

11/09/2013
20/09/2013

Réseaux

Jean-Marc PIERSON
IRIT
pierson@irit.fr

Michelle SIBILLA

2 QCM + CT

Objectifs

1. Architecture en couches OSI
2. Couche Réseau } IP / ICMP) Ethernet (liaison)
3. Couche Transport } TCP / UDP
4. Sécurité, administration réseau.

Livres:

A. Tanenbaum

"Réseaux"

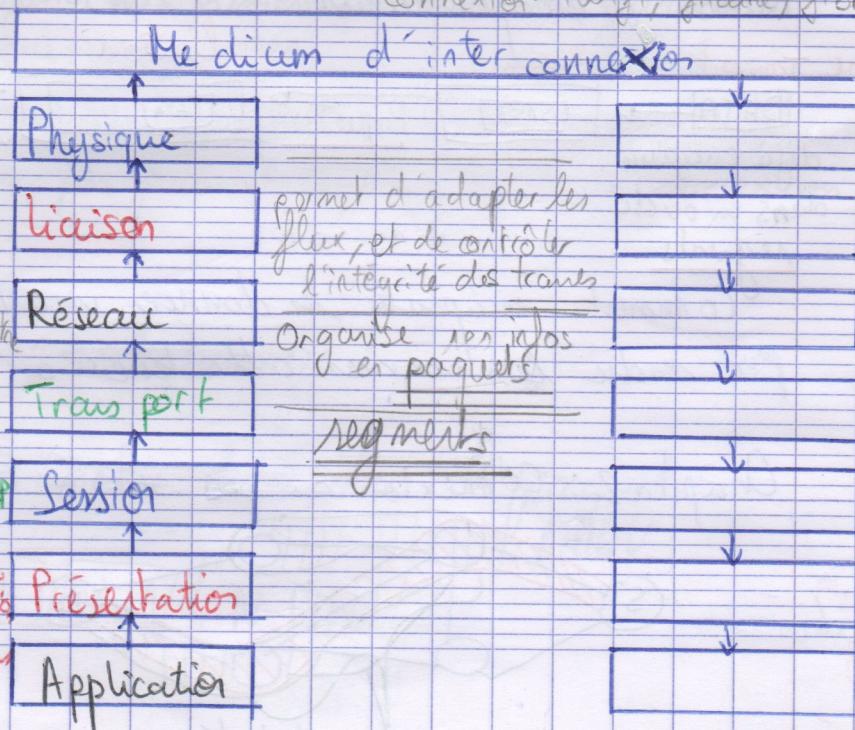
G. Pujolle

"Les Réseaux"

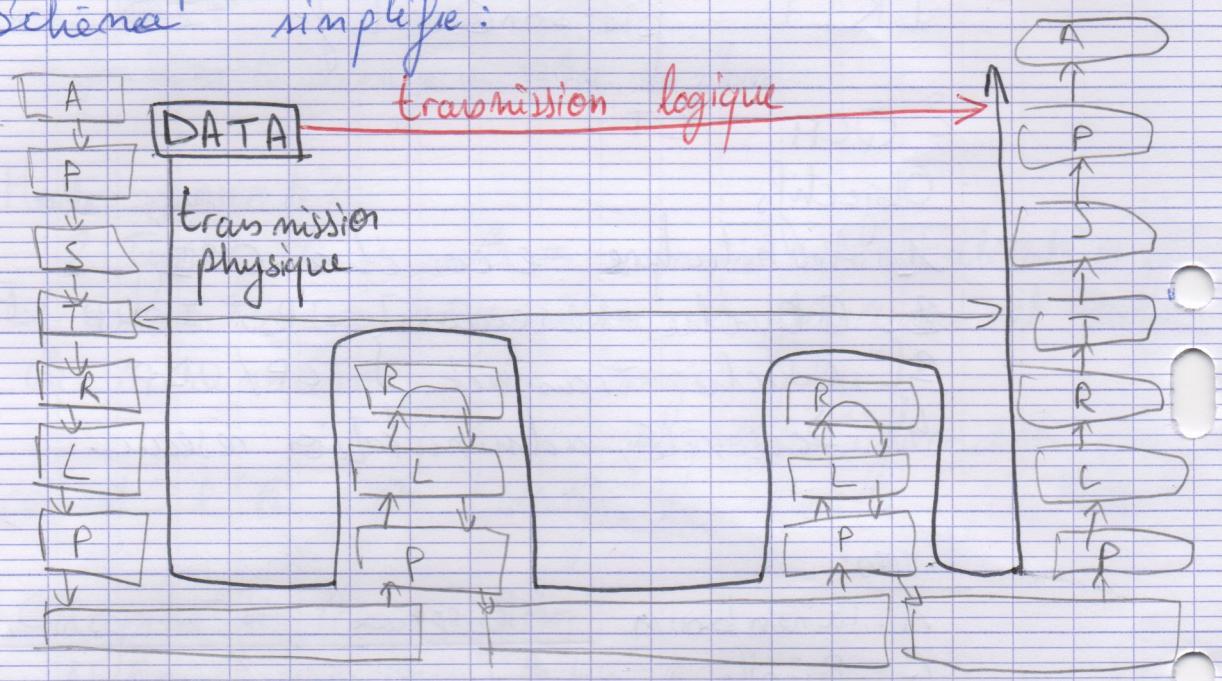
Chapitre 1. Modèle OSI

Modèle en 7 couches

Il peut y avoir plusieurs médium d'interconnection (wifi, filaire, fibre optique)

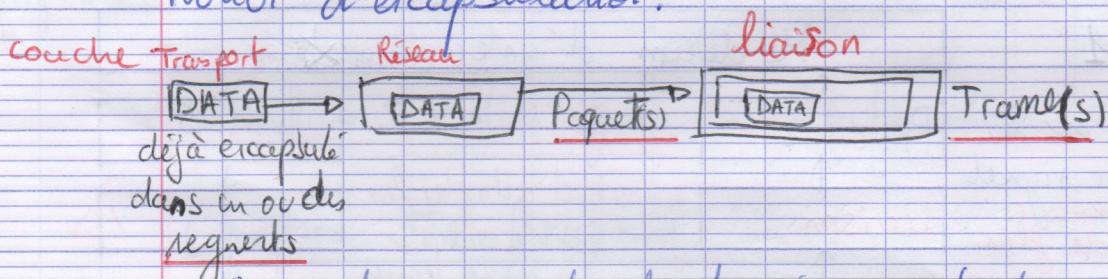


Entre deux machines il y a ce qu'on appelle des nœuds intermédiaires (box, passerelle, routeur) et ne voit pas au dessus de la couche Réseau :
 Schéma simplifié :



Notion de service : une couche offre des services à la couche immédiatement supérieure en s'appuyant sur ceux offerts ^{par cette} immédiatement inférieure.

Notion d'encapsulation :



Comment encapsuler les données venant des couches supérieures pour rendre les services attribués à la couche

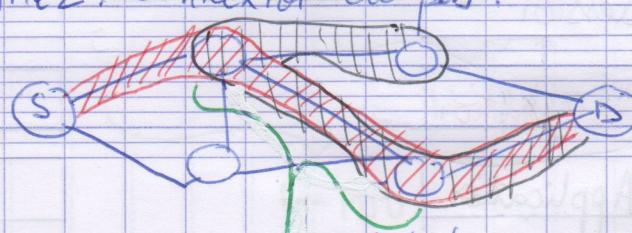
Chapitre 2 : Connexion au pas.

S : source

D : destination



Chemin réservé pour l'utilisation.



possibilité de partager liaison physiques

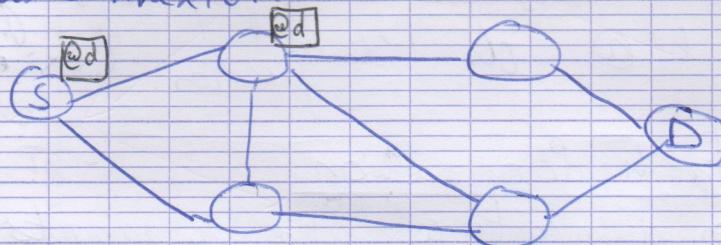
Connexion? S'assurer de la présence du destinataire avant de communiquer

Ex: téléphone (besoin d'une connexion préalable) ou RTC les ressources sont réservées pour la communication. sur les noeuds intermédiaires

Ex: réseau RNTS, ATM.

Création d'un chemin virtuel entre la source (S) et la destination (D). Tous les paquets, une fois la communication établie, passent par ce chemin.

Sans connexion:



Ex: la Poste
IP

Nécessité d'avoir l'@ destination

	Sans Connexion TCP	Avec Connexion ATM
+	- pas de réservation - diffusion vers plusieurs facilitateurs - tolérance aux pannes	- ordre respecté - délai garantie - assez sécurisé
-	- destinataire pas sûr - ordre pas forcément respecté	- temps nécessaire à la connexion, et à la reconnexion

Chapitre 3: Adressage

Consiste à mettre l'@ dans le paquet NADV

(@) DATA / N-PDU

Mode non-connecté :

Chaque paquet à l'adresse du destinataire et de la source.

ex: @IP

⇒ fonctions de routage à mettre en place:

Mode connecté:

2 types de paquets = - paquet d'initialisation de la connexion ⇒ réservation des ressources et identification de la connexion (ex: "rouge", "bleu"):

- paquet d'information qui contient l'@ de l'identification (ex: "rouge", "bleu").

Chapitre 4: Priorisation

Notion d'importance relative entre les paquets

Dans le N-PDU (paquet) on doit pouvoir distinguer plusieurs classes ⇒ gestion de la qualité de service (QoS)

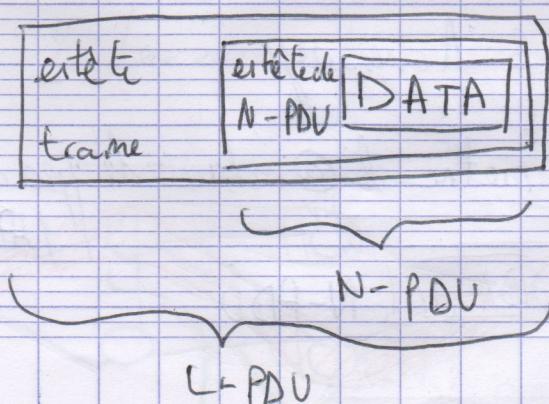
Permet de :

- Choisir une route plus ou moins rapide.
- Choisir les paquets à détruire si il y a congestion.

ex: dans IP il s'agit de 8 bits dans l'en-tête IP (TOS)



Chapitre 5: Fragmentation



Réseau

~~20/09/2013 Taille max d'un N-PDU~~

25/09/2013 ex: IP, c test 65535 octets (d)

Taille max d'un L-PDU:

ex : Ethernet, c'est 1500 octets

Qu'est ce qu'il se passe quand la taille du paquet est supérieure à la taille de la trame ?

\Rightarrow Nécessiter de fragmenter.

Notion de MTU: Maximum Transmission Unit = taille max d'un NPDU

Chaque enetisseur et récepteur sur une liaison doivent être au courant de ce MTU.

La fragmentation doit être gérée dans l'ététe du N-OSU

- identifiant du paquet original. — est-ce le dernier paquet ?
 - position du fragment dans le paquet original.

Deux possibilités pour le réassemblage sur le chemin,
ou à la destination. (ex: IP) (pas de réassemblage sur le
chemin mais que à la dest). Chaque fragment est indép
des autres.

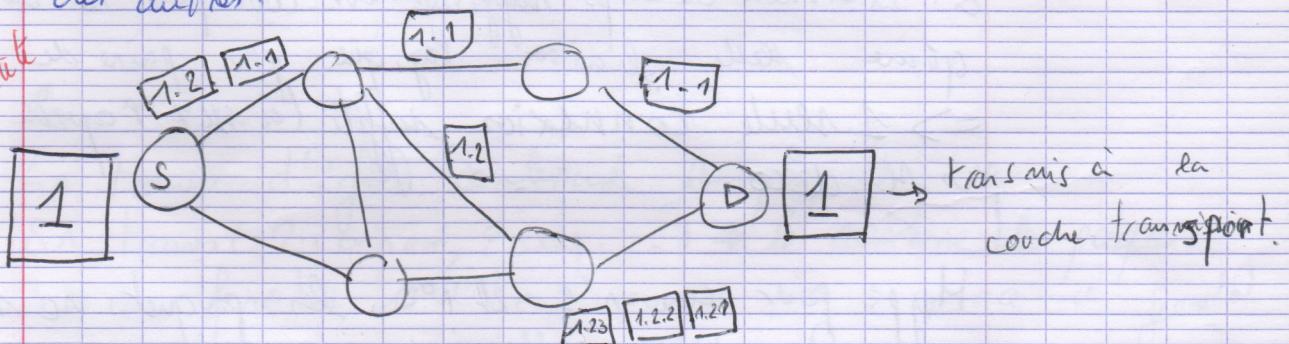
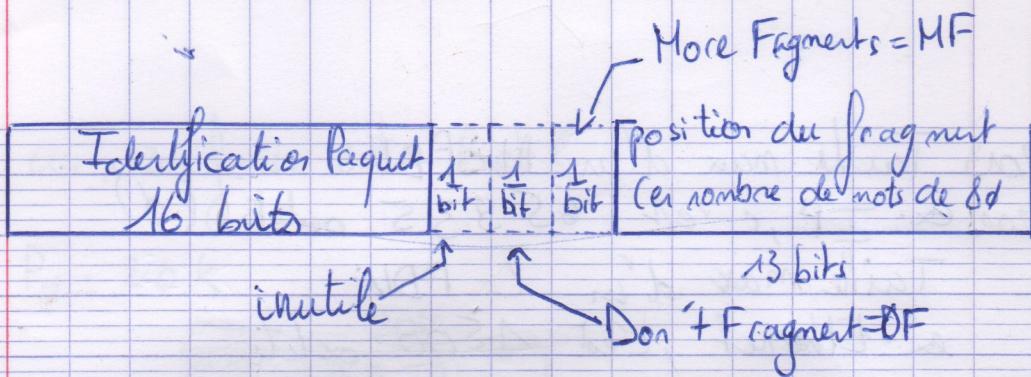


Illustration dans l'IP : hg



Pourquoi un C-PDU a une taille limite?

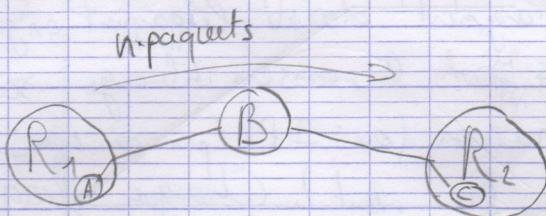
- à ce niveau CS on a la notion de frame
- les systèmes d'exploitation gèrent des zones de buffers de taille limite
- les lignes physiques limitent la taille pour fiabiliser la transmission et favoriser l'égalité

Exercice 6:

R₁ et R₂ sont reliés par un routeur B
deux réseaux

A et C échangent des paquets A cq nécessaires avant toute fermeture de connexion.

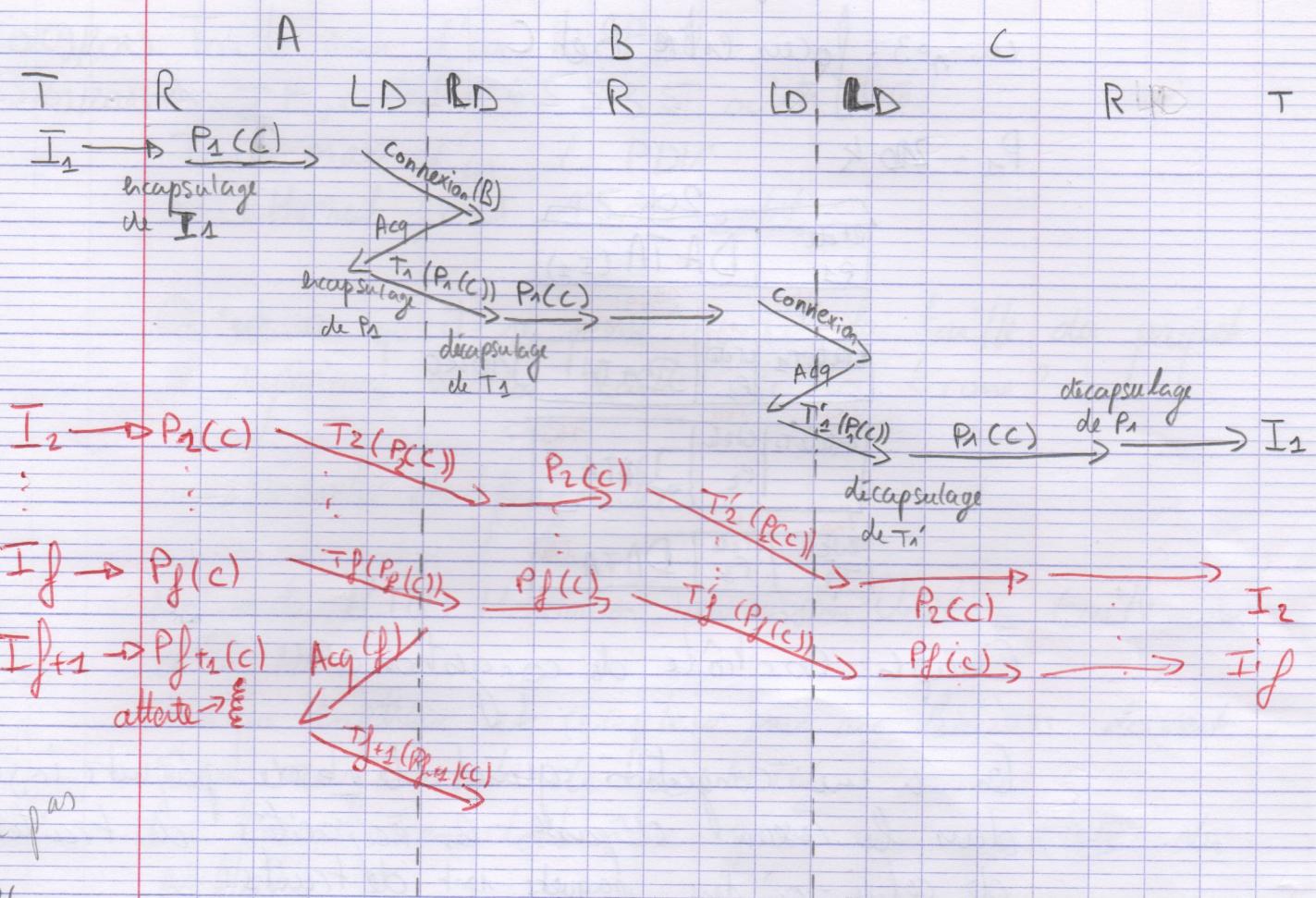
Orienté connexion
avant d'avoir
A doit se connecter
à B



a) Intervalle de temps suffisamment court pour ne demander qu'une seule liaison logique par paire de stations
=> 1 seule connexion suffit (à une rapide pour se pas se déco).

Hyp: pour cas n°1 et n°2 les paquets ne sont

- pas frag, MTU suffisent grande.
- si la taille de la fenêtre d'attente est g (le même pour A → B et B → C).



Cas n°1 :

- 2ⁿ frames d'informations contenant les paquets
- + 8 frames (2 connexion, 2 acq, 2 deconnexion, 2 ack).
- + 2 $\lceil \frac{n}{f} \rceil$ (acq A \rightarrow B et B \rightarrow C) par rapport à la fenêtre.

Cas n°2 :

- 2ⁿ frames d'infos contenant les paquets
- + 8ⁿ frames (2 connexion, 2 acq...) + le nombre de paquets.
- + 2ⁿ pour fenêtre 1. (il n'y a pas de mise en place de fenêtre).

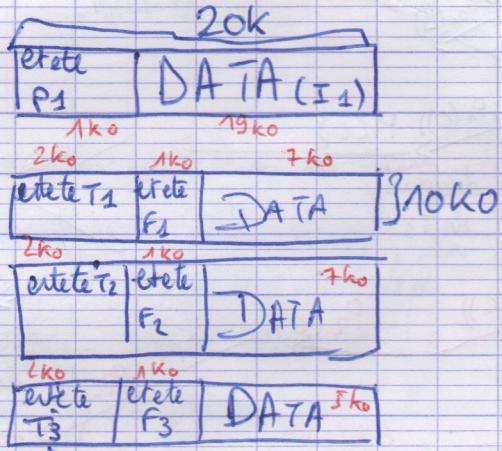
On était en mode déconnection (comme Ethernet : protocole de couche LD \rightarrow non connexion).

\rightarrow seuls acq).

$\Rightarrow 2^n$.

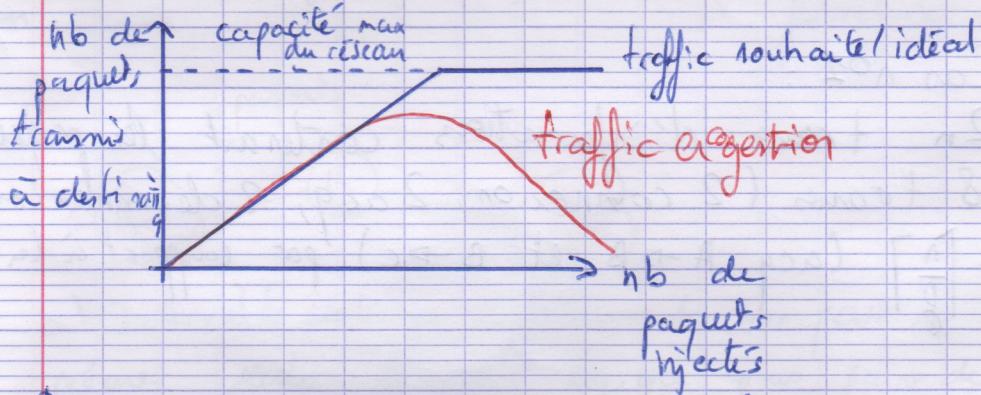
cas n°3 : focus entre B et C

P₁ : 20 k



Chap 6 : Contrôle de congestion

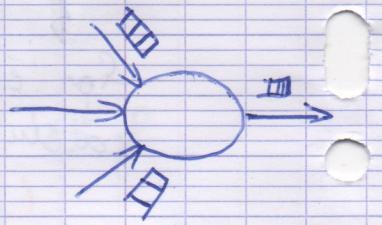
On a une congestion quand le nb de paquets injecté dans le réseau dépasse les capacités de traitement de celui-ci des paquets sont détruits



Raison de l'évolution des performances : à cause de la retransmission de paquets (effet d'avalanche).

Raison de la congestion :

- arrivées en même temps ; buffers pleins
- faibles performances des routeurs par rapport à toutes leurs fonctions,
- gérer les tables de routage
- gérer les fragmentations.



Réseaux

25/09/2013

- gestion des files d'attente

On a 2 voies d'action contre la congestion

1. Prévention.

- étudier le traffic attenue et dimensionner les équipements et les lignes! (phase d'étude des usages)
- contrôle d'admission (en mode connecté).
- créer le débit plus stable pour éviter des pics => forme de la canalisation de l'traffic

Ex: Algo du man percé



écoulement régulier
=> ni man plein, paquets perdus.

2. Guérison

Basée sur le retour d'informations:

- % de paquets détournés.
- nb de retransmissions
- % des emplis d'attente

Ces informations sont utilisées par les routeurs : ils signalent qu'ils ne peuvent/bouleverser plus recevoir de paquets
=> changement des tables de routage chez ses voisins et au delà

=> défraction préventive des paquets sélectionnés.
(etia, pio, faille...).

- les émetteurs peuvent réduire leur traffic.

Chap 7: Contrôle d'erreur

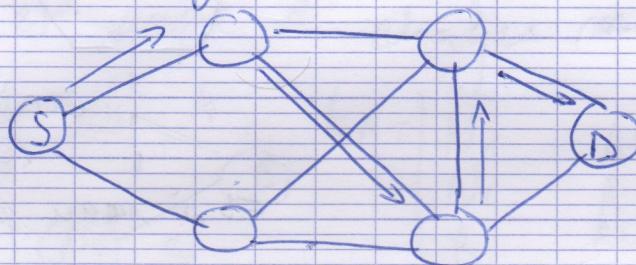
Généralement le contrôle d'erreur se fait seulement sur l'intégrité du paquet.

↳ il faut un champ dans l'en-tête pour gérer les erreurs.

ex: dans IP, c'est 16 bits (un CRC).

Si il y a erreur, on le détruit.

Chap 8: Routage



Consiste à trouver le meilleur chemin entre une source et une destination. En mode connecté on le fait en une seule fois alors qu'en mode non connecté on le fait pour chaque paquet.

C'est quoi un meilleur chemin ?

- le temps de traversée le plus court
- le moins sollicité
- le plus gros MTU sur le chemin (p. MTU)
- le plus redondé
- le moins de sauts
- prévenir la congestion
- le plus sûr (sécurité)
- le moins cher
- le plus fiable
- celui qui assure l'égalité entre utilisateurs.
- celui qui s'adaptera (parce qu'il chargera le mode non-connecté).

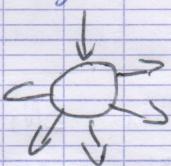
Algorithme de routage

- Statique ou Dynamique.
- locale ou Globale
 - meut au
sur le
routeur
(et ses voisins).
- Algorithme centralisé ou distribué
 - meut au
l'ensemble
du réseau.
une entité prend
la décision pour
tous.

Chaque décision prise
par un routeur est indép.

Algos dist. loc.

1. Algo par inondation.
Chaque noeud renvoie un paquet info vers toutes ses autres lignes.



Pb : gestion des boucles
- mauvaise util des ressources.) pas efficace pour les ressources.

- Avantages : - on est sûr qu'un paquet arrivera à destination par le plus court chemin
- pas obligé de connaître la destination.
- 2 - Routage adaptatif local.

Ex. Un routeur choisit les lignes de sorties fonction de critères pertinents locaux. Ex : % d'abitaculiers des lignes, longueur file d'att.

\Rightarrow mauvaise utilisation globale des ressources.

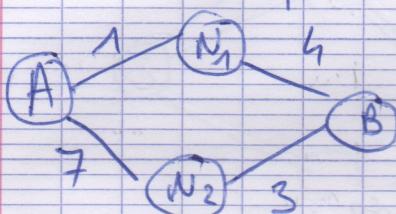
3. Routage adaptatif global

Chaque nœud calcule sa table de routage en fonction des infos locales mais aussi d'infos venant des autres nœuds.

Table de routage

C'est l'association entre une destination et le lien physique pour l'atteindre et le coût pour l'atteindre.

Elle est propre à chaque nœud.



$T_A =$	N_1	AN_1	1
	N_2	AN_2	7
	B	AN_2	10

$T_{N1} =$	A	$N_1 A$	1
	B	$N_1 B$	4
	N_2	$N_1 N_2$	7

$T_B =$	N_1	BN_1	1
	N_2	BN_2	3
	A	BN_2	10

$T_{N2} =$	A	$N_2 A$	7
	B	$N_2 B$	3
	N_1	$N_2 B$	7

Exercice 2 :

Ques 3.1

$T_{R1} =$	R ₂	$R_1 R_2$	0
	R ₃	$R_1 R_3$	0
	R ₄	$R_1 R_4$	1
	R ₅	$R_1 R_5$	0

$T_{R2} =$	R ₂	$R_2 R_1$	0
	R ₃	$R_2 R_3$	1
	R ₄	$R_2 R_4$	0
	R ₅	$R_2 R_5$	0

$T_{R3} =$	R ₁	$R_3 R_1$	0
	R ₂	$R_3 R_2$	1
	R ₄	$R_3 R_4$	0
	R ₅	$R_3 R_5$	0

$$S_{121} = 128.0.0.1$$

$$128.0.0.2$$

$$128.0.0.3$$

S_{121}	local	0
S_{140}	R ₁ R ₂	1
S_{150}	R ₁ R ₃	1
S_{153}	R ₁ R ₅	2
default	R ₁ N	-

S_{121}	R ₂ R ₁	1
S_{140}	local	0
S_{150}	R ₂ R ₅	2

S_{121}	R ₃ R ₁	1
S_{140}	R ₃ R ₅	1
S_{150}	local	0
S_{153}	R ₃ R ₄	1

S_{121}	R ₂ R ₁	1
S_{140}	R ₂ R ₅	1
S_{150}	R ₂ R ₄	-
default	R ₂ R ₁	-

S_{121}	R ₃ R ₁	1
S_{140}	R ₃ R ₅	1
S_{150}	local	0
S_{153}	R ₃ R ₄	1

S_{121}	R ₃ R ₁	1
S_{140}	R ₃ R ₅	1
S_{150}	local	0
S_{153}	R ₃ R ₄	1

Réseau

02/10/2013	$T_{R4} =$	R ₁	R ₄ R ₂	1
06/10/2013		R ₂	R ₄ R ₂	0
		R ₃	R ₄ R ₃	0
		R ₅	R ₄ R ₅	0
		S ₁₂₀	R ₄ R ₅	2
		S ₁₅₀	R ₄ R ₃	1
		S ₁₅₃	local	0
		S ₁₄₀	R ₄ R ₂	1
	<i>default</i>	R ₄ R ₁	-	

$T_{R5} =$	R ₁	R ₅ R ₁	0
	R ₂	R ₅ R ₂	0
	R ₃	R ₅ R ₃	0
	R ₄	R ₅ R ₄	0
	S ₁₂₈	R ₅ R ₁	1
	S ₁₅₀	R ₅ R ₃	1
	S ₁₅₃	R ₅ R ₄	1
	S ₁₄₀	R ₅ R ₂	1
	<i>default</i>	R ₅ R ₁	-

$\frac{1}{2} T_{R1} =$	R ₂	R ₁ R ₃	3
	R ₃	R ₁ R ₅	3
	R ₄	R ₁ R ₂	3
	R ₅	R ₁ R ₂	3

$T_{R2} =$	R ₁	R ₂ R ₄	3
	R ₃	R ₂ R ₄	1
	R ₄	R ₂ R ₅	2
	R ₅	R ₂ R ₄	2

$T_{R3} =$	R ₁	R ₃ R ₅	1
	R ₂	R ₃ R ₅	2
	R ₄	R ₃ R ₄	0
	R ₅	R ₃ R ₅	0

$T_{R4} =$	R ₁	R ₄ R ₃	2
	R ₂	R ₄ R ₂	0
	R ₃	R ₄ R ₃	0
	R ₅	R ₄ R ₃	1

$T_{R5} =$	R ₁	R ₅ R ₁	0
	R ₂	R ₅ R ₄	1
	R ₃	R ₅ R ₂	2
	R ₄	R ₅ R ₃	1

Algo de routage distribué. (algo adaptatif global)
 Exemple de routage RIP: Routing Internet Protocol
 Permet de construire et mettre à jour les tables de routage. Il est basé sur l'échange de tables entre voisins directs. L'info sur une partie se propage de proche en proche.

L'algorithme qui réalise l'affichage de tables est l'algorithme Bellman-Ford

Hop: coût d'une liaison = +∞ si noeud est inaccessible

Principe : - A intervalles réguliers un nœud envoie l'ensemble des couples (destination, coût) à ses voisins qu'il connaît (qui sont dans sa table de routage).

A la réception d'un message par un nœud N , il y a mise à jour de sa propre table de routage
le message est de la forme (D, c') et il vient de N' .

- Si D est pas déjà dans la table de routage
on ajoute : $(D, NN', c' + \text{cout de } (NN'))$

- Si D existe dans la table, il y a une entrée dans la table (D, NN', c'')

Si $c' + \text{cout}(N, N') < c''$
on remplace l'entrée existante par $(D, NN', c' + \text{cout}(N, N'))$

- Sinon on ne fait rien

Si $N' = N$, alors on met à jour le coût $\rightarrow c' \leftarrow c' + \text{cout}(N, N')$

Lors de ~~l'explosion~~ d'un nœud il suffit qu'il soit reconnu par ses voisins

La forme d'un nœud est plus délicate. Les voisins s'en aperçoivent si ils se ressoutent pas la table au bout du temps impartie (2 mn dans RIP) \rightarrow la valeur too est positionnée tous les chemins qui passent près de ce voisin.

Ex 3. h. 1)

$t_{1A} =$	1B	1A1B	0	2D	1A1B	3	4C	1A1C	3
	1C	1A1C	0	3A	1A1B	3	4D	1A1C	4
	1D	1A1B	1	3B	1A1B	4	4E	1A1C	2
	2A	1A1B	1	3C	1A1B	4	5A	1A1B	5
	2B	1A1B	2	4A	1A1C	1	5B	1A1C	4
	2C	1A1B	2	4B	1A1C	2	5C	1A1B	5

h. 2).

Out 1A1B
→ 5

$t_{1A} =$	1B	1A1B	0
	1C	1A1C	0
	1D	1A1B	1
region 2	1A1B	1	
region 3	1A1B	3	
region 4	1A1C	1	
region 5	1A1C	4	

$t_{1B} =$	1A	1B1A	0
	1C	1B1A	2
	1D	1B1D	0
region 2	1B2A	0	
region 3	1B2A	3	
region 4	1B1D	1	
region 5	1B1C	3	

$t_{2A} =$	2B	2A2B	0
	2C	2A2C	0
	2D	2A2C	1
region 1	2A1B	0	
region 3	2A2C	1	
region 4	2A1B	2	
region 5	2A2C	2	

$t_{3A} =$	3B	3A2C	0
	3C	3A5A	0
region 1	3A5A	2	
region 2	3A2C	2	
region 4	3A3B	0	
region 5	3A3C	0	

$t_{4A} =$	4B	4A4B	0
	4C	4A4B	1
	4D	4A4E	1
	4E	4A4E	0
region 1	4A4C	0	
region 2	4A4C	3	
region 3	4A4E	4	
region 5	4A4E	2	

$t_{5A} =$	5B	5A5B	0
	5C	5A5C	0
region 1	5A5A	3	
region 2	5A3A	1	
region 3	5A3A	0	
region 4	5A5B	1	

Routage hiérarchique :

Avantages : - les tables sont plus compactes

- marquage de ce qu'il se passe à l'intérieur (réau administratif).

Inconvénients : - les routes ne sont pas forcément optimales

Protocole IP

C'est un protocole de niveau 3 (cad. réseau)

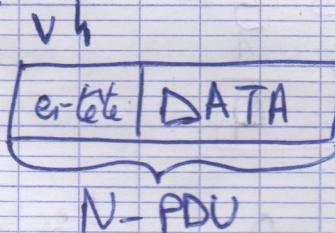
Il reste peu normalisé, c'est une norme de fait.

Il est décrit dans un RFC (Request for comment).

C'est un protocole en mode "non connecté".

Note : Cheminement entre source et destination.

Entête IP.



4 bits version du protocole de couche réseau	8 bits IHL	TOS	Taille du Paquet en nb d'octets
--	---------------	-----	------------------------------------

20 octets

F R A G M E N T A T I O N

TTL	protocole de niveau supérieur TCP / UDP	C H E C K S U M
	adresse IP source	
	adresse IP destination	

version de IP : ex: 4.

IHL : Internet header length : taille de l'entête

un nombre de mot de
32 bits, valeur
entre 5 et 15
(max op) (avec op)

résumé

- 04/10/2013
- TOS : gêne la priorisation des paquets
 - Taille du paquet max = $2^{16}-1$ octets
= 65535
 - TTL : Time to live : Décrément de 1 à chaque saut, quand il atteint 0 le paquet est détruit
 - Checksum : contrôle erreur entête CRC
 - @ IP : sur 4 octets.

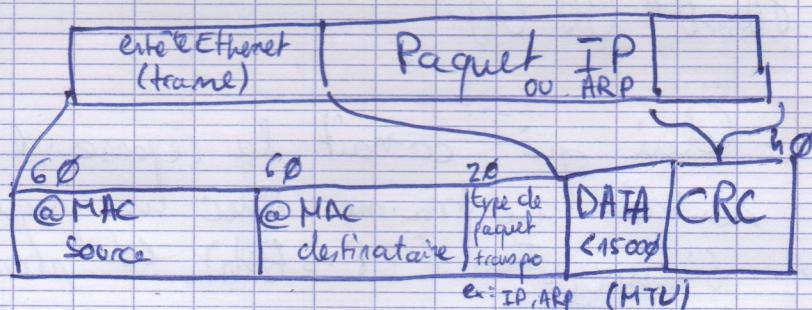
Protocole ARP: Address Resolution Protocol

Protocole de niveau 3

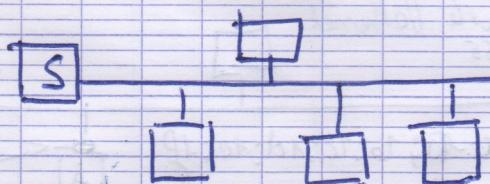
Rôle : Trouver l'adresse de niveau liaison de données à partir d'une adresse de niveau réseau.

Eax: Trouver l'@Mac (Ethernet) à partir d'une @IP.

Ethernet :



Ethernet est un protocole de niveau LD qui fonctionne par diffusion et en mode non connecté.

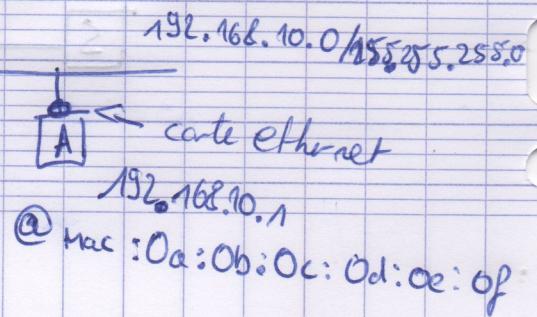
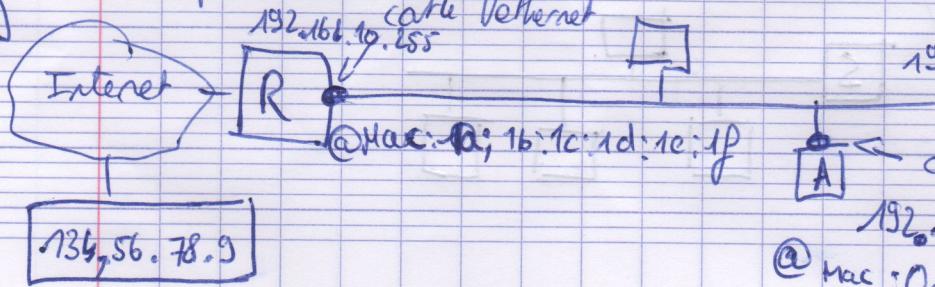


- Format d'un paquet ARP :
- HW Type : Hardware Type
 - Proto de Niv 3
 - longueur de l'adresse HW
 - niv3
 - Opérateur Request / Response
 - @ de niveau liaison de données du demandeur
 - @ IP du demandeur
 - @ IP demandée
 - @ de niveau LD demandé
- Ex: Ethernet
- Ex: IP
- Ex: 6 ↘
- Ex: 4 ↘
- Ex: @ Mac / Ethernet
- Ex: @ IP
-
- Ex: 0 si c'est une requête, valeur @ Mac si réponse
- Fonctionnement: Une machine cherche l'adresse Ethernet d'une machine prochaine destination. Elle envoie un frame Ethernet sur le réseau avec une requête ARP (l'@MAC de cette frame est d'@ avec de diffusion ff·ff·ff·ff·ff·ff).

- Celui qui connaît la réponse : (l'association IP → Ethernet) envoie une branche avec un paquet ARP réponse.
- L'association (IP → Eth) est stockée dans un cache ARP.



Exemple récapitulatif :



Question: Que se passe t'il dans le réseau local et dans le routeur quand un paquet doit être envoyé entre la machine A et la machine 134.56.78.9?

A: regarde dans sa table de routage pour attendre @ dest
→ il faut passer par R, relié par ethernet, construction
de la frame Eth contenant le paquet IP → 1^{ère}
frame est une requête ARP, on répond à A (R, autre machine);
il connaît maintenant @ Mac du routeur.

R: il regarde si il y a des prob, il peut fragmenter (tout
comme A). (Tout dépend de la taille du paquet).