QOS – CLASSIFICATION ET MARQUAGE

Finger In The Net

INTRODUCTION

Allons droit au but !

TYPE DE FLUX ET EXIGENCES

- **Données** (HTTP, FTP)

  Ses exigences sont uniques par application (supportent souvent le délai et la perte). Afin de mettre en œuvre une QOS de qualité, il faut bien analyser les applications.

- **Voix** (VoIP)

  Afin d'avoir une bonne qualité d'appel audio :

  Delay = 150 ms max.
  Jitter = 30 ms max.
  Loss = 1% max.
  Bandwidth = associé au codec utilisé, la signalisation est estimé à 150 b/s

  Attention la voix s'accompagne de la signalisation donc il va falloir gérer les deux flux.

- **Vidéo** (Skype, Facetime, Streaming)

  Afin d'avoir une bonne qualité d'appel vidéo :

  Delay = 150 ms max.
  Jitter = 30-50 ms
  Loss = 0.1 – 1%
  Bandwidth = irrégulière donc prévoir 20% supplémentaire à celle de base

Ok on a les critères… C'est cool !
Mais comment connaître les protocoles qui circulent sur mon réseau ?

ANALYSER SA SITUATION – NBAR – PDLM

Ce n'est pas forcément facile d'analyser une situation déjà en place mais pour ça il existe la fonctionnalité **Network Based Application Recognition NBAR.**

-> Dépendant du CEF (a vérifier en début de sh run)
-> S'applique sur une interface
-> Fourni les statistiques de 128 protocoles (les plus utilisés) du trafic entrant et sortant (paquets, octets, bande passante moyenne sur 5 min.)

Syntaxe :

```
Mise en place de la fonctionnalité

R1(config-if)# ip nbar protocol-discovery

Visualisation des statistiques

R1# show ip nbar protocol-discovery interface interface
```

Tips:

Laisser le temps à votre fonctionnalité de récolter la quantité d'information nécessaire à son exploitation (bénéfique à une meilleure QOS).

Il se peut que votre trafic ne soit pas recensé par ces 128 protocoles (vision dans la case **unknow** lors de la commande de visualisation de la récolte par NBAR). La na
du fichier recensent les protocoles est **Packet Description Language Module (PDLM). Il est possible de le mettre à jour** (fichier à télécharger sur**www.cisco.com)**.

```
R1# show ip nbar version

R1# show ip nbar pdlm
```

Tips:

Seul un fichier PDLM de version supérieur à celui en place pourra le remplacer.

Maintenant que je sais quel vont être mes critères…

CLASSIFIER

class-map

Comme son nom l'indique nous allons classifier nos groupes dans des « **class-map** » que l'on peut comparer à des dossiers.

Les class-map sont les **éléments moteurs** de la QOS, nous allons en créer pour colorer nos flux, gérer la congestion ou encore prévenir celle-ci. Ces **class-map** seront
remplies d'une ou plusieurs conditions (un peu comme une boucle de programme).

Nous pouvons dès le début de la commande indiquer si nous voulons que toutes les conditions soient réunies ou si une seule d'entre elles suffira.

```
Toutes les conditions présentes doivent être respectées:

Router(config)# class-map match-all name

Une des conditions présentes doit être respectée:

Router(config)# class-map match-any  name
```

En l'absence de **match-all** ou **match-any** l'équipement implémente automatiquement un **match-all**.

```
Création de la class-map C_fingerinthenet:

Router(config)# class-map C_fingerinthenet

Mise en place d'un critère via la commande match:

Router(config-cmap)# match
```

Une fois le dossier créé **(config-cmap)#**, voyons les différents critères possibles :

- **Access-group** suit les conditions d'une access list ;
- **Protocol** match directement les paquets transportant le protocole sélectionné ;
- **Input-interface :** tous les paquets provenant de cette interface ;
- **Destination-MAC-address :** tous les paquets ayant cette adresse en destination MAC ;
- **Source MAC adress :** tous les paquets ayant cette adresse en destination MAC ;
- **Any :** tous les paquets ;
- **Class-map :** vous pouvez imbriquer les class-map (attention au manque de lisibilité de la configuration) ;
- **Différentes coloration :** cos, ip dscp, ip précédence… (détaillé dans la section « marquer »).

Nous pouvons classifier les flux très précisément en impliquant plusieurs commandes **match** dans une **class-map** (une ACL et un autre **match** par exemple). Si vous vous souvenez plus trop des possibilités des ACL je vous invite à aller lire l'article **Access-list**.

Exemple de classification des flux http et https en provenance de notre serveur mac adresse F I N T :

Nous sommes obligés de créer une ACL pour le protocol HTTPS (ne fait pas parti des possibilités de la commande **match protocol)** :

```
Création de l'ACL HTTPS:

Router(config)# ip access-list extended HTTPS
Router(config-ext-nacl)# permit tcp host any any eq 443
Autorise les paquets de n'importe quelle source IP
vers n'importe quelle destination IP utilisant le port 443
```

Première version avec deux **class-map** séparées:

```
Création de la class-map C_Serveur_HTTPS (match-all par défaut):

Router(config)# class-map C_Serveur_HTTPS
Router(config-cmap)# match access-group name HTTPS
Router(config-cmap)# match source-MAC-adress F :I :N :T

utilisant le port 443 ET qui ont F:I:N:T comme adresse MAC source

Création de la class-map C_Serveur_HTTP:
Router(config)# class-map C_Serveur_HTTP
Router(config-cmap)# match protocol http
Router(config-cmap)# match source-MAC-adress F :I :N :T
utilisant le protocol HTTP (port 80) ET qui ont F:I:N:T comme adresse MAC source
```

Deuxième version, pas la plus simple, mais montrant la possibilité d'imbrication :

```
Création de la class-map C_Prot_Serv:

Router(config)# class-map match-any C_Prot_Serv
Router(config-cmap)# match access-group name HTTPS
Router(config-cmap)# match protocol http

utilisant le port 443 OU utilisant le protocole HTTP (port 80)

Création de la class-map C_Serveur:

Router(config)# class-map C_Serveur
Router(config-cmap)# match source-MAC-adress F :I :N :T
Router(config-cmap)# match class-map C_Prot_Serv
ont F:I:N:T comme adresse MAC source
ET qui remplissent les conditions de la C_Prot_Serv
```

Quelques règles sur le class-map avant de continuer :

-> Par défaut l'équipement applique un **match-all**
-> **256** class-map par équipement maximum
-> maximum **40** caractères pour le nom de class-map

MARQUER

Notre volonté est de peindre les voitures provenant de notre serveur FINT et utilisant les protocoles HTTP et HTTPS. Nous avons appris à trier les voitures (**classificat** maintenant il faut choisir la bonne couleur.

Il faut créer une stratégie appelée « **policy-map** » dans laquelle on va :

- **affecter** les **class-map** (identifie les bonnes voitures) ;
- **utiliser** la commande **set** afin de choisir la peinture à appliquer.

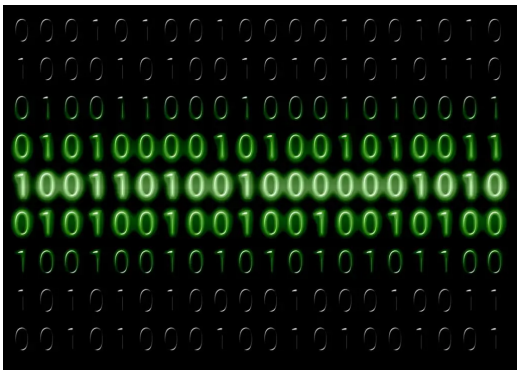
Nous n'allons qu'aborder les **policy-map**, dans le but du marquage. Nous en reparlerons dans la « gestion de la congestion ».

Création de la stratégie fingerinthenet:

```
Router(config)# policy-map fingerthenet
Router(config-pmap)# class C_Serveur
Router(config-pmap-c)#
Implication de la class-map C_Serveur dans la stratégie
```

La stratégie est créée, nous avons sélectionné les voitures (class-map), maintenant il va falloir choisir la couleur pour ces voitures.

SET



Modification des en-têtes de paquet

Qu'est-ce que la couleur ? C'est un changement d'état des bits de champs spécifiques faisant partie de l'en-tête de la couche 2 ou de la couche 3.

Le marquage peut être effectué dans différentes en-têtes de la couche 2 :

- **cos :** dans la trame 802.1Q sur 3 bits ;
- **exp :** dans la trame MPLS sur 3 bits ;
- **de :** dans la trame Frame Relay sur 1 bit ;
- **clp :** marquage dans la trame **ATM** sur 1 bit.

Ces marqueurs seront préservés dans leur propre réseau **uniquement** (à moins de mettre en place une « **class-map** » qui match leur coloration -critères des **class-ma** peu plus haut- !!!!

Version 4 bits	IHL 4 bits	Type Of Service 8 bits	TPL 16 bits	
Fragment ID 16 bits			Flags 3 bits	Fragment offset 13 bits
TTL 8 bits		Protocol 8 bits	Checksum 16 bits	
Source IP 32 bits				
Destination IP 32 bits				
Options				

en-tête IP et position du champ contenant le marquage

Nous allons nous attarder sur le marquage de la couche 3 :

– **Ip précedence :** les 3 bits de poids fort du champ Type Of Service (ToS) ;
– **DSCP (Differentiated Service Code Point) :** les 6 bits de poids fort du champ ToS.

		b7	b6	b5	b4	b3	b2	b1	b0			
		Type Of Service										
		DSCP								DSCP		
Notation décimale	Précedence IP NOM									NOM	Notation décimale	
7	**Network**	1	1	1	0	0	0	x	x	**CS7**	**56**	
6	**Internet**	1	1	0	0	0	0	x	x	**CS6**	**48**	
5	**Critical**	1	0	1	1	1	0	x	x	**EF**	**46**	
4	**Flash-override**	1	0	0	0	0	0	x	x	**CS4**	**32**	
					0	1	0	x	x	**AF41**	**34**	
					1	0	0	x	x	**AF42**	**36**	
					1	1	0	x	x	**AF43**	**38**	
3	**Flash**	0	1	1	0	0	0	x	x	**CS3**	**24**	
					0	1	0	x	x	**AF31**	**26**	
					1	0	0	x	x	**AF32**	**28**	
					1	1	0	x	x	**AF33**	**30**	
2	**Immediate**	0	1	0	0	0	0	x	x	**CS2**	**16**	
					0	1	0	x	x	**AF21**	**18**	
					1	0	0	x	x	**AF22**	**20**	
					1	1	0	x	x	**AF23**	**22**	
1	**Priority**	0	0	1	0	0	0	x	x	**CS1**	**8**	
					0	1	0	x	x	**AF11**	**10**	
					1	0	0	x	x	**AF12**	**12**	
					1	1	0	x	x	**AF13**	**14**	
0	**Routine**	0	0	0	0	0	0	x	x	**BE**	**0**	

Les Class Selector représentent des **catégories.**

L'équipement lie **l'importance** de la transmission du paquet à la valeur de la précedence IP.

Les Assured Forwarding représent des **sous categories.**

Les bits b4 et b3 indiquent une priorité de poubellisation (DROP).

A l'inverse des CS, l'équipement lie la poubelisation (DROP) à la valeur des bits b4 et b3.

Exemple:

AF41 > AF43 > AF21

Bilan de la coloration couche 3

A vous d'organiser votre marquage, **mais j'ai quelques conseils** :

– **CS7 & CS6 :** application sur les échanges des protocoles de routage ;
– **Express Forwarding EF:** application sur les flux de VOIP ;
– **cohérence :** vis à vis de la totalité de votre réseau.

On sait de quoi est composée la couleur, nous nous étions arrêtés dans la stratégie après avoir impliqué la **class-map**.

Exemple de modification du champ dscp: :

```
Router(config)# policy-map fingerthenet
Router(config-pmap)# class C_Serveur
Router (config-pmap-c)# set ip dscp cs4 (ou 32)
```

APPLICATION DE LA STRATÉGIE

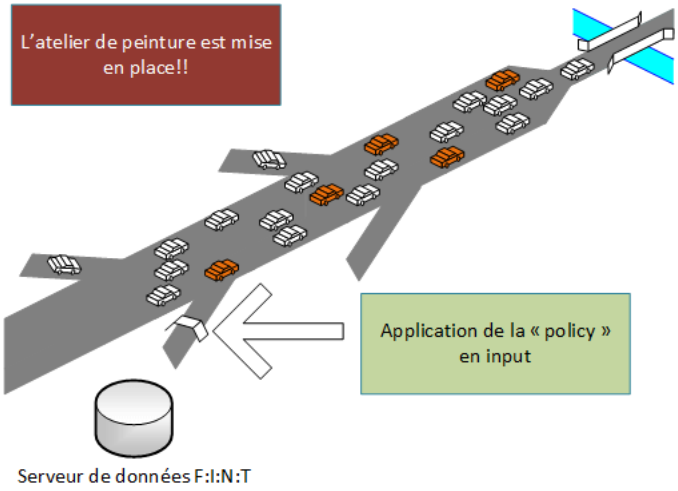
On a choisi les voitures, la couleur et préparer nos ateliers de peintures. Nous devons mettre en place cette coloration **au plus près** des services, soit l'interface directe connectée à mon serveur.

```
Router(config)# interface fastEthernet 0/1
Router(config-if)# service-policy input fingerinthenet
```

On remarquera le **input**, cela signifie que tous les paquets entrant par cette interface seront examinés par la stratégie.
A partir de ce moment, les paquets traversant cette stratégie sont considérés comme des **BEHAVIOR AGREGATE "BA"**.

Afin de le vérifier :

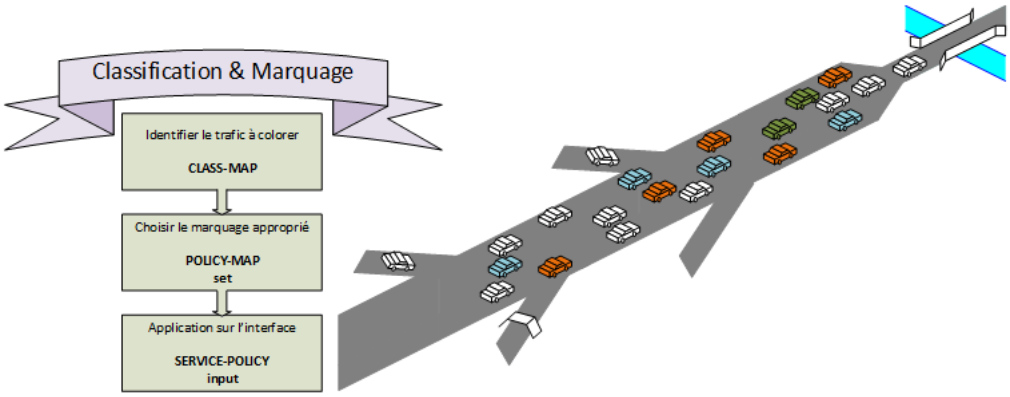
```
Router# sh policy-map
```



Mise en place d'une stratégie en input

Comme on peut le remarquer sur l'image, il va falloir installer des marquages à chaque entrée et au plus près de nos services.

BILAN



L'étape de classification et marquage est terminée mais ça ne change rien à la congestion existante… Ce n'était que la première étape ! Maintenant on peut :

- **Gérer** la congestion ;
- Faire de la **prévention** de congestion.

En espérant que vous avez apprécié cet article !
N'hésitez pas à me la faire savoir !!

FingerIn

CONTENU RÉSERVÉ AUX MEMBRES DU SITE

S'enregistrer Se connecter

OFFRE LIMITÉ

DÉBLOQUE L'INTÉGRALITÉ DU SITE POUR SEULEMENT

19.90€

Abonne-toi

Noël NICOLAS

Expert Réseau
15 ans d'expérience
CCNP Routing and Switching
Fondateur du site FingerInTheNet

J'aime 9 personnes aiment ça. Inscription pour voir ce que vos amis aiment.

LES BASES DU RÉSEAU

Les adresses IPv6
Débuter avec CISCO

LA PARTIE SWITCHING

Les Vlans
Les liens Trunk
Le routage inter-vlan
Le protocole VTP
Les VACLs
Les Private VLANs
Le Spanning-Tree
Introduction au FHRP
Le protocole HSRP
Le protocole VRRP
Le protocole GLBP
La méthode VSS
Etherchannel

LA PARTIE ROUTING

Configuration d'un routeur
Le routage statique
Le routage dynamique
Le protocole RIP
Le protocole OSPF
Le protocole EIGRP
Le protocole BGP
Le Frame Relay
HDLC / PPP
Le protocole PPPoE
Le metroEthernet
Les services CLOUD

LES SERVICES

Le NAT
Le SLA
Le protocole DHCP
Le protocole NTP
Le protocole SNMP
Le protocole NetFlow
La gestion des logs
Spanning-Tree PortFast

LA SÉCURITÉ

Le protocole SSH
Les access-lists
Le DHCP Snooping
Le Dynamic ARP Inspection
L'IP Source guard
Le protocole AAA
Le port-security
Le port-based authentication
Le Storm-control

LES VPNS

Les tunnels GRE
Le protocole IPSec
Le DMVPN

SPANNING-TREE

Introduction au Spanning-Tree
le Protocole STP
le Protocole RSTP
le Protocole MST
le Protocole PVST+
le Protocole Rapid-PVST+
Spanning-Tree Portfast
Spanning-Tree BPDUFilter
Spanning-Tree BPDUguard
Spanning-Tree Root Guard
Spanning-Tree Loop Guard

© FingerInTheNet.com. Tous droits réservés

Conditions général d'utilisation

Site hébergé par **WPServeur**

POLITIQUE DE CONFIDENTIALITÉ