: décrire les caractéristiques du flot
 - Le trafic émis (TSpec)
 - Le service désiré (RSpec)

Etablissement de flux RSVP en détail

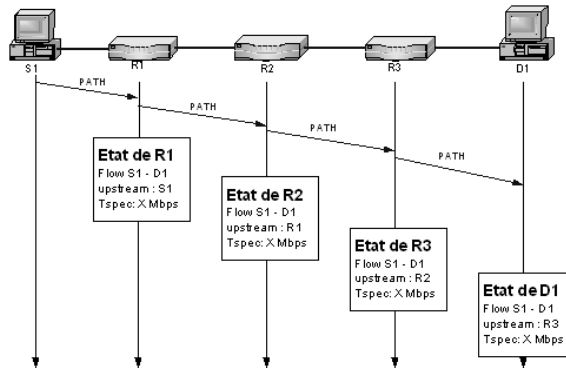
- ◆ Le message PATH est encapsulé dans un paquet IP
 - Adresse source = S1 , adresse destination =D1
 - ◆ Pour que les paquets de données suivent le même chemin que les messages de réservation (PATH et RESV)



- ◆ Le routeur R1 intercepte puis interprète le message RSVP
 - Grâce au champ protocole (46) du paquet IP
 - Extraction du contenu du message RSVP
 - Régénération du message RSVP et envoi vers le saut suivant
 - ◆ Mise à jours du **saut précédent (S1→R1)**

Etablissement de flux RSVP en détail

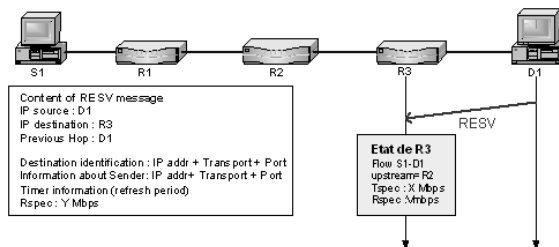
- Chacun des routeurs garde un état d'information sur chacun des flux



- Les informations d'état sert également à router le message RESV
- Aucune réservation n'est effectuée au niveau des messages PATH

Etablissement de flux RSVP en détail

- Le message **RESV** est transmis de proche en proche
- RESV** suit le même chemin que le message **PATH** correspondant:
 - En se basant sur l'information d'état (*previous hop*)

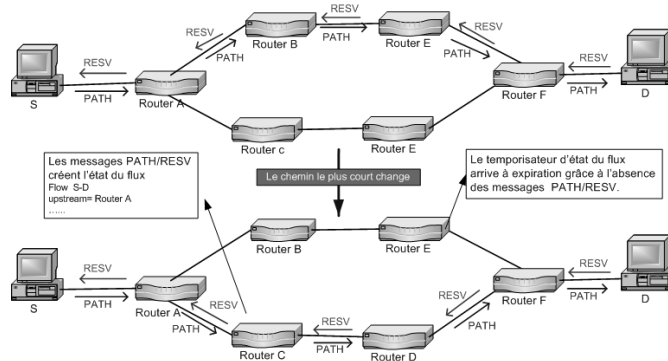


- La destination est responsable de la demande de réservation
- Peut demander plus ou moins que ce que demande la source
- La réservation effective s'effectue par les routeurs impliqués supportant le protocole RSVP

Changement de route

- Lorsque le chemin le plus court change au cours d'une session avec RSVP

- S retransmet son message PATH toutes les «Refresh» secondes (Par défaut : 30 sec)
- D retransmet son message RESV toutes les «Refresh» secondes (Par défaut : 30 sec)



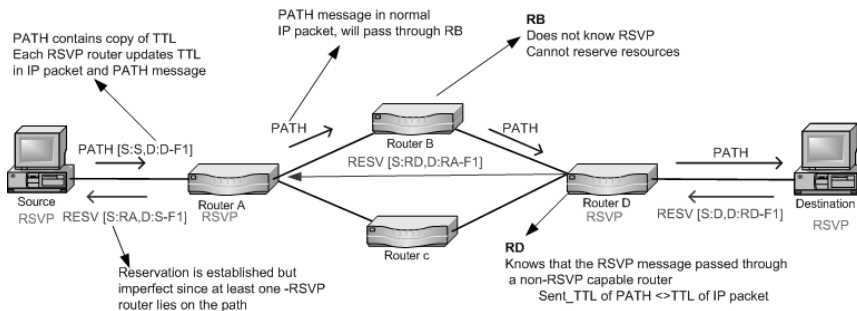
Ahmed BADER - DNSys

UTBM - 2003/04

29

Réseau hétérogène

- ◆ RSVP peut être utilisé dans un réseau hétérogène
- ◆ Offre des facilités pour un déploiement progressif



Ahmed BADER - DNSys

UTBM - 2003/04

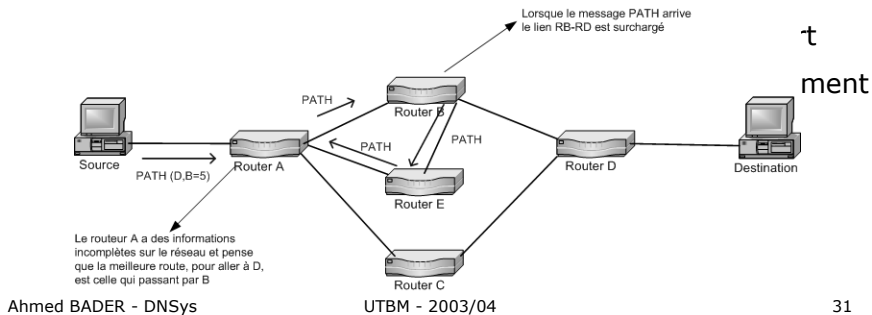
30

Etablissement d'un flux

- ◆ Il existe deux méthodes pour le choix du chemin emprunté par un flux RSVP

- De proche en proche

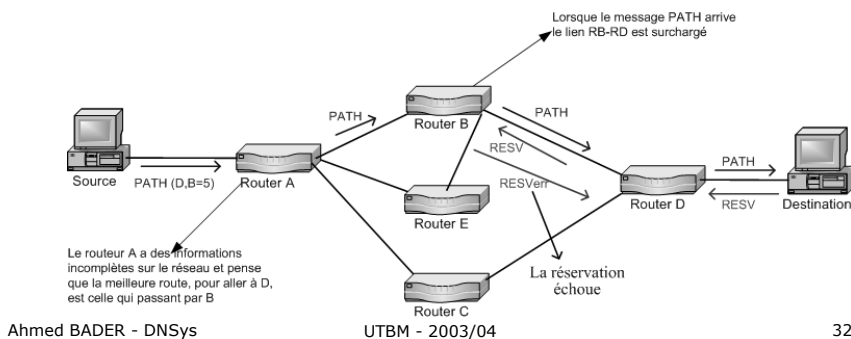
- ◆ Chacun des routeurs intermédiaires décide du saut suivant



Etablissement d'un flux

- Routage par la source

- ◆ La source calcule un itinéraire explicite qui peut être fixe ou variable
- ◆ Elle se base pour cela sur sa connaissance actuelle sur l'état du réseau
- ◆ Méthode très utile pour des flux MPLS



Libération de flux

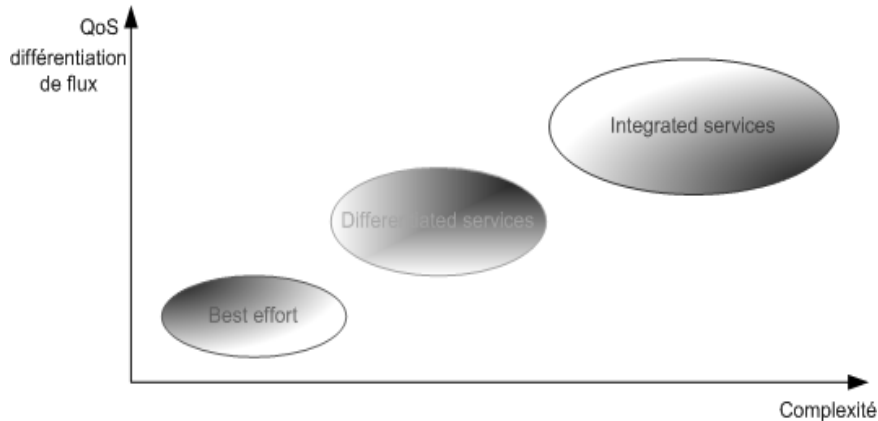
- ◆ Deux méthodes pour libérer les réservations de flux:
 - **Expiration**
 - ◆ L'émetteur et le récepteur arrêtent la transmission des messages PATH et RESV
 - ◆ Ce n'est pas la solution optimale
 - Les routeurs doivent maintenir un flux pendant un temps supérieur à cinq fois la période d'émission des messages PATH et RESV (30s)
 - **Messages de libération**
 - ◆ Définir deux nouveaux messages de signalisation
 - **PATHTear**
 - ◆ Utilisé par l'émetteur pour informer le réseau de la fin du flux
 - **RESVTear**
 - ◆ Utilisé par le récepteur pour informer le réseau de la fin de la réservation
 - ◆ Un temporisateur d'état est nécessaire
 - Tous les messages de RSVP (y compris PATHTear et RESVTear) sont transmis en tant que datagrammes simples sujet à la perte.

Mécanismes de contrôle de trafic: à l'échelle d'un Réseau

- ◆ **Routage avec QoS**
 - ◆ Comment choisir un chemin avec des contraintes de QoS
- ◆ **Signalisation de la QoS**
 - ◆ Comment indiquer les exigences de QoS d'un flux
- ◆ **Solutions standardisées**
 - ◆ Services intégrés (IntServ)
 - ◆ Services différenciés (DiffServ)

Comparaison de services

- ◆ Utilisant tous les mécanismes de contrôle de trafic, quelle type de service pouvons-nous offrir?

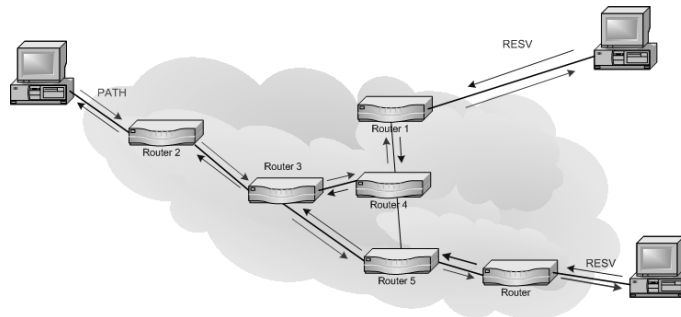


Services intégrés (IntServ)

- ◆ Principe de base
 - Quelques applications spécifiques exigent de la QoS
 - ◆ garanties de délai
 - ◆ Garanties de bande passante
 - La QoS sera fournie aux flux de la couche 4
 - ◆ chaque flux de la couche 4 informera le réseau au sujet de ses exigences en matière de QoS
 - ◆ Le réseau acceptera ou rejettera le flux en se basant sur les exigences et l'état actuel du réseau
 - QoS devrait supporter l'unicast et le multicast
 - Les flux avec QoS devraient coexister avec des flux Best-effort dans le même réseau
 - Aucune modification des protocoles de routage

Réseau IntServ

- ◆ Les routeurs se basent sur l'ordonnement des flux niveau 4 pour fournir la QoS exigée
- ◆ Les routeurs effectuent un contrôle d'admission pour réserver les ressources pour chaque flux
- ◆ Le routage classique est employé pour choisir le chemin vers la destination
- ◆ Les applications se fondent sur RSVP pour signaler leurs besoins en QoS



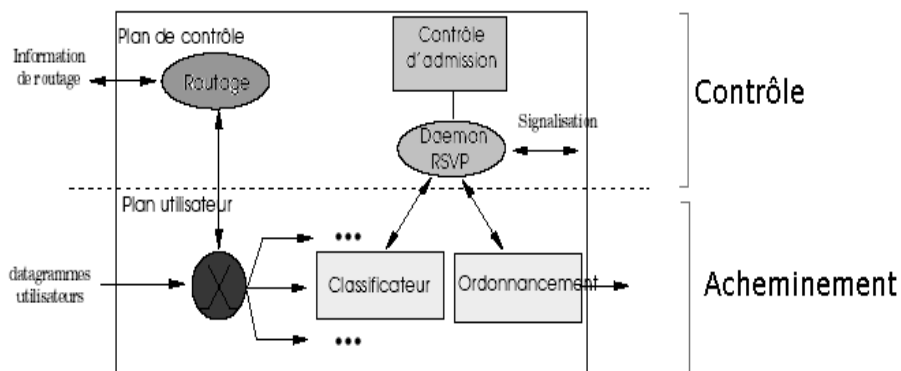
Ahmed BADER - DNSys

UTBM - 2003/04

37

Routeur IntServ

- ◆ L'architecture interne d'un routeur IntServ



Ahmed BADER - DNSys

UTBM - 2003/04

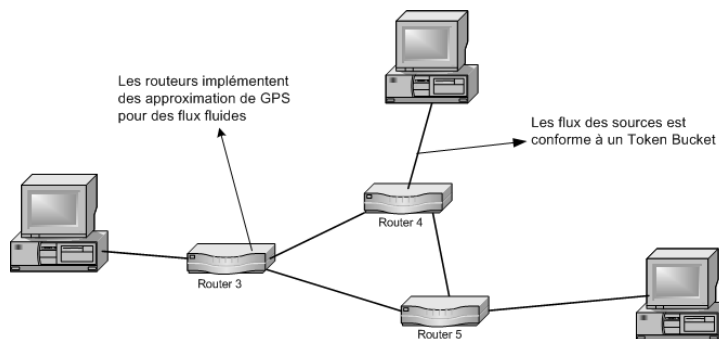
38

Type de Garantie

- ◆ IntServ propose trois types de garanties
 - Service Garanti (Guaranteed Service)
 - ◆ Une garantie ferme basée sur les réservations RSVP
 - Charge contrôlée (Controlled Load)
 - ◆ Une garantie comparable à celle d'un flux Best-effort dans un réseau peu chargé et non congestionné
 - Service Nul (Null Service)
 - ◆ Les routeurs prennent en charge la réservation pour le besoin des applications

Service Garanti

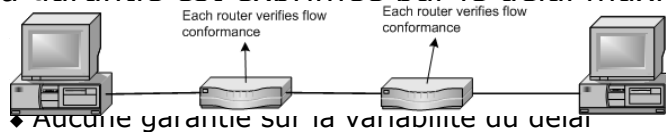
- ◆ Idée
 - Fournir un service qui s'apparente à un lien physique entre la source et la destination
 - ◆ Le service idéal serait celui rendu à un flux fluide
 - ◆ On assure une garantie de délai bout en bout et avec 0% de perte



Service Garanti

◆ Détails

- La garantie est favorable aux paquets conformes
- conformité définie par le mécanisme du Token Bucket
- la garantie est exprimée par le délai maximum



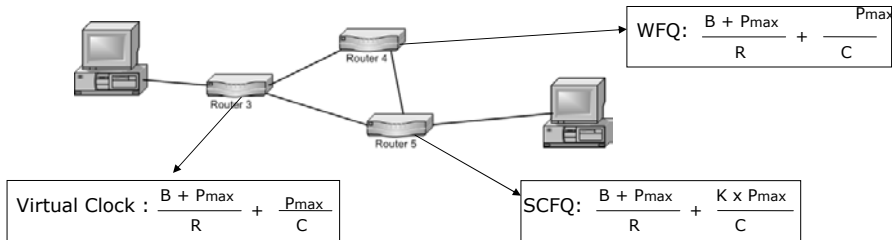
- Garantie de perte

Service Garanti

- ◆ Le contenu du Tspec, spécifié par l'expéditeur dans les messages PATH
 - M = taille maximale des paquets (octet)
 - ◆ Les paquets de tailles plus grandes que M sont considérés comme Best-effort,
 - ◆ La fragmentation de paquet n'est pas supporté par le service garanti
 - Token bucket
 - ◆ Token Bucket du débit pique:
 - Profondeur: M
 - p = débit maximal (octet/sec)
 - durant T secondes, le flux n'enverra pas plus que $M+pT$
 - ◆ Token Bucket du débit moyen
 - b = profondeur du seau (octet)
 - r = débit maximal (octet/sec)
 - durant T seconde, le flux n'enverra pas plus que $b+rT$ octets
 - ◆ m = taille minimale des paquets
 - ◆ Les paquets de taille plus petites que m sont considérés comme de taille m par le Token Bucket

GS: borne du délai

- ◆ Quelle est la limite du délai fourni par GS ?
 - flux de fluide avec GPS
 - ◆ Temps de transmission et de traitement GPS
 - ◆ Temps d'attente dans la queue $\leq b/r$
 - Flux de paquet
 - ◆ Temps de transmission et de traitement
 - ◆ Temps d'attente dans la queue dépend du type de Scheduler
 - ◆ Que peut-on dire des réseaux hétérogènes ?



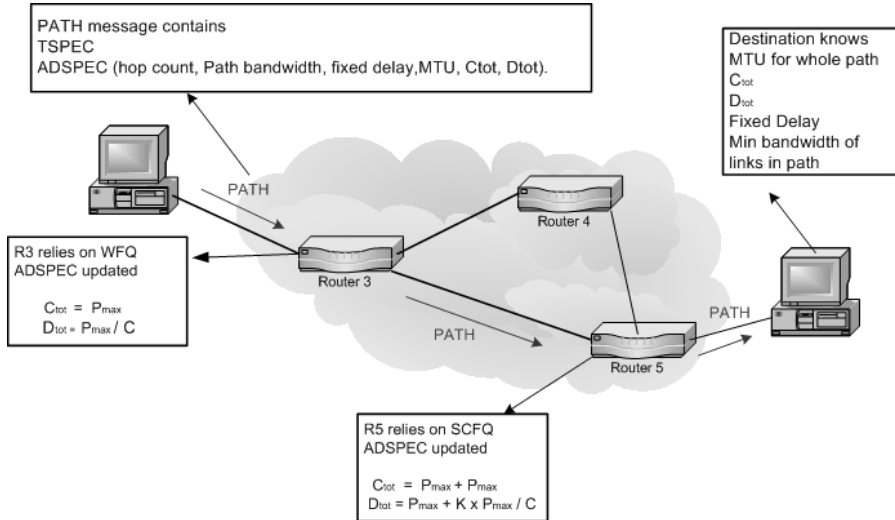
GS: borne du délai

- ◆ Comment peut-on supporter différents scheduler
 - La borne du délai de chaque scheduler est approximativement égale à celle d'un GPS
 - La borne du délai passé dans une queue gérée par un GPS, s'écrit sous la forme:

$$\frac{b}{r} + \frac{C}{r} + D$$

- ◆ C: terme d'erreur dépendant du débit
 - ◆ D: terme d'erreur indépendant du débit
- Chaque scheduler de type GPS devrait connaître ses valeurs C et D
- Les schedulers annonceront ces valeurs dans ADspec des messages PATH
 - ◆ ADspec sera modifié par tous les routeurs intermédiaires
 - ◆ ADspec contiendras alors: $C_{\text{tot}} = \sum C_i$ et $D_{\text{tot}} = \sum D_i$

GS: borne du délai



Ahmed BADER - DNSys

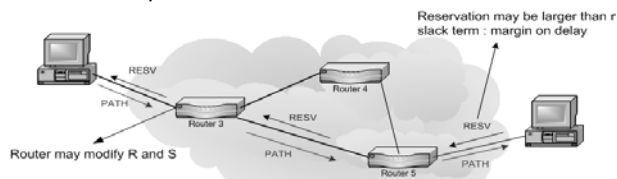
UTBM - 2003/04

45

GS: borne du délai

◆ Message RESV

- La destination spécifie sa réservation dans RSpec
 - ◆ Les paramètres reçus: Token Bucket (r , b , p , M , m)
 - ◆ Les paramètres envoyés:



- **R** : le débit en octets par sec
 - ◆ $R \geq r$
 - ◆ Peut être utilisé pour demander moins de temps d'attente dans les queues
- **S** : Slack term (microsecondes)
 - ◆ Peut être utilisé par la destination lorsque le délai annoncé est meilleur que celui dont on a besoin
 - ◆ Les routeurs intermédiaires peuvent se servir de ce «slack term» pour réduire la réservation du flux en question

Ahmed BADER - DNSys

UTBM - 2003/04

46

Charge Contrôlée

♦ Idée

- Fournir un service avec les mêmes performances que le service Best-effort dans un réseau peu chargé et incongestionné

• Détails

- ♦ Une garantie non déterministe pour les pertes

- mais les pertes de paquet devraient être presque aussi rares qu'avec le GS

- ♦ Aucune garantie de délai bout en bout ni de variabilité de délai

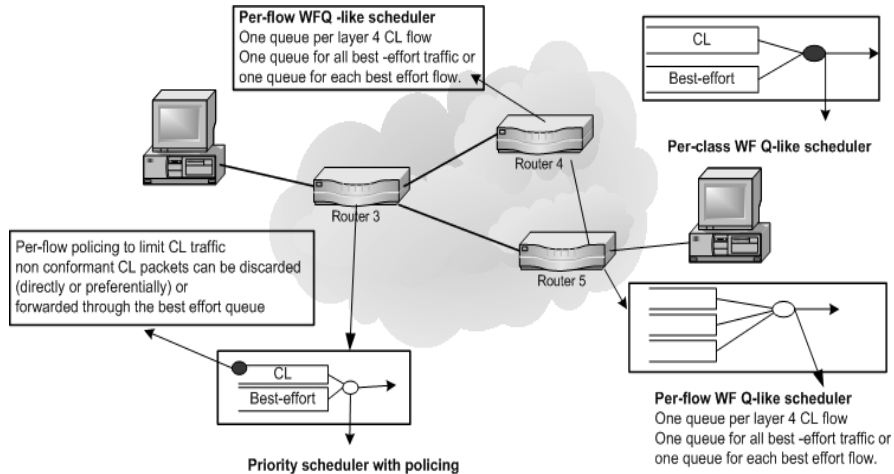
- Mais le temps moyen d'attente dans la queue d'un routeur intermédiaire devrait être bas pour les paquets conformes

• Utilisations

- ♦ Application exigeant une réservation de bande passante

Implémentation

- ♦ Plusieurs implémentations sont possibles



Une critique des services intégrés

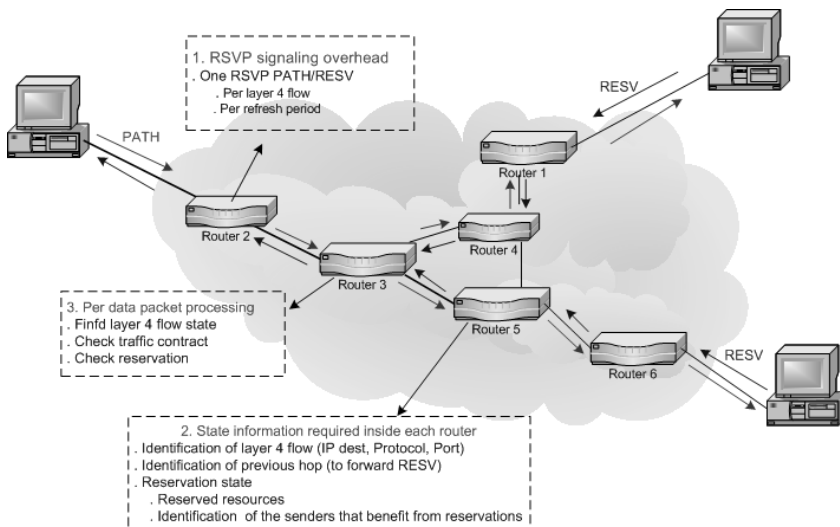
♦ Avantages :

- fournit des garanties de QoS par flux de couche 4
- Gs avec des garanties de délai/bande passante
- CL avec des garanties de bande passante

♦ Inconvénients :

- Exigeante: Chaque routeur intermédiaire doit effectuer quelques opérations pour chaque flux de niveau 4
 - ♦ Traitement de message de RSVP
 - ♦ Classification des flux par flux niv 4
 - ♦ Policing/Queueing/Scheduling par flux niv 4
- Les applications ne peuvent pas toutes exprimer avec précision leur trafic et leurs besoins en matière de QoS

IntServ et les routeurs



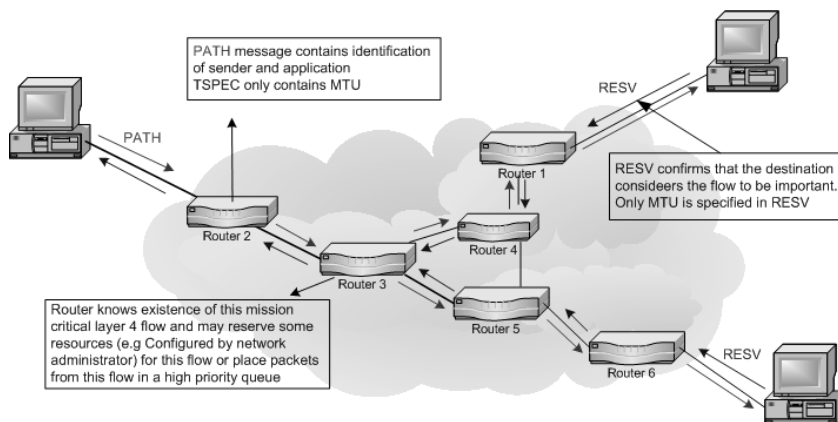
Le service Nul

♦ Analyse

- Dans beaucoup d'entreprises, quelques applications sont beaucoup plus critiques que d'autres
 - ♦ e-commerce
 - ♦ Opérations bancaires temps-réel
 - ♦ commande de processus industriel
- Les managers veulent que le réseau fournisse un service, meilleur que la normale, pour ces applications
 - ♦ Mais ces applications ne peuvent pas définir un contrat de trafic
- Idée du service nul :
 - ♦ Les applications utilisent RSVP pour informer le réseau de l'existence d'un flux critique de couche 4
 - ♦ Le réseau fournit un meilleur service Best-effort pour ces applications critiques.

Fourniture du service nul

- Avec le service nul, un flux de niveau 4 ne recevra pas des engagements forts de QoS, mais le réseau fournira toujours un meilleur service Best-effort.

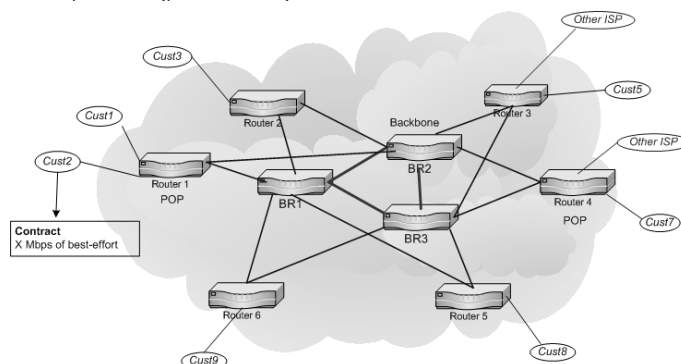


Mécanismes de contrôle de trafic: à l'échelle d'un Réseau

- ◆ Routage avec QoS
 - ◆ Comment choisir un chemin avec des contraintes de QoS
- ◆ Signalisation de la QoS
 - ◆ Comment indiquer les exigences de QoS d'un flux
- ◆ Solutions standardisées
 - ◆ Services intégrés (IntServ)
 - ◆ Services différenciés (DiffServ)

Réseau Best-effort

- ◆ Réseau typique d'un ISP
 - A l'installation, des architectures économiques sont retenues
 - On connaît, pour chaque consommateur le trafic souscrit mais, pas la destination
 - Supervision et statistiques sur la charge de tous les liens
 - ◆ Changement ou augmentation de débit dès que la charge d'un lien devient importante (plus de 60%)



Une approche pragmatique de la QoS

♦ Idée

- Permettre au consommateur de spécifier dans un contrat quelques classes de trafic
 - ♦ Exemple de classe
 - Best-effort, meilleur que Best-effort, temps réel,...
 - ♦ Différents engagements pour chaque classe
 - Délai moyen
 - Pertes moyennes
 - ♦ L'ISP configure ses routeurs pour qu'ils
 - Supportent les besoins de ces classes de trafic
 - N'aient à effectuer que de simples tâches

Contract
1 Mbps of best-effort
7 Mbps of better
0.5 Mbps of realtime

Cust2

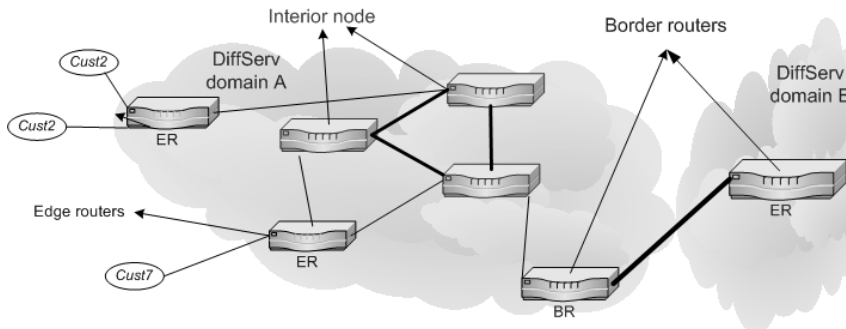
Router 1

BR1

Services différenciés: Architecture

♦ Principe

- Le réseau est divisé en deux parties
 - ♦ Les nœuds intérieurs sont simples pour opérer à haute vitesse
 - ♦ Les fonctionnalités complexes sont fournies par les nœuds frontières (Border et Edge)
 - ♦ Aucun changement sur les protocoles de routage

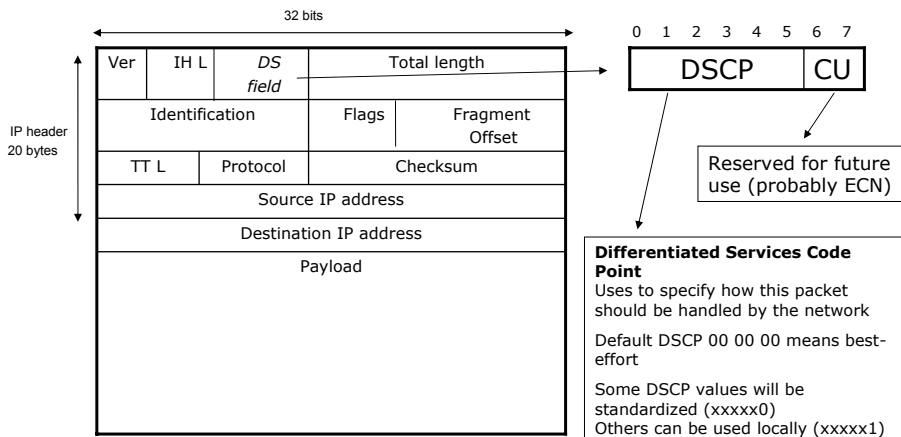


Services différenciés: Architecture

- ◆ Fourniture de différents types de services
 - Le type de service est indiqué explicitement à l'intérieur de chaque paquet
 - ◆ Le marquage peut être effectué par:
 - Consommateur
 - Routeur frontière
 - Routeur du bord
 - Les routeurs intérieurs relient uniquement
 - ◆ Se basant sur cette indication pour fournir les différents types de services
 - ◆ La complexité des routeurs intérieurs dépend
 - Du nombre de services différents
 - Et non du nombre des flux de niveau x

DiffServ: Marquage de paquets

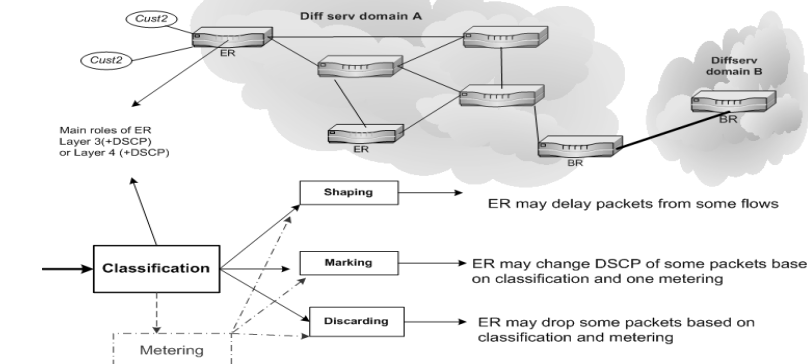
- ◆ Marquage de paquets
 - On redéfinit la sémantique de l'octet TOS rarement utilisé



Dispositifs du réseau DiffServ

◆ Contrat SLA:

- Indique pour chaque service supporté par le domaine de DiffServ la quantité du trafic que le

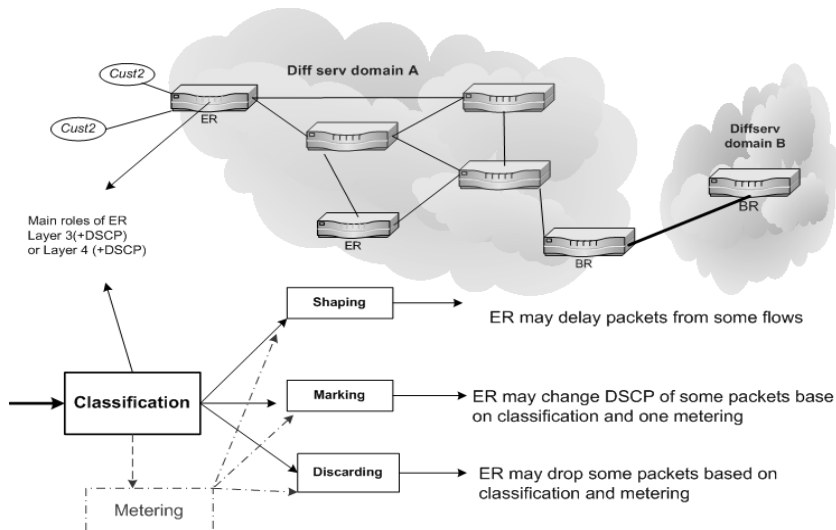


Ahmed BADER - DNSys

UTBM - 2003/04

59

Dispositifs du réseau DiffServ

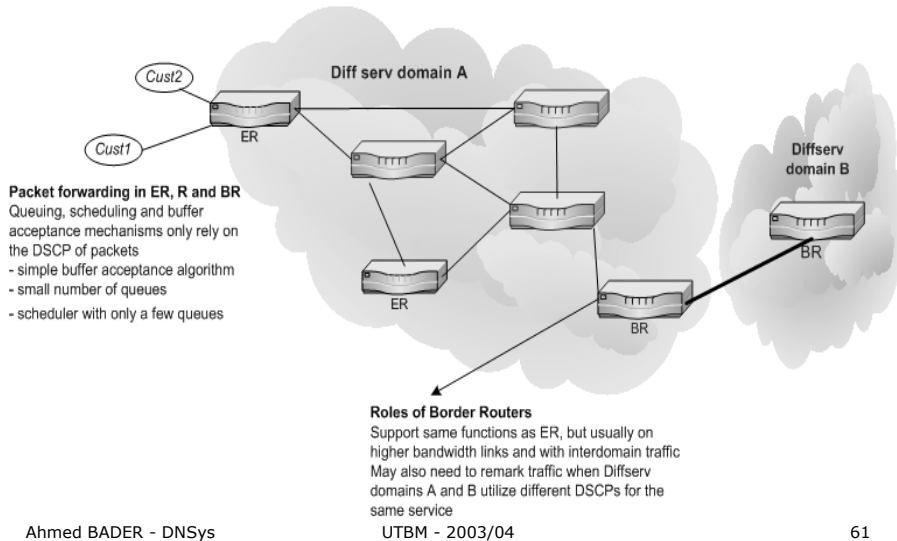


Ahmed BADER - DNSys

UTBM - 2003/04

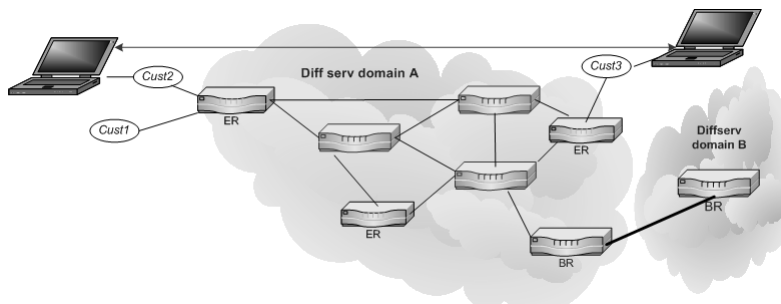
60

Dispositifs du réseau DiffServ



61

Les services différenciés



- ◆ PDH: (Per-Domain-Behaviour) le comportement par domaine ou le service bout en bout fourni par un réseau DiffServ dépend de:
 - Fonctions de conditionnement de trafic (shaping, policing, marking...) des routeurs frontières
 - Fonctions de manipulation des paquets (queueing, scheduling, buffer acceptance) de chaque routeur
 - ◆ PHB: per hop behaviour dans la terminologie DiffServ

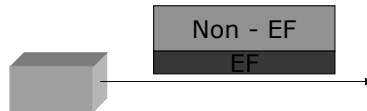
Le comportement par saut (PHB)

◆ Les PHBs définis

- Service Best-effort
 - ◆ Comportement d'un routeur normal
- Class Selector
 - ◆ compatibilité avec le champ 'IP Precedence'
- Expedited Forwarding (EF)
 - ◆ Doit fournir à travers un domaine de diffserv un service avec
 - Un faible taux de perte de paquet
 - Un faible délai bout en bout
 - Une faible variabilité de délai
- Assured Forwarding (AF)
 - ◆ réserver différente quantité de ressources (bande passante, mémoire) pour 4 classes de trafic dans chaque routeur
 - ◆ Des classes de trafic devraient être servies indépendamment

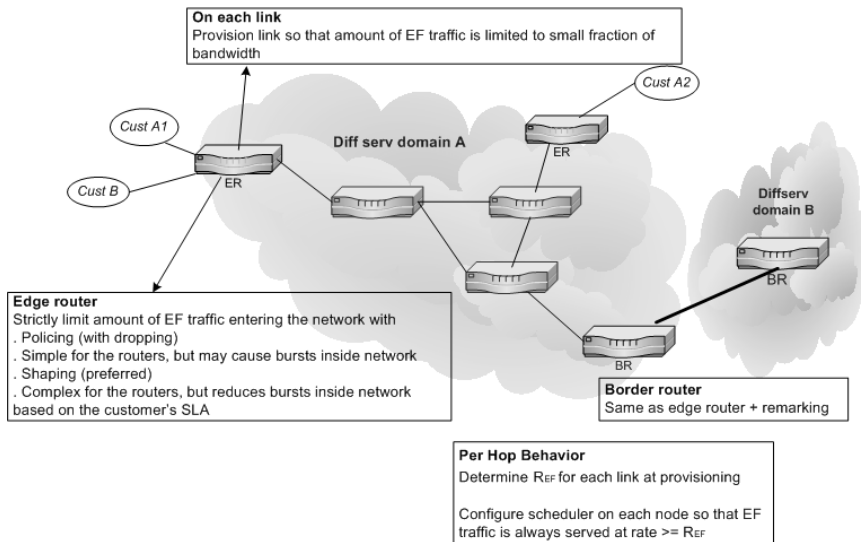
EF : Expedited Forwarding

◆ Principe



- Sur chaque nœud l'administrateur doit être capable de configurer un débit pour le trafic EF
- Le trafic EF doit être servi à un débit $\geq R_{EF}$ indépendamment de l'intensité de n'importe quel autre trafic à travers le nœud
- Les autres trafics ne doivent pas perturber le trafic EF

Exemple d'implémentation EF



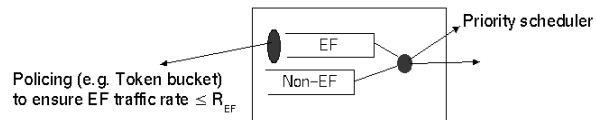
Ahmed BADER - DNSys

UTBM - 2003/04

65

Exemple d'implémentation EF

◆ Implémentation basée sur priorité



◆ Commentaires

- Fournir le plus petit délai possible pour les paquets du trafic EF sur chaque lien qui doit être régulièrement surveillé pour s'assurer que le débit EF ne soit pas supérieur à R_{EF}
 - Si le réseau est bien dimensionné, aucun paquet ne sera rejeté par le « policer » de EF dans les nœuds intérieurs
 - Si des paquets EF sont rejetés dans les nœuds intérieurs, cela pourrait être dû à un problème dans le réseau
 - Le taux des paquets EF rejetés doit être surveillé régulièrement

Ahmed BADER - DNSys

UTBM - 2003/04

66

AF : Assured Forwarding

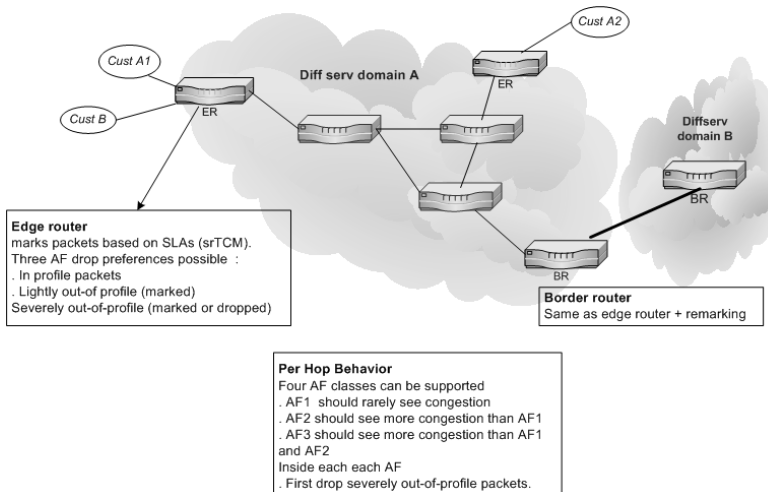
♦ Objectif

- Fournir différemment des services sur un réseau IP simple
- Servir le service 'business' avec un grand surdimensionnement
- Le service dialup payé est correctement dimensionné
- Sous dimensionner le service dialup libre

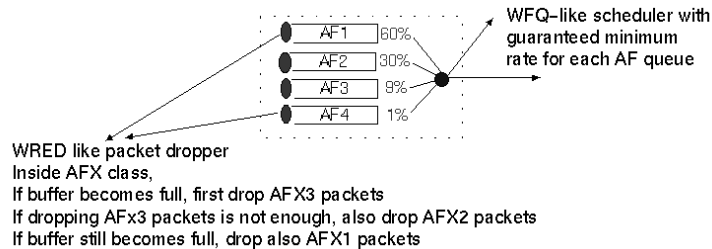
♦ Principe :

- Une couleur est associée à chaque classe de trafic
- les routeurs du bord colorent les paquets selon leurs classes
- Les ressources du réseau sont partagées entre quelques classes
 - ♦ Le service 'business' a beaucoup de ressources pour éviter sa congestion
 - ♦ Le service Dialup a peu de ressources, mais peut tirer bénéfice des ressources inutilisées par le service 'business'
 - ♦ Le service Dialup libre utilise les ressources de surplus

Exemple d'implémentation AF



Exemple d'implémentation AF



◆ Commentaires

- Un réseau n'a pas besoin de supporter les 4 classes AF et les 3 fonctions 'rejet préférentiel' dans chaque classe
- Le poids associé à chaque queue dépend de la quantité prévue de trafic AF dans chaque classe

Manipulation PDB en Bloc Bulk Handling

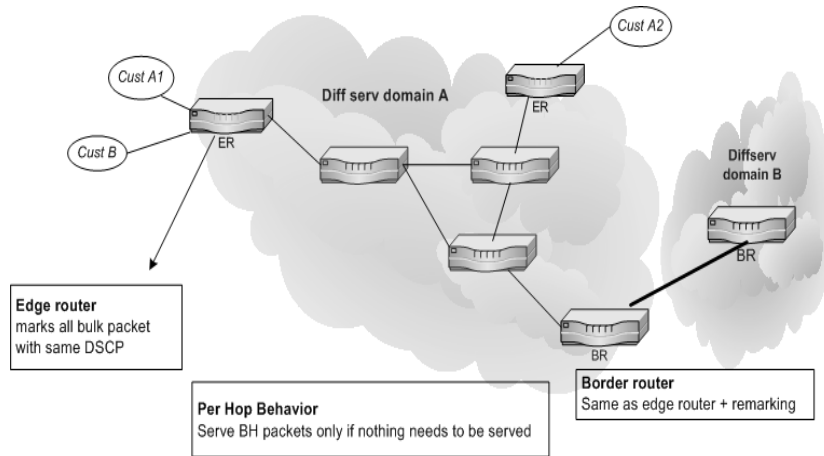
◆ Forcer le trafic en bloc pour être manipulé pendant les heures creuses

- Les applications non importantes doivent consommer la bande passante pendant les heures creuses
 - ◆ Exemple: netnews des ISP ou des compagnies
- Les utilisateurs non importants se décourageront de l'utilisation des ressources réseau pendant les heures creuses

◆ Déploiement

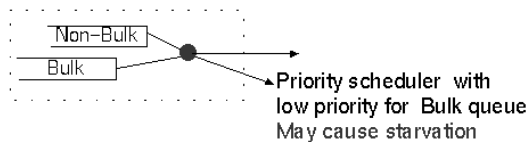
- Basé sur le PHB 'Class Selector'
- Basé sur le PHB 'Assured Forwarding'

Support de Bulk Handling



Support de la manipulation par Bloc

- ◆ Implémentation basée sur la priorité

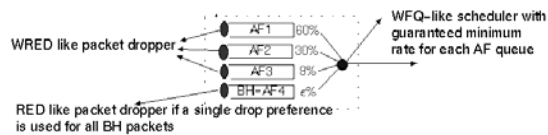


- ◆ Commentaires

- Les paquets BH devront être servis uniquement lorsque aucun autre paquet n'a besoin d'être transmis.
 - ◆ Intéressant du point de vue utilisation d'un réseau
- Peut causer la pénurie qui pourrait mener à des problèmes avec des applications basées sur TCP
 - ◆ Après N échecs de retransmission de paquets pendant la période de pénurie, TCP mettra fin à la connexion
 - ◆ quelques applications peuvent ne pas supporter de telles déconnexions

Support de la manipulation par Bloc

◆ Implémentation basée sur AF



• Commentaires

- ◆ On suppose que la classe AF4 peut être utilisée
 - Mettre le débit de la queue BH au plus petite valeur supporter par le 'scheduler'
 - ◆ Les paquets de BH utilisent toujours une certaine bande passante, mais c'est habituellement meilleur que PQ lors d'une pénurie pour TCP si la queue AF4 est grande
 - Utiliser RED pour le rejet des paquets de la queue AF4
- ◆ Si les 4 classes ou les queues sont déjà utilisées:
 - Utiliser AF43 pour les paquets BH, les placer dans la queue AF4 et les rejetés au fur et à mesure que la file d'attente se remplit