Le protocole IP

Internet Protocol

(Protocole Internet)

Introduction

- Les différents réseaux hétérogènes d'Internet coopèrent grâce au protocole IP.
- IP permet l'identification de tout équipement (grâce à l'adressage IP).
- IP permet l'échange de datagrammes entre tout couple d'équipements.
- Objectif : faire le mieux possible pour transmettre les datagrammes de leur source vers leur destination.

Communication via IP

- La couche transport (protocole TCP) découpe le flux de données en datagrammes IP.
- Chaque datagramme est transmis au travers du réseau Internet. Il peut être re-découpé en fragments IP.
- À destination, tous les morceaux sont ré-assemblés par la couche réseau pour recomposer le datagramme.
- La couche transport reconstitue le flux de données initial pour la couche application.

Le service offert par IP

Le service offert par le protocole IP est dit non fiable :

- **remise de paquets non garantie,**
- **sans connexion (paquets traités indépendamment les uns des autres),**
- **pour le mieux (best effort, les paquets ne sont pas éliminés sans raison).**

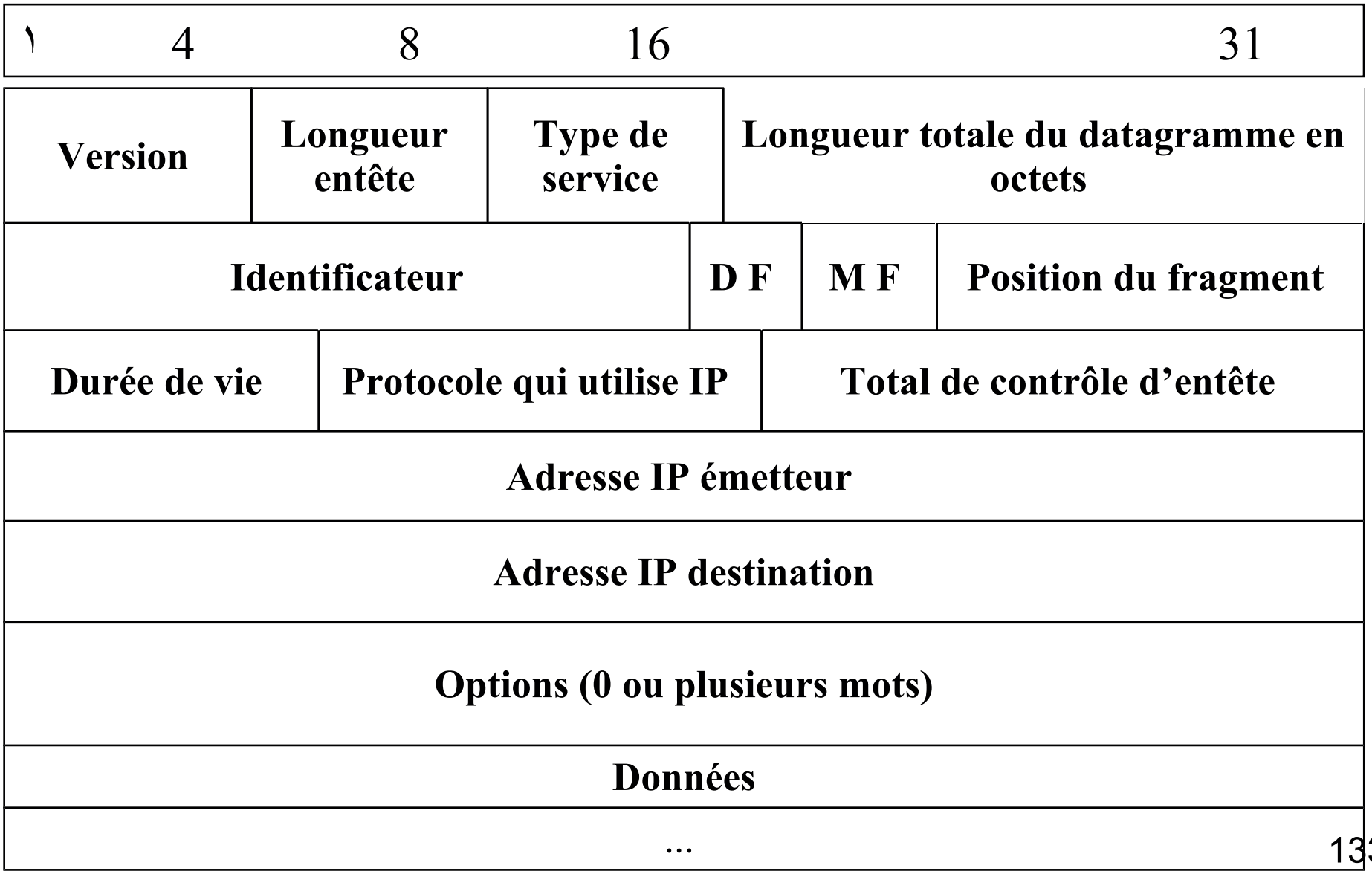
Datagramme IP

Constitué de deux parties : un entête et des données.

En-tête : partie fixe (20 octets) + partie optionnelle Variable

Données : charge utile du datagramme

Format du Datagramme IP



Datagramme IP

Chaque ordinateur convertit l'en-tête de sa représentation locale vers la représentation standard d'Internet.

Le récepteur effectue la conversion inverse.

Le champ *données* n'est pas converti.

Champs d'en-tête

Version : numéro de la version du protocole utilisé pour créer le datagramme (4 bits) : version 4 ou 6.

Longueur entête : longueur de l'en-tête exprimée en mots de 32 bits (égal à 5 s'il n'y a pas d'option)

Type de service : précise le mode de gestion du datagramme (8 bits)

- **Priorité : 0 (normal) → 7 (supervision réseau) (3 bits)**
- **Indicateurs D, T, R : indiquent le type d'acheminement désiré du datagramme, permettant à un routeur de choisir entre plusieurs routes (si elles existent) :**
  D signifie délai court, T signifie débit élevé et R signifie grande fiabilité.
- **2 bits inutilisés**

Champs d'en-tête

Longueur totale : en octets (16 bits)

Identificateur : permet au destinataire de savoir à quel datagramme appartient un fragment (16 bits)

Drapeau : 3 bits

- **DF : " Don't fragment "**
- **MF : " More fragments "**
- **1bit inutilisé**

Champs d'en-tête

Position du fragment : localisation du déplacement du fragment dans le datagramme (13 bits)

Durée de vie (TTL) : compteur utilisé pour limiter la durée de vie des datagrammes (8 bits). Nombre maximal de routeurs que ledatagramme peut traverser :

- **décrémenté à chaque saut**
- **détruit quand il passe à 0**

Protocole : indique par un numéro à quel protocole confier le contenu du datagramme (8 bits)

- **6 = TCP, 17 = UDP, 1 = ICMP.**
- **Protocole de niveau supérieur ayant créé le datagramme**

Champs d'en-tête

Total de contrôle d'en-tête : vérifie la validité de l'en-tête, doit être recalculé à chaque saut (16 bits)

Adresse IP source : 32 bits

Adresse IP destination : 32 bits

Les options du datagramme

Le champ OPTIONS est facultatif et de longueur variable. Les options concernent essentiellement des fonctionnalités de mise au point. Une option est définie par un champ octet

H	classe d'option	Numéro d'option

copie (C) indique que l'option doit être recopiée dans tous les fragments (c=1) ou bien uniquement dans le premier fragment (c=0).

les bits classe d'option et numéro d'option indiquent le type de l'option et une option particulière de ce type

Les options du datagramme

Enregistrement de route (classe = 0, option = 7)
Oblige chaque passerelle/routeur d'ajouter son adresse dans la liste.

Routage strict prédéfini par l'émetteur (classe = 0, option = 9) prédéfinit le routage qui doit être utilisé dans l'interconnexion en indiquant la suite des adresses IP.

Routage lâche prédéfini par l'émetteur (classe = 0, option = 3) autorise, entre deux passages obligés, le transit par d'autres intermédiaires.

Horodatage (classe = 2, option = 4) permet d'obtenir les temps de passage (*timestamp*) des datagrammes dans les routeurs. Exprimé en heure et date universelle

Le routage

- **Pour passer d'une machine source à une machine destination, il peut être nécessaire de passer par plusieurs points intermédiaires.**

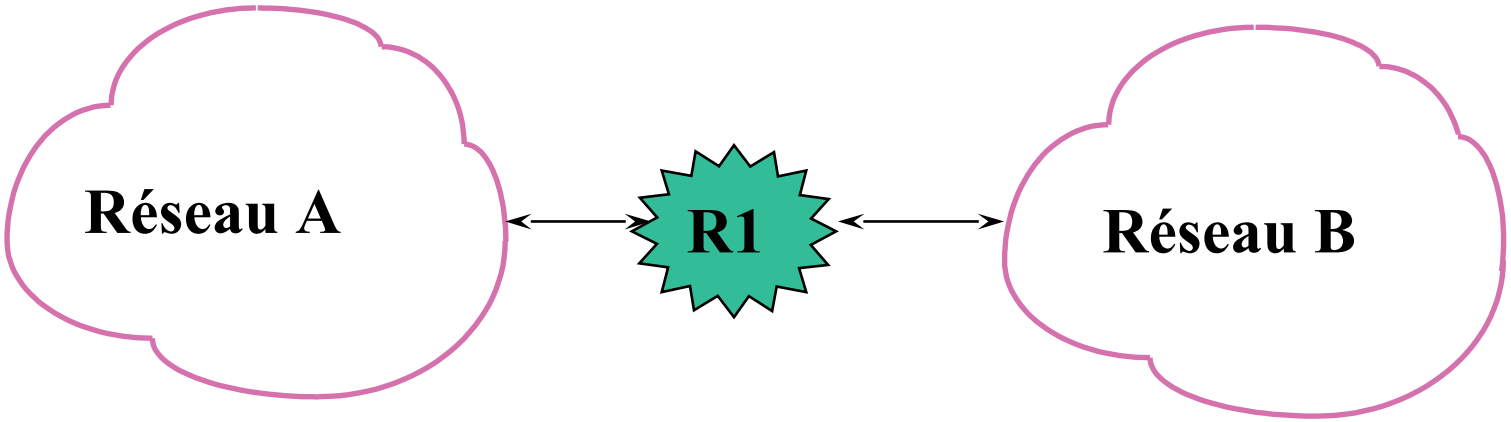
- **Pour faire transiter des informations depuis un réseau vers un autre réseau on utilise des passerelles ou le plus souvent des routeurs.**

- **Les routeurs possèdent une connexion sur chacun des réseaux qu'ils interconnectent.**

Routeurs

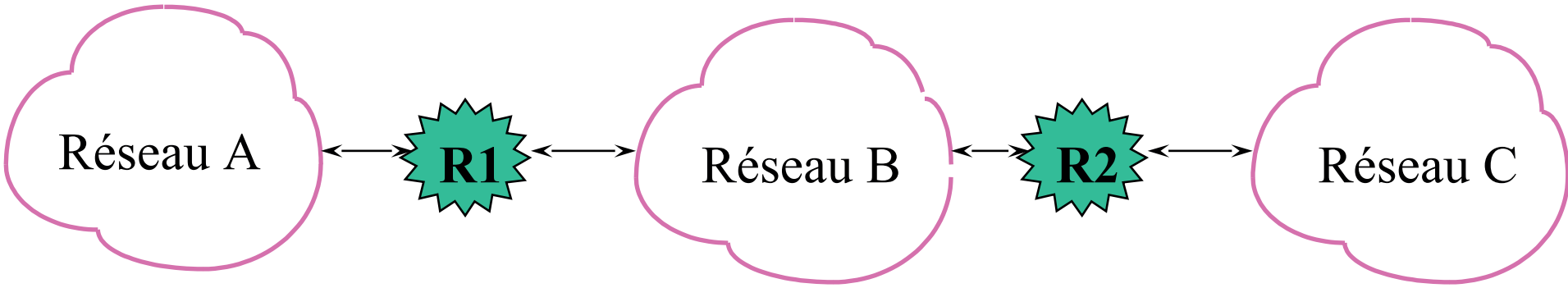
- **Les routeurs disposent de plusieurs adresses IP et plusieurs adresses physiques :**
  - **1 adresse physique par interface**
  - **1 adresse IP par réseau.**
- **Ils sont chargés de l'acheminement des paquets IP.**

Interconnexion simple



R1 transfère sur le réseau B, les paquets circulant sur le réseau A et destinés au réseau B

interconnexion multiple



- **R1 transfère sur le réseau B, les paquets circulant sur le réseau A et destinés aux réseaux B et C.**
- **R1 doit avoir connaissance de la topologie du réseau, à savoir que C est accessible depuis le réseau B.**
- **Le routage n'est pas effectué sur la base de la machine destinataire mais sur la base du réseau destinataire**

Problème du routage

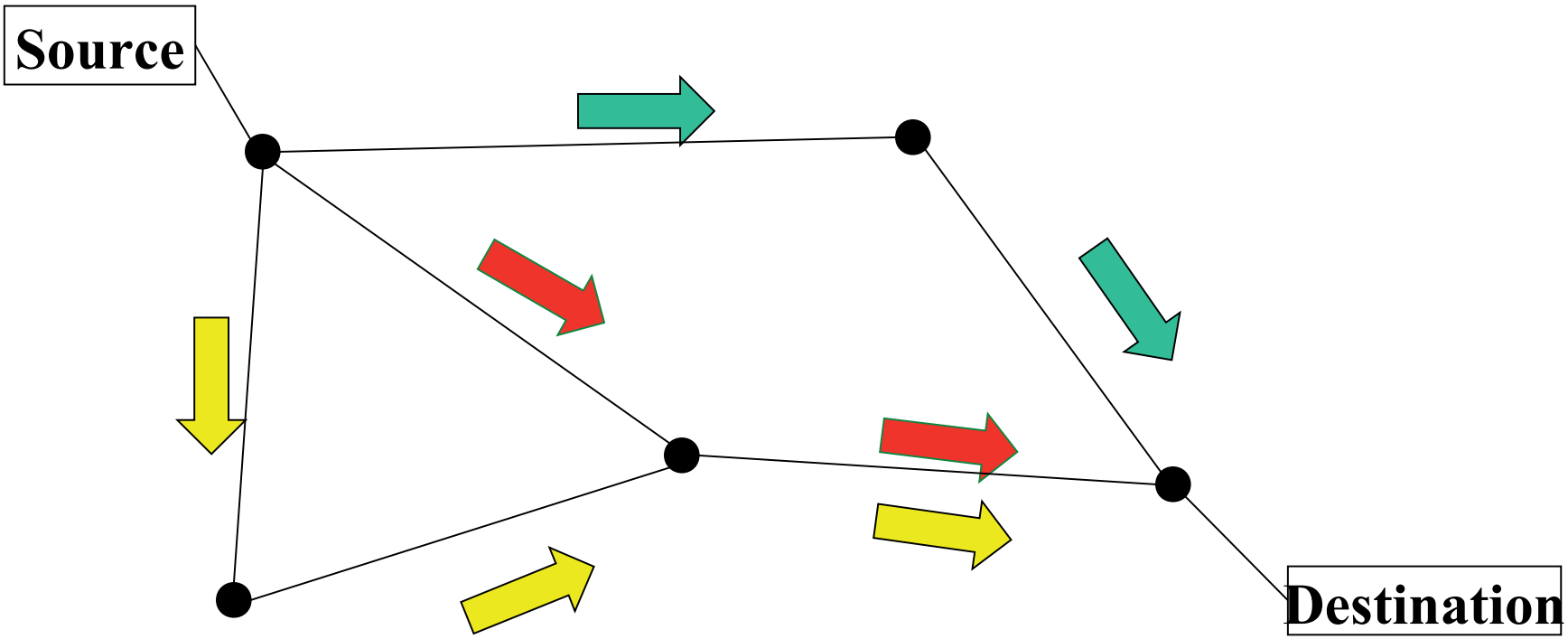


Table de routage

- **Les routeurs décident de la route à faire suivre aux paquets IP par consultation d'une table de routage.**
- **La maintenance des tables de routages est une opération fondamentale. Elle peut être manuelle, statique ou dynamique.**

Table de routage

Ne contiennent que les identifiants réseau des adresses IP.

La table contient, pour chaque numéro de réseau à atteindre, l'adresse IP du routeur le plus proche.

Chaque routeur possède une liste de couples (réseau, 0) [définit comment accéder à un réseau distant] ou (ce_réseau, ordinateur) [comment accéder à un ordinateur du réseau local].

Table de routage (exemple)

destination	masque	passerelle	interface
10.1.90.1	255.255.255.255	10.1.50.9	10.1.50.1
10.1.40.0	255.255.255.0	10.1.40.2	10.1.40.1
10.1.50.0	255.255.255.0	10.1.50.1	10.1.50.1
10.1.60.0	255.255.255.0	10.1.60.1	10.1.40.1
10.1.70.0	255.255.255.0	10.1.40.8	10.1.40.1
10.1.80.1	255.255.255.0	10.1.50.9	10.1.50.1
0.0.0.0	0.0.0.0	127.0.0.1	127.0.0.1

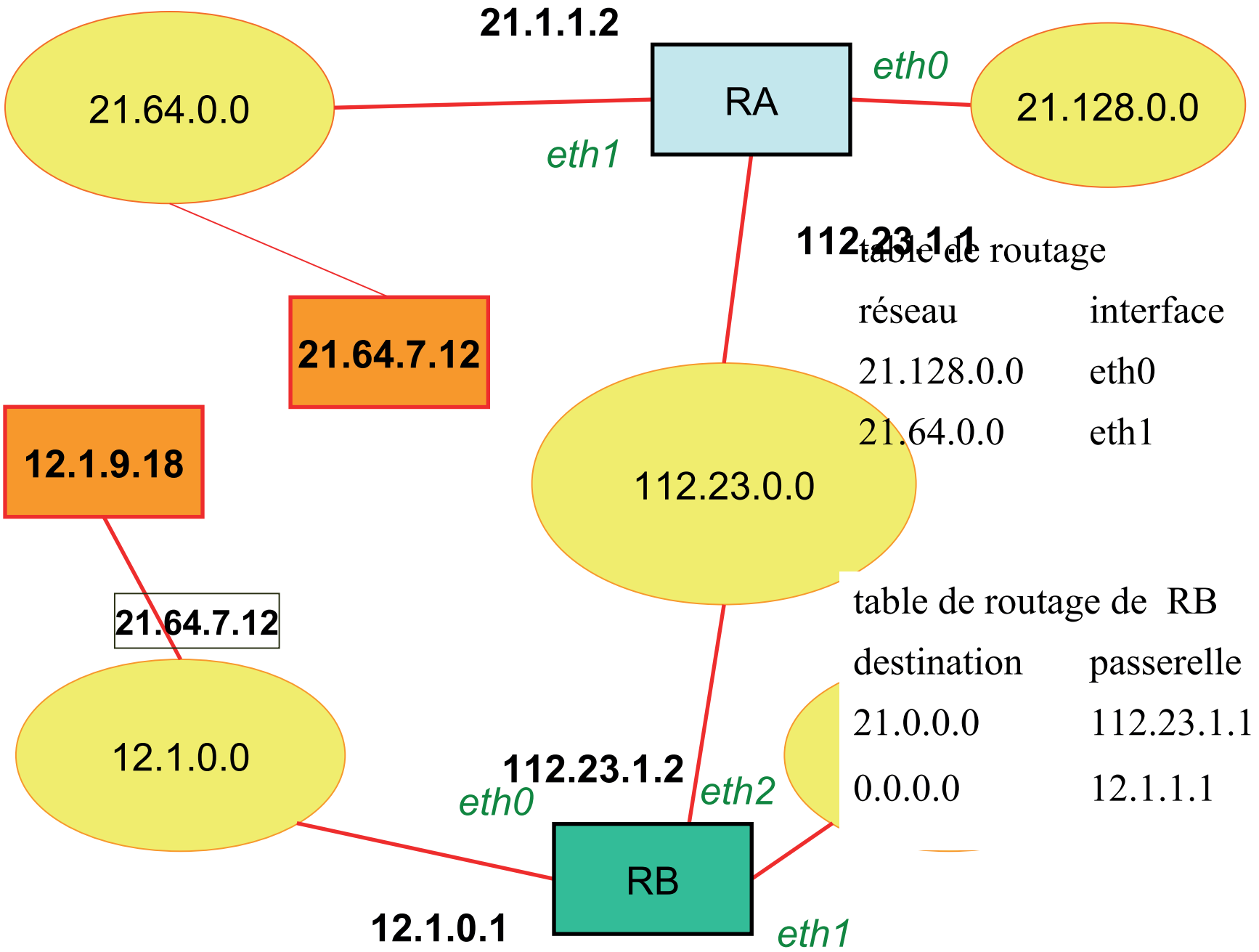


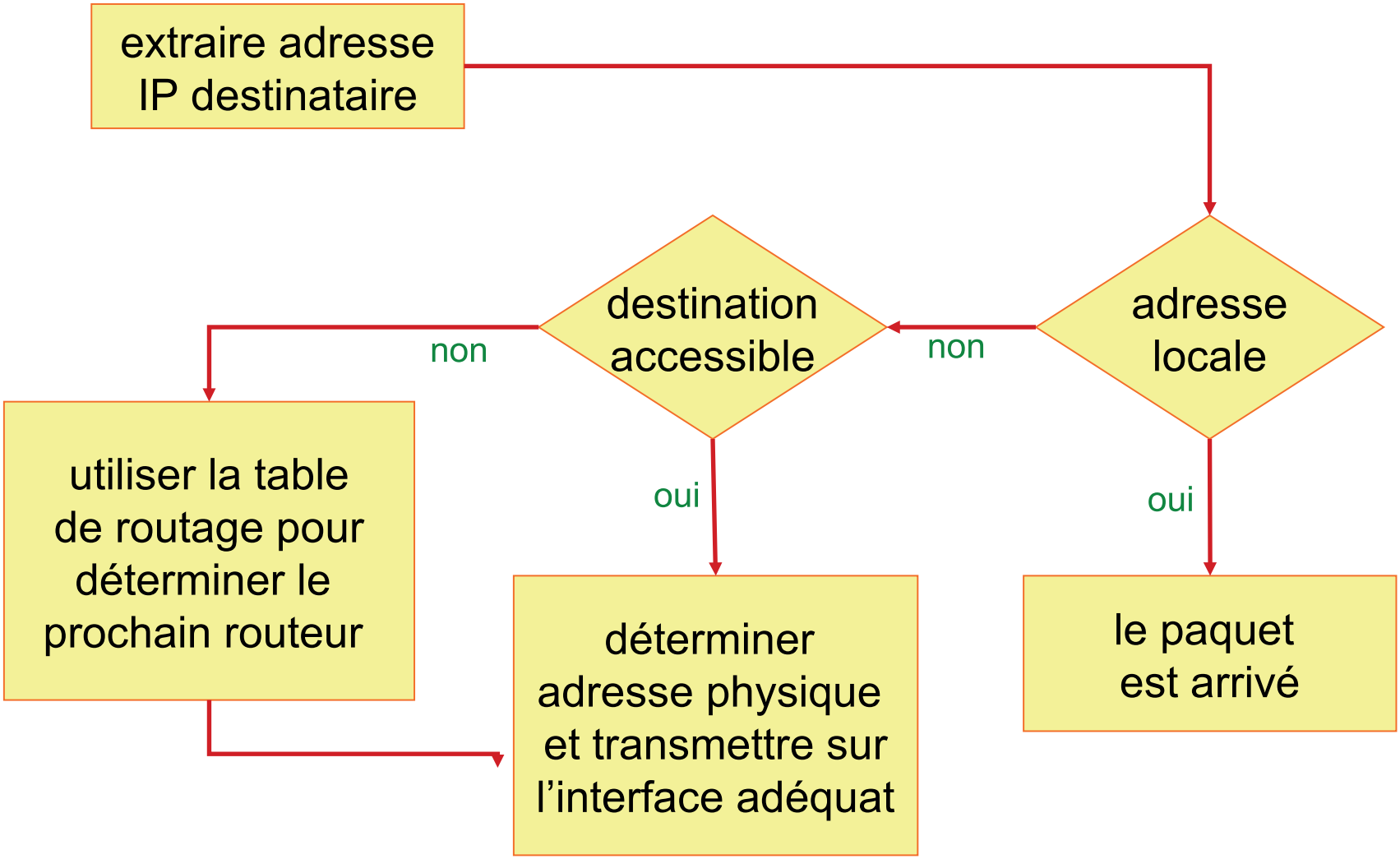
table de routage

réseau	interface
21.128.0.0	eth0
21.64.0.0	eth1

table de routage de RB

destination	passerelle
21.0.0.0	112.23.1.1
0.0.0.0	12.1.1.1

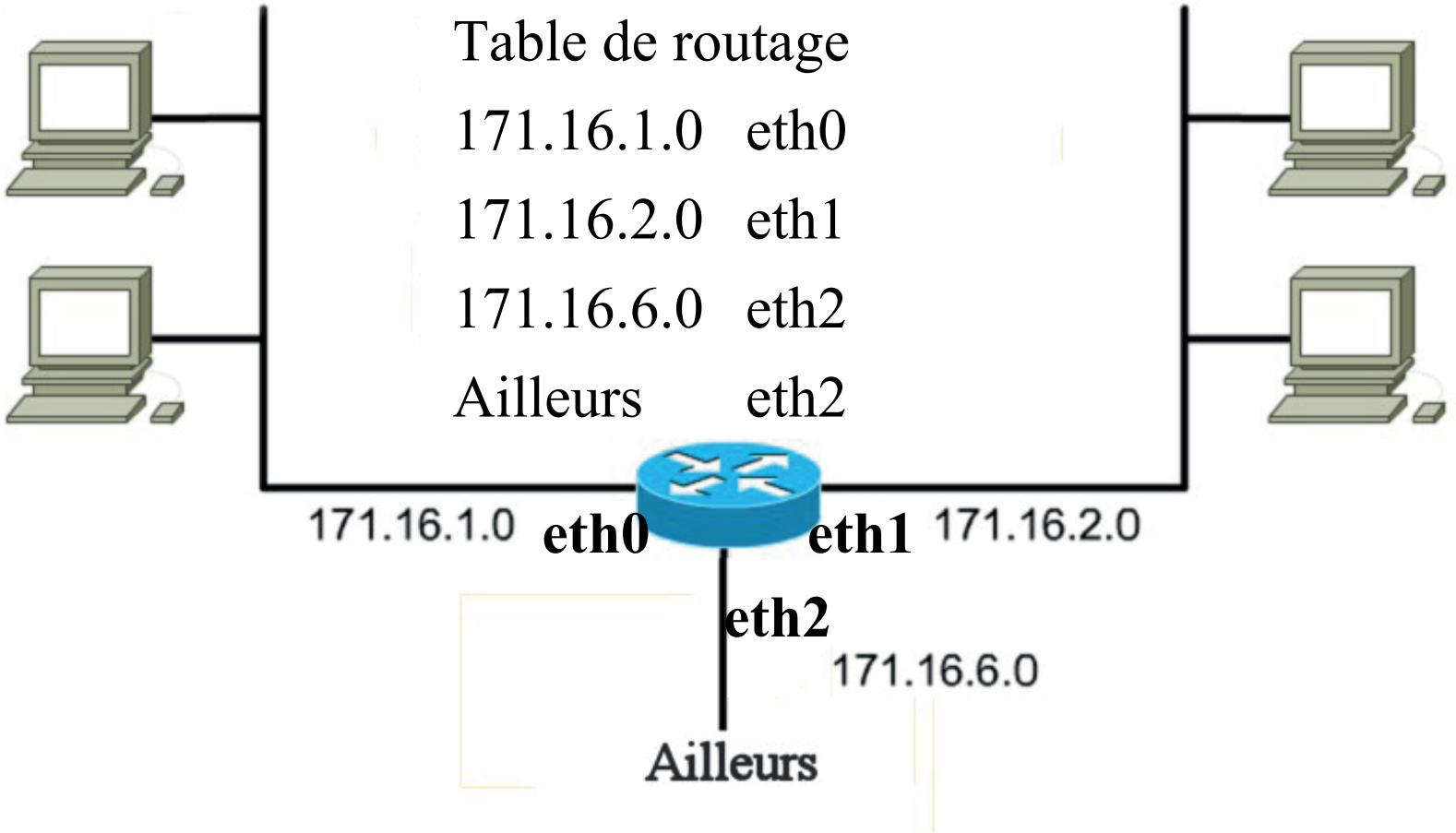
Méthode de routage



Route par défaut

Une route par défaut permet d'acheminer un paquet dont la destination ne correspond à aucune autre route de la table de routage

Route par défaut



Objet du routage

Trouver (calculer) le plus court chemin à emprunter d'une source à une destination

Distance ?

- **nombre de sauts**
- **distance kilométrique**
- **temps moyen de transmission**
- **longueur moyenne des files d'attente**

Propriétés d’un algorithme de routage

Exactitude

Simplicité

Robustesse (capacité d'adaptation aux pannes et changement de topologie)

Stabilité (convergence vers un état d'équilibre)

Optimisation

Classes d'algorithmes de routage

Comment un routeur peut-il connaître les différents chemins le reliant aux autres routeurs ?

- **routage statique**
- **routage dynamique (protocole de routage)**

Routage Statique

L’administrateur réseau spécifie manuellement la table de routage

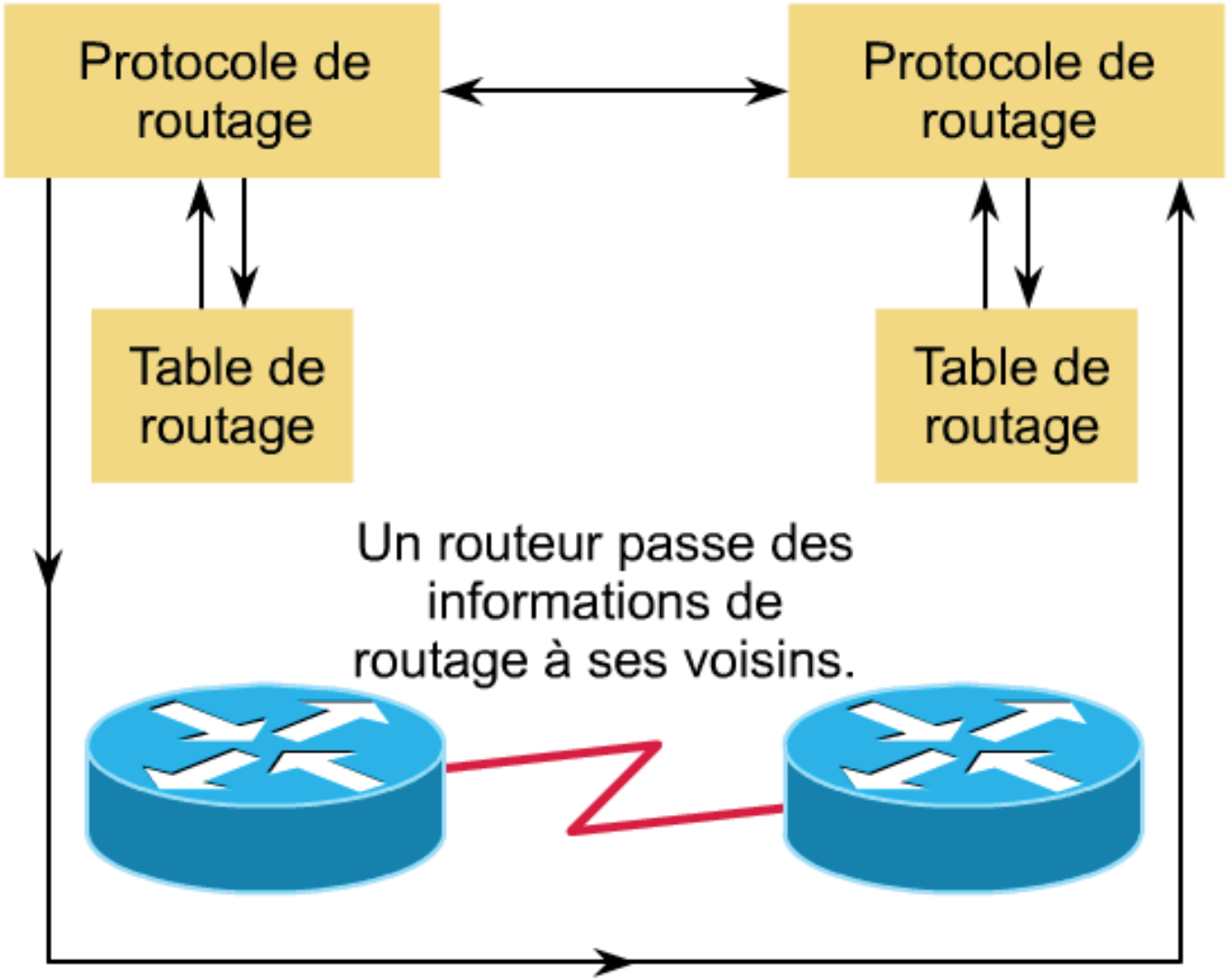
- **<u>Inconvénients :</u> l’administrateur doit faire les mises à jour en cas de changement de la topologie du réseau**
- **<u>Avantages :</u> réduction de la charge du système, car aucune mise à jour de routage n'est envoyée**

Routage dynamique

L'administrateur réseau met en place un protocole de routage établissant automatiquement les chemins entre deux routeurs

- **<u>Inconvénients :</u> augmentation de la charge du système, car des mises à jour de routage doivent êtres envoyées**
- **<u>Avantages :</u> prise en compte automatique d'un changement de la topologie du réseau**

Routage dynamique



Le protocole UDP

User Datagram Protocol

Introduction

La couche transport d'Internet dispose de deux protocoles pour la communication entre applications :

UDP : protocole en mode sans connexion

TCP (Transmission Control Protocol) : protocole en mode orienté connexion

UDP

- **Service en mode non connecté**
- **Livraison des messages sans garantie**
- **Ordonnancement et arrivé des messages non garanti**

UDP

Identification du service : les ports

les adresses IP désignent les machines entre lesquelles les communications sont établies. Lorsqu'un processus désire entrer en communication avec un autre processus, il doit adresser le processus s'exécutant cette machine.

L'adressage de ce processus est effectué selon un concept abstrait indépendant du système d'exploitation :

- les processus sont créés et détruits dynamiquement sur les machines, il faut pouvoir remplacer un processus par un autre sans que l'application distante ne s'en aperçoive,
- il faut identifier les destinations selon les services offerts, sans connaître les processus qui les mettent en oeuvre,
- un processus doit pouvoir assurer plusieurs services.

UDP : les ports

Ces destinations abstraites permettant d'adresser un service applicatif s'appellent des ports de protocole.

***Port* : entier identifiant l'application à laquelle la couche transport doit remettre les messages**

L'émission d'un message se fait sur la base d'un port source et un port destinataire.

Les processus disposent d'une interface système leur permettant de spécifier un port ou d'y accéder.

Les accès aux ports sont généralement synchrones, les opérations sur les ports sont « tamponnés » (files d'attente).

Datagrammes UDP

Port source	Port destination
Longueur	Checksum
Données	

Port source : optionnel (identifie un port pour la réponse)
Port destination : numéro de port (démultiplexage)
Longueur : longueur totale du datagramme en octets (8 au minimum)
Checksum : optionnel (seule garantie sur la validité des données)

Classement des ports

`1-1023` : services réservés s'exécutant avec des droits privilégiés (*root*)

`1024-49151` : services enregistrés auprès de l'IANA et pouvant s'exécuter avec des droits ordinaires

`49152-65535` : libres de toutes contraintes

Numéros de ports

Les derniers numéros de ports peuvent être obtenus sur le site de IANA :
http://www.iana.org/assignments/port-numbers

Sous Linux le fichier /etc/services contient les numéros de ports et les services associés.

Les ports standards

Certains ports sont réservés

N° port	Mot-clé	Description
J	ECHO	Echo
11	USERS	Active Users
13	DAYTIME	Daytime
37	TIME	Time
42	NAMESERVER	Host Name Server
53	DOMAIN	Domain Name Server
MJ	BOOTPS	Boot protocol server
68	BOOTPC	Boot protocol client
69	TFTP	Trivial File Transfer protocol
123	NTP	Network Time Protocol
161	SNMP	Simple Network Management prot.

D'autres numéros de port (non réservés) peuvent être alloués dynamiquement aux applications.

Le protocole TCP/IP

Transport Control Protocol

Introduction

S'appuie sur IP (réseau non fiable)

Communication en mode connecté

- **Ouverture d'un canal**
- **Communication Full-Duplex**
- **Fermeture du canal**

TCP doit :

- **assurer la délivrance en séquence (Arrivée et ordre garanties)**
- **contrôler la validité des données reçues**
- **organiser les reprises sur erreur**
- **réaliser le contrôle de flux**

Connexion

une connexion de type circuit virtuel est établie

connexion = une paire d'extrémités de connexion

extrémité de connexion = couple (adresse IP, numéro port)

Exemple de connexion : ((124.32.12.1, 1034), (19.24.67.2, 21))

Une extrémité de connexion peut être partagée par plusieurs autres extrémités de connexions (multi-instanciation)

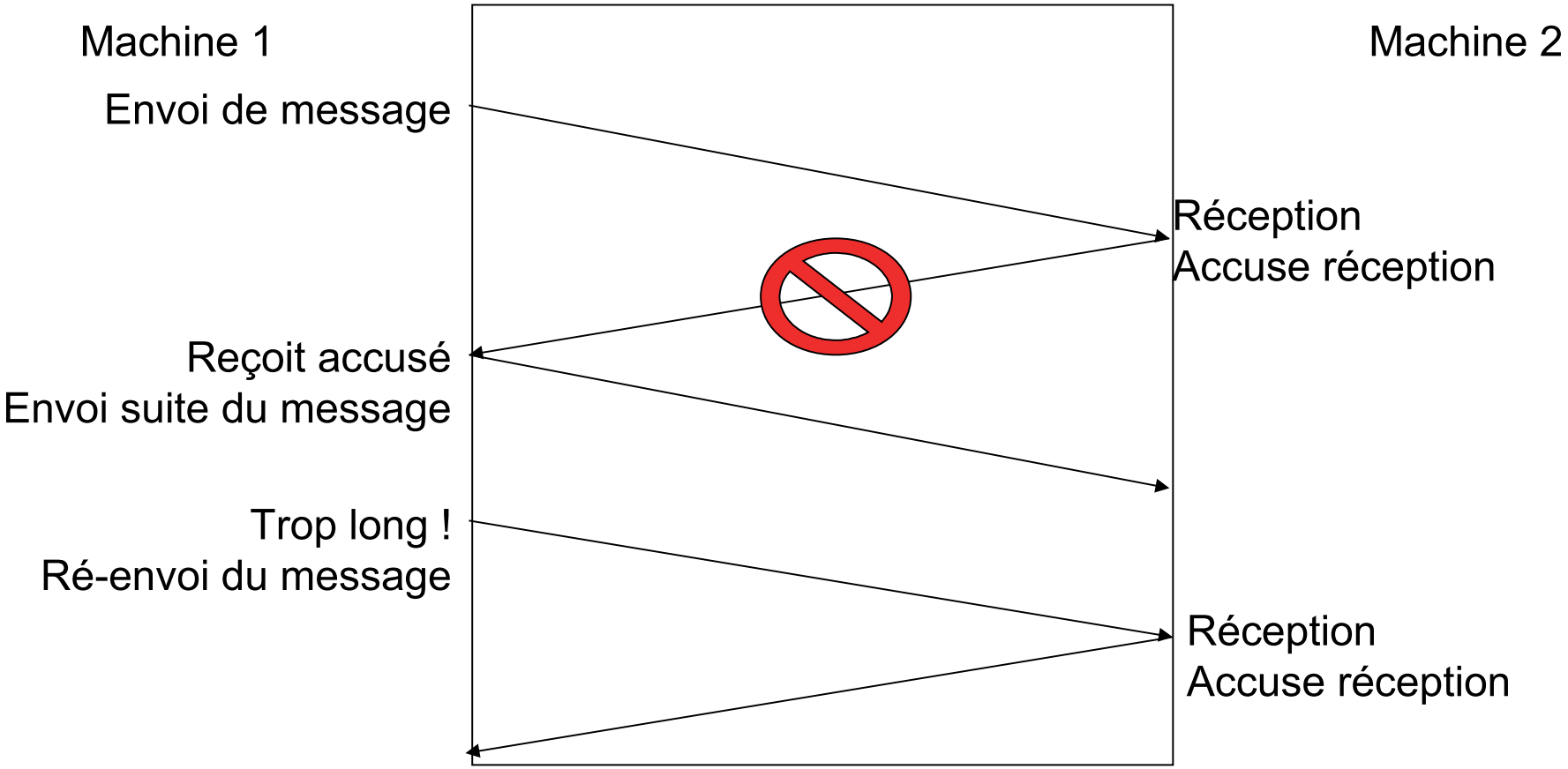
La mise en oeuvre de la connexion se fait en deux étapes :

- **une application (extrémité) effectue une ouverture passive en indiquant qu'elle accepte une connexion entrante,**
- **une autre application (extrémité) effectue une ouverture active pour demander l'établissement de la connexion.**

Garantie de réception

Comment savoir si un paquet arrive ?

➔ Accusé de réception



Utilisation du réseau

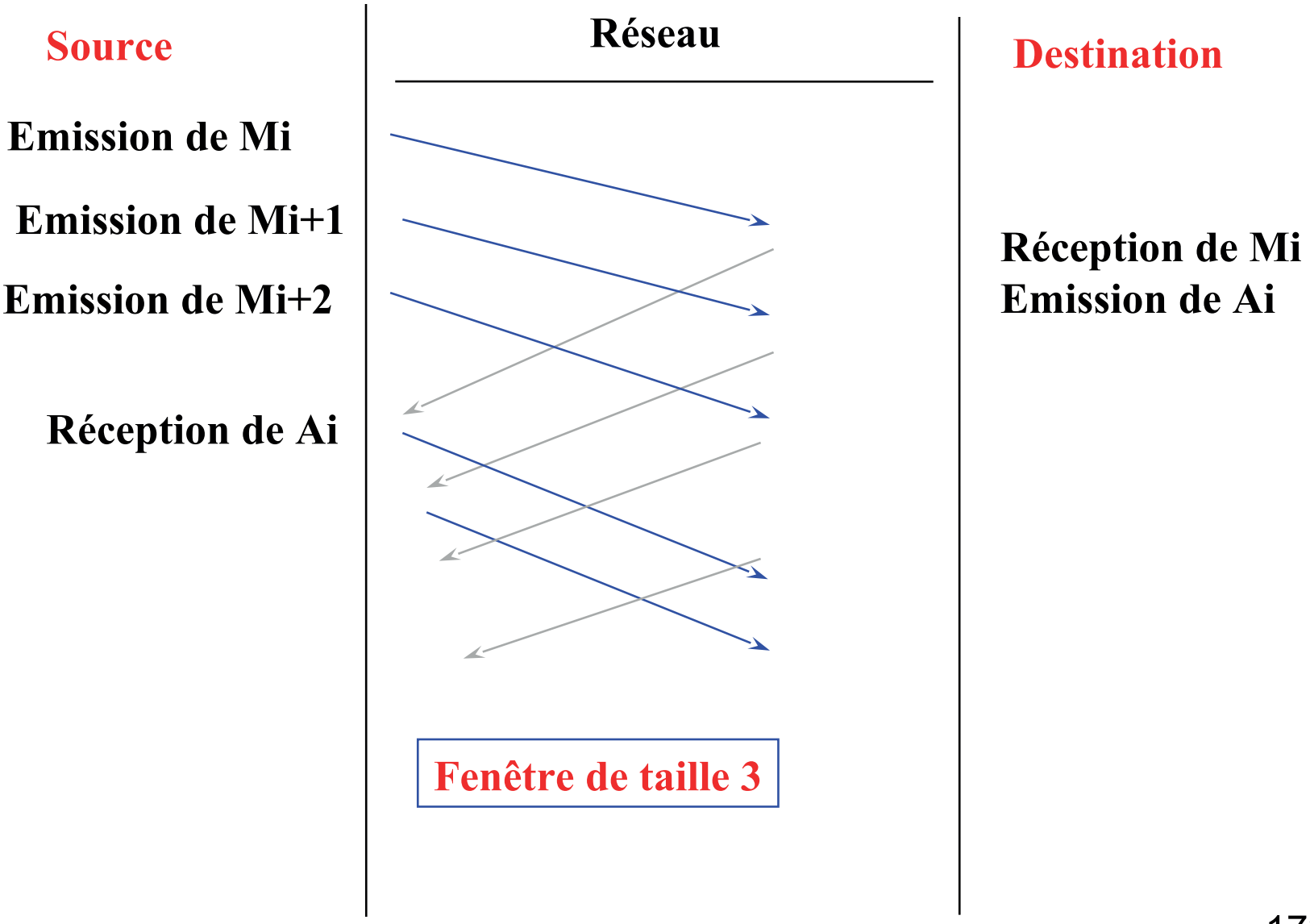
La technique acquittement simple pénalise les performances puisqu'il faut :

- **Envoyer des données**
- **Attendre**
- **Envoyer un accusé de réception**
- **Attendre**
- **...**

On peut faire mieux !

➔ **Fenêtrage : une fenêtre de taille T permet l'émission d'au plus T messages "*non acquittés*" avant de ne plus pouvoir émettre**

Gestion de la fenêtre



Gestion de la fenêtre

fenêtre glissante permettant d'optimiser la bande passante

permet également au destinataire de faire diminuer le débit de l'émetteur donc de gérer le contrôle de flux

Le mécanisme de fenêtre mis en oeuvre dans TCP opère au niveau de l'octet et non pas au niveau du segment ; il repose sur :

- **la numérotation séquentielle des octets de données,**
- **la gestion de trois pointeurs par fenêtre**

Notion de segments

UDP gère des messages

TCP gère une communication

- **Echange soutenu entre deux machines**
- **Durée importante**
- **Messages ➔ Flux de données**
- **Taille inconnue à l'avance**

  ➔ segmentation

  TCP gère des segments de 64K (ou moins)

Réception

Chaque segment est un morceau

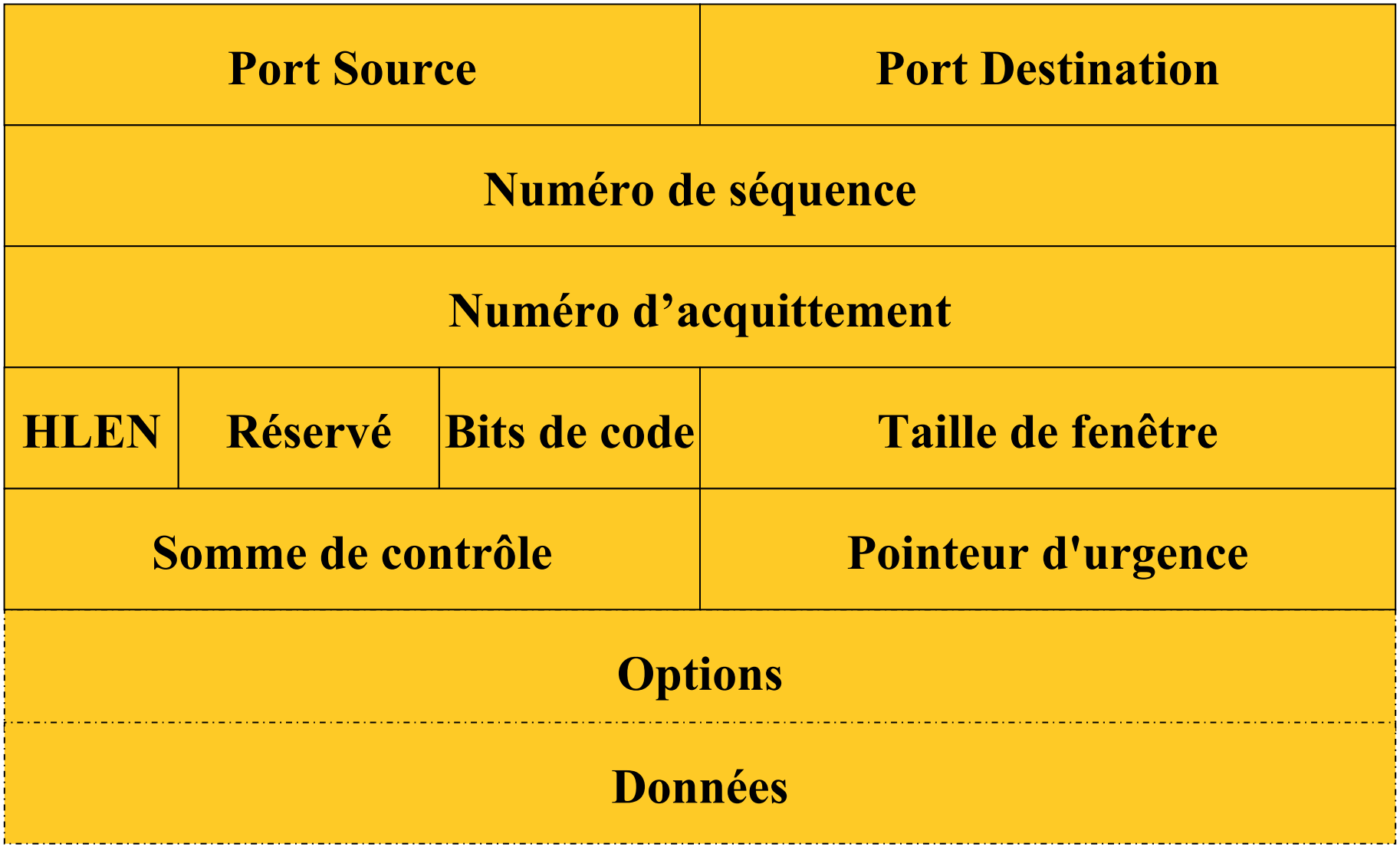
- **Ressemble à la fragmentation IP**
- **Ordre nécessaire pour recomposer le message initial**
- **IP ne garantit pas l'ordre**
  - **Chaque paquet est routé séparément**
  - **Certains routeurs équilibrent la charge des réseaux**
  - **➔ routes différentes pour paquets successifs**
- **Problème des pertes de trames**
  - **➔ trou dans la séquence (➔ retransmission)**
  - **Fenêtre glissante ➔ retard d'un segment**
  - **➔ les segments sont reçus en désordre**

Notion de segment

Segment : unité de transfert du protocole TCP

- **pour établir les connexions**
- **transférer les données et émettre des acquittements**
- **fermer les connexions**

Format du segment TCP



Le contenu du segment

Port Source	Port Destination

Port **(16bits) : entier identifiant l'application à laquelle la couche transport doit remettre les messages**

Le contenu du segment

Numéro de séquence
Numéro d'acquittement

Numéro de séquence (32bits) **: identifie la position des données par rapport au segment original**

Numéro d'accusé de réception (32bits) **: identifie la position du dernier octet reçu dans le flux entrant**

Le contenu du segment

#!'@	Réservé	Bits de code	Taille de fenêtre

HLEN : **longueur de l'en-tête**

Bits de code **: modificateur du comportement de TCP par caractérisation du segment**

Taille de fenêtre **: nombre d'octets que l'émetteur est prêt à accepter**

Le contenu du segment
Bits de code

URG : le champ "pointeur d'urgence" doit être exploité

ACK : "numéro d'accusé de réception" doit être exploité

PSH : l'émetteur indique au récepteur que toutes les données collectées doivent être transmises à l'application sans attendre les éventuelles données qui suivent.

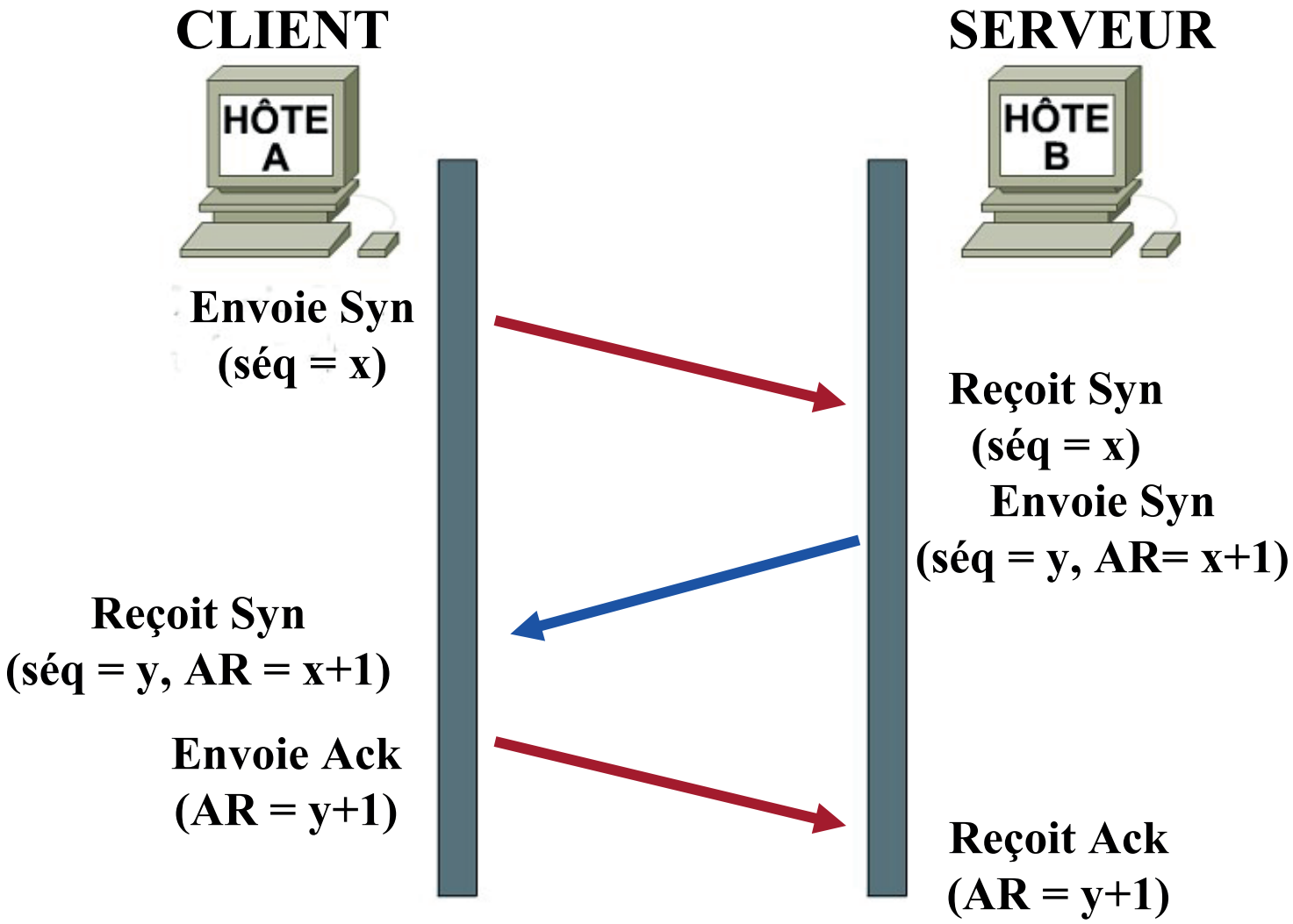
RST : Re-initialisation de la connexion

SYN : Le champ "numéro de séquence" contient la valeur de début de connexion.

FIN : L'émetteur du segment a fini d'émettre.

Début et clôture de connexion

Ouverture (1)


CLIENT
SERVEUR
HÔTE A
HÔTE B
Envoie Syn
(séq = x)
Reçoit Syn
(séq = x)
Envoie Syn
(séq = y, AR= x+1)
Reçoit Syn
(séq = y, AR = x+1)
Envoie Ack
(AR = y+1)
Reçoit Ack
(AR = y+1)

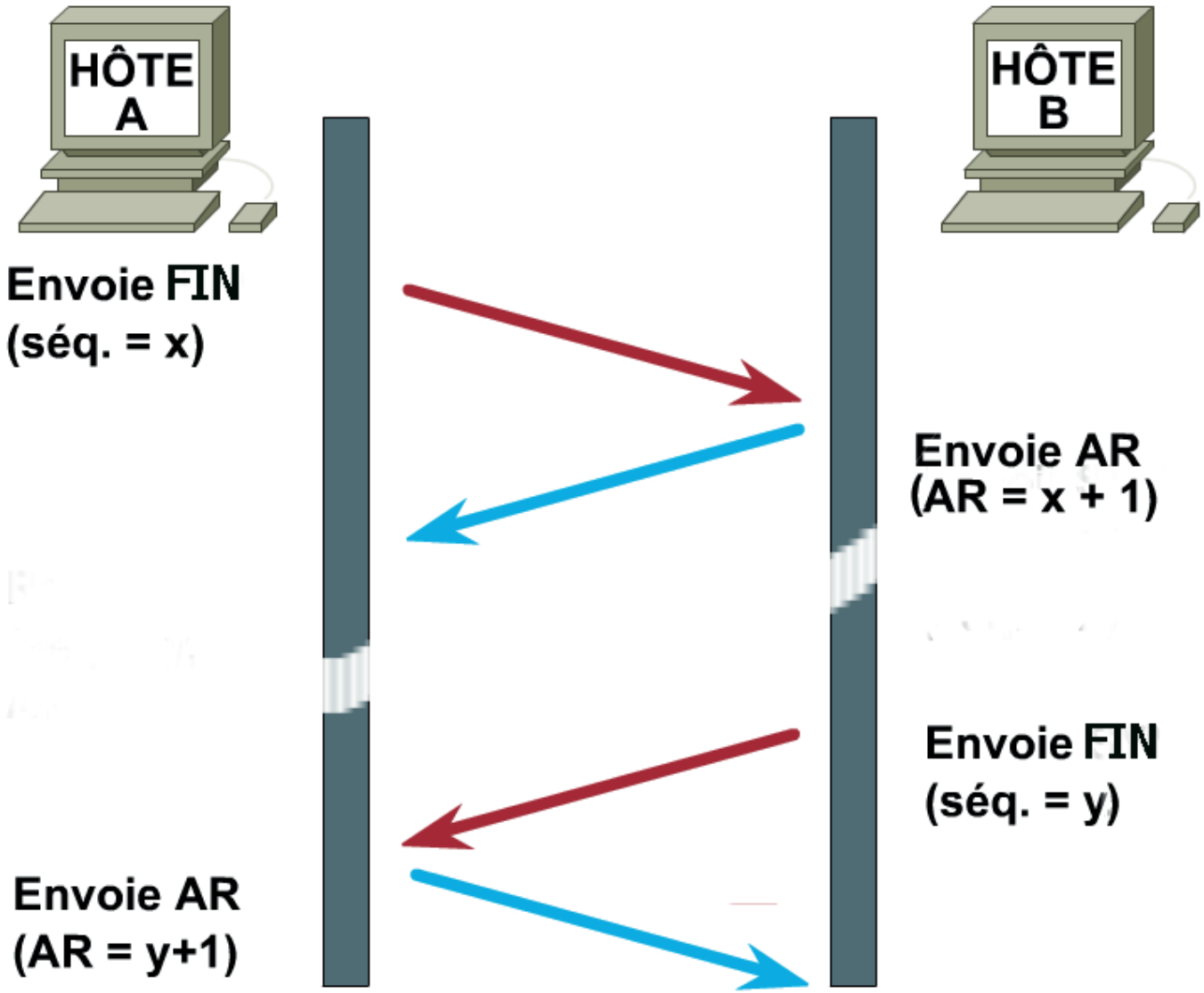
Ouverture (2)

Etablissement de la connexion en trois temps

L’émetteur (appelé client) du premier paquet est à l’origine de l’établissement du circuit. on parle "d’ouverture active"

Le récepteur (appelé serveur) du premier paquet accepte l'établissement de la connexion. On parle "d’ouverture passive"

Clôture (1)


HÔTE A
HÔTE B
Envoie FIN
(séq. = x)
Envoie AR
(AR = x + 1)
Envoie FIN
(séq. = y)
Envoie AR
(AR = y+1)

Clôture (2)

- **Etablissement de la clôture normale en quatre temps car TCP est Full Duplex**
- **Sur certains systèmes la clôture se déroule en trois temps :**
  - **Demande de fin de connexion**
  - **Acquittement et demande fin de connexion**
  - **Acquittement**
- **Possibilité de clore brutalement la connexion par l'envoi d'un segment *RST***

Mécanismes de contrôle du transport

Principes fondamentaux

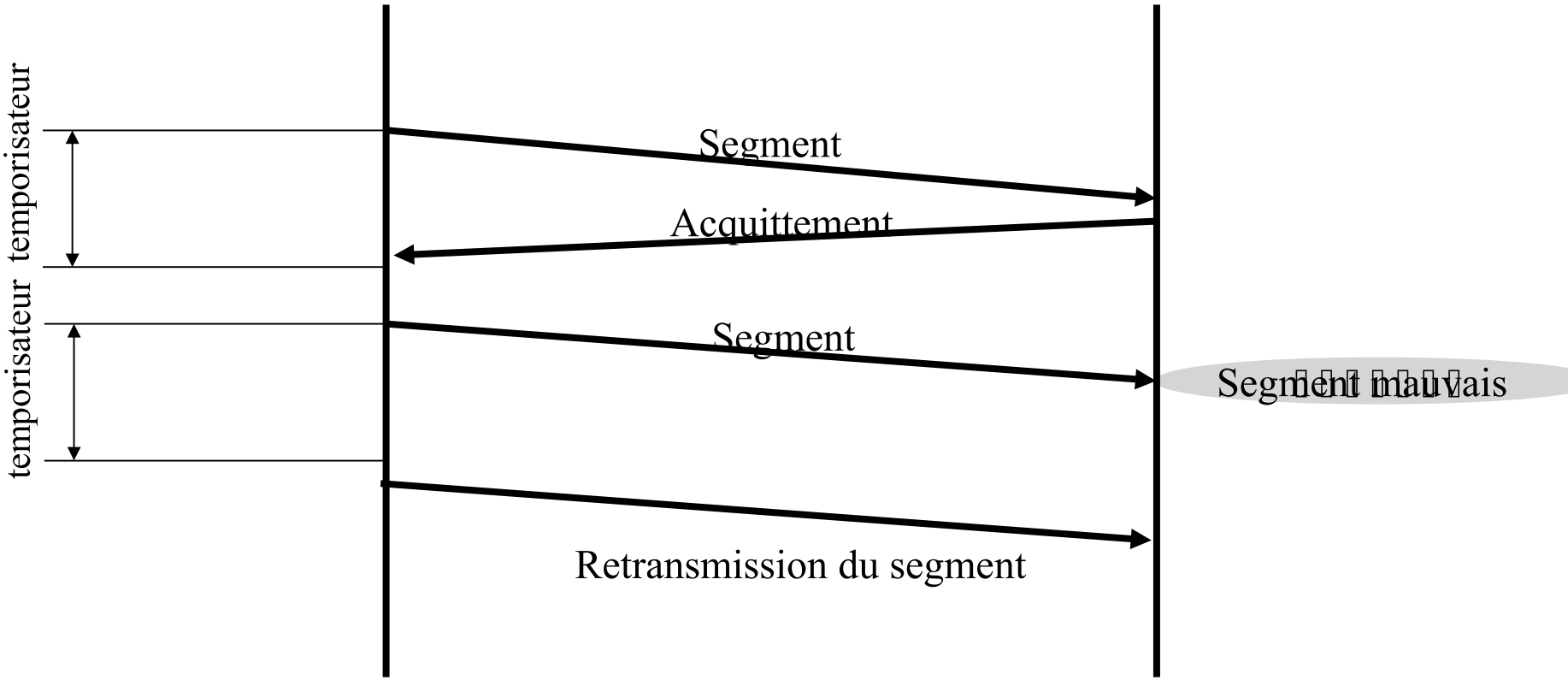
Acquittement positif : un segment bien reçu doit être acquitté ; un segment non acquitté doit être réémis au bout d'un certain temps

Numérotation des segments envoyés

Acquittement cumulatif et par anticipation

Utilisation d'une fenêtre glissante dont la taille variera au cours de l'échange

Acquittement positif

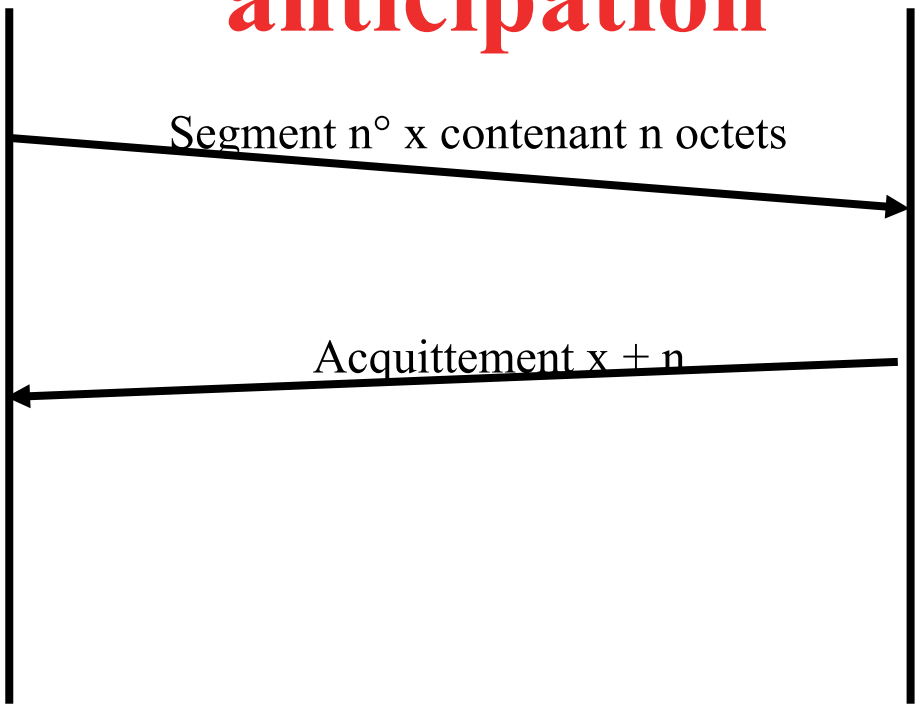


Si le segment n'est pas acquitté, le segment est considéré comme perdu et TCP le retransmet

Un réseau d'interconnexion offre des temps de transit variables nécessitant le réglage des temporisations

TCP gère des temporisations variables pour chaque connexion en utilisant un algorithme de retransmission adaptative

Acquittement cumulatif et par anticipation

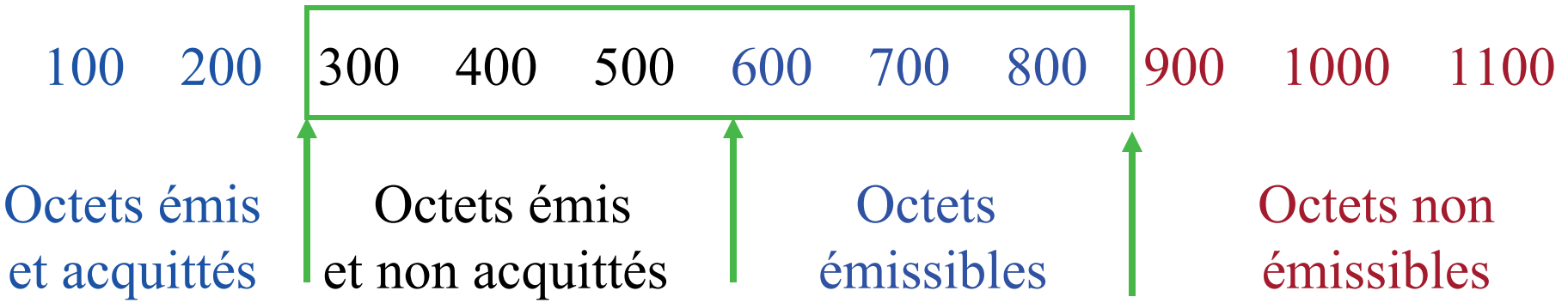


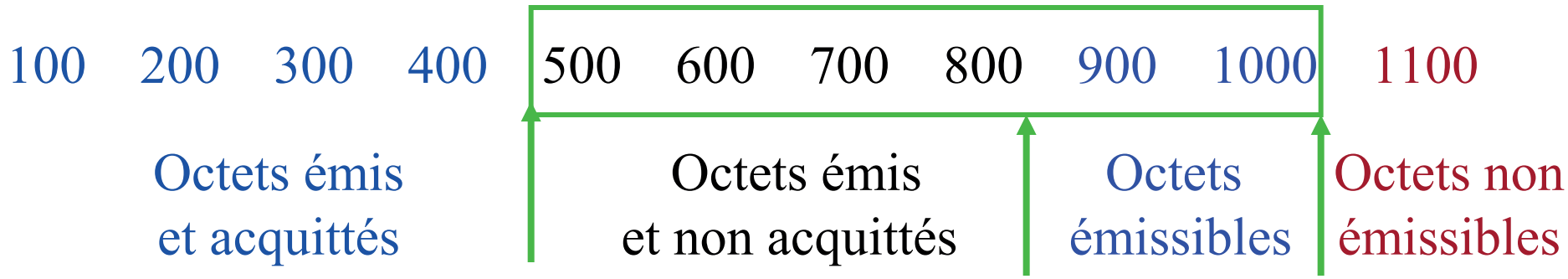
Il indique le numéro de séquence du prochain octet attendu : tous les octets précédents cumulés sont implicitement acquittés

Si un segment a un numéro de séquence supérieur au numéro de séquence attendu (bien que dans la fenêtre), le segment est conservé mais l'acquittement référence toujours le numéro de séquence attendu

Fenêtre glissante

Taille de la fenêtre 600 octets
Taille du segment émis 100 octets





Time keep alive

Cette fonction permet de détecter les « absences » : si aucune donnée ne circule, la connexion est silencieuse

permet de refermer les connexions que les utilisateurs ont laissé ouvertes

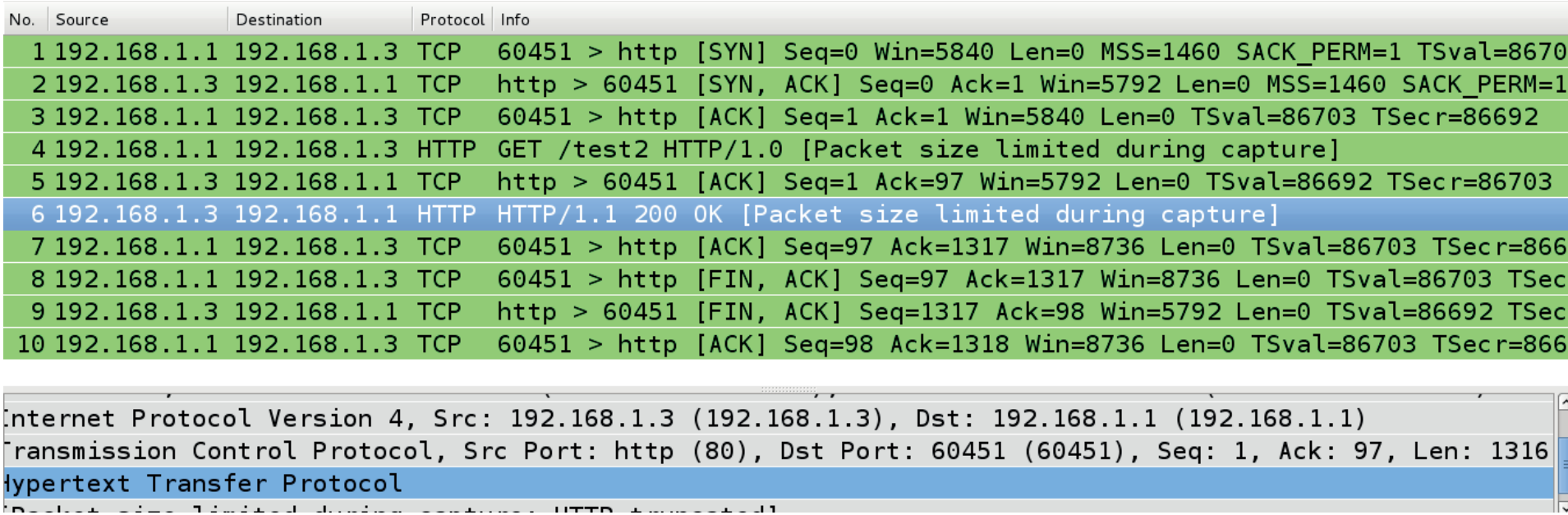
(exemple : si 9 segments « sondes » consécutifs, émis avec des intervalles de 75 secondes restent sans réponse, la connexion est fermée)

Bilan

Une connexion TCP :

- **Ouverture de connexion**
  - **Sychronisation**
  - **Acknowledge Synchronisation**
- **Envoi de trames selon fenêtre disponible**
- **Si accusé réception, décaler la fenêtre**
- **Si TimeOut, ré-envoyer le segment fautif**
- **Envoi trame de fin**
- **Accuse réception de la trame de fin**

Exemple



No.	Source	Destination	Protocol	Info
1	192.168.1.1	192.168.1.3	TCP	60451 > http [SYN] Seq=0 Win=5840 Len=0 MSS=1460 SACK_PERM=1 TSval=8670
2	192.168.1.3	192.168.1.1	TCP	http > 60451 [SYN, ACK] Seq=0 Ack=1 Win=5792 Len=0 MSS=1460 SACK_PERM=1
3	192.168.1.1	192.168.1.3	TCP	60451 > http [ACK] Seq=1 Ack=1 Win=5840 Len=0 TSval=86703 TSecr=86692
4	192.168.1.1	192.168.1.3	HTTP	GET /test2 HTTP/1.0 [Packet size limited during capture]
5	192.168.1.3	192.168.1.1	TCP	http > 60451 [ACK] Seq=1 Ack=97 Win=5792 Len=0 TSval=86692 TSecr=86703
6	192.168.1.3	192.168.1.1	HTTP	HTTP/1.1 200 OK [Packet size limited during capture]
7	192.168.1.1	192.168.1.3	TCP	60451 > http [ACK] Seq=97 Ack=1317 Win=8736 Len=0 TSval=86703 TSecr=866
8	192.168.1.1	192.168.1.3	TCP	60451 > http [FIN, ACK] Seq=97 Ack=1317 Win=8736 Len=0 TSval=86703 TSec
9	192.168.1.3	192.168.1.1	TCP	http > 60451 [FIN, ACK] Seq=1317 Ack=98 Win=5792 Len=0 TSval=86692 TSec
10	192.168.1.1	192.168.1.3	TCP	60451 > http [ACK] Seq=98 Ack=1318 Win=8736 Len=0 TSval=86703 TSecr=866

- **1-2-3 : connexion**
- **4 : requête http (demande d'un fichier)**
- **5 : acquittement**
- **6 : envoie de du fichier**
- **7 : acquittement**
- **8-9-10 : déconnexion (en trois temps)**

TCP : ports standards

No port	Mot-clé	Description
20	FTP-DATA	File Transfer [Default Data]
21	FTP	File Transfer [Control]
23	TELNET	Telnet
25	SMTP	Simple Mail Transfer
37	TIME	Time
42	NAMESERVER	Host Name Server
43	NICNAME	Who Is
53	DOMAIN	Domain Name Server
79	FINGER	Finger
80	HTTP	;;;
110	POP3	Post Office Protocol - Version 3
111	SUNRPC	SUN Remote Procedure Call

Serveur DHCP
Dynamic Host Configuration Protocol (Protocole de configuration dynamique des hôtes)

Introduction

Une adresse réseau peut être configurer soit de manière statique ou dynamique.

- **Statique : vous déterminez vous-même l'adresse IP de la machine.**
- **dynamique : obtention de l'adresse grâce à un serveur DHCP.**

Remarque : ne pas confondre statique et fixe.

Serveur dhcp3-server

Installation :

#apt-get install dhcp3-server

Configuration :

Deux cas seront traités :

- **adresse fixe alloué à la machine web-smi.**
- **adresses dynamiques alloués aux autres machines.**

Cas traité

On suppose que le serveur dispose de trois interfaces réseaux.

- **eth0 : interface pour se connecter à Internet ; adresse obtenue par dhcp.**
- **eth1 et eth2 : interfaces connectés à deux réseaux différents.**
- **eth1 @IP = 192.168.1.1**
- **eth2 @IP = 192.168.10.1**

Interface(s) d'écoute(s)

Si vous voulez que le serveur écoute sur certaines interfaces vous devez les spécifier dans /etc/default/dhcp3-server à :

INTERFACES="eth1 eth2"

L'écoute se fera sur les interfaces eth1 et eth2.

Configuration du serveur

La configuration se fait dans le fichier :

/etc/dhcp3/dhcpd.conf

Les options sont définies de façon globale ou par réseau.

Dans ce qui suit, nous allons voir un exemple de configuration pour le cas traité.

Options générales

 # options communes aux différents réseaux.

#Nom du domaine DNS

option domain-name "ump.ma";

#Nom(s) de(s) serveur(s) DNS

option domain-name-servers 192.168.100.10, 192.168.10.11;

Options générales

#Temps de renouvèlement des adresses

#par défaut en s (1 h)

default-lease-time 3600;

 # maximum (2 h)

max-lease-time 7200;

Options générales

 # Mode autoritaire (autoritaire)

#Est-ce-que ce serveur DHCP est le serveur principal?

authoritative;

Masque de sous-réseau

option subnet-mask 255.255.255.0;

```
# déclaration du sous réseau 192.168.1.*
subnet 192.168.1.0 netmask 255.255.255.0 {
  # Si vous voulez spécifier un domaine
différent de celui par défaut :
  #option domain-name "fso.ump.ma";
  ## Adresse de diffusion
  option broadcast-address 192.168.1.255;
  # routeur par défaut
  option routers 192.168.1.1;
  range 192.168.1.2  192.168.1.100;
}
```

```
# déclaration sous réseau 192.168.10.*
subnet 192.168.10.0 netmask 255.255.255.0 {
  # Si vous voulez spécifier un domaine
différent de celui par défaut :
  #option domain-name "fso.ump.ma";
  ## Adresse de diffusion
  option broadcast-address 192.168.10.255;
  # routeur par défaut
  option routers 192.168.10.1;
  range 192.168.10.2  192.168.10.200;
}
```

hôte « web-smi »

```
host web-smi {
  # adresse mac de la carte réseau !
  # A remplacer par celle de la machine
  hardware ethernet 08:00:27:A6:C2:50;
  # adresse attribué
  fixed-address 192.168.1.200;
}
```

Serveurs HTTP

Le protocole HTTP

- **Le protocole HTTP (HyperText Transfer Protocol) est le protocole de transport de données le plus utilisé sur Internet depuis 1990. La version 0.9 était uniquement destinée à transférer des données sur Internet (en particulier des pages Web écrites en HTML. La version 1.0 du protocole (la plus utilisée) permet désormais de transférer des messages avec des en-têtes décrivant le contenu du message.**
- **Le but du protocole HTTP est de permettre un transfert de fichiers (essentiellement au format HTML) localisés grâce à une chaîne de caractères appelée URL entre un navigateur (le client) et un serveur Web (appelé httpd sur les systèmes de type UNIX).**

Pourquoi utiliser HTTP ?

- **HTTP est devenu le protocole de communication de l'Internet.**
- **HTTP est disponible sur toutes les plates-formes.**
- **HTTP est un protocole simple, qui ne requière que peu de support pour fonctionner correctement.**
- **HTTP: Peu de paquets sont nécessaires pour échanger des informations**
- **HTTP offre un niveau de sécurité simple et efficace.**
- **HTTP est le seul protocole utilisable à travers des pare-feu.**

Fonctionnement de HTTP

- **Fonctionnement**
  - **Connexion du client vers le serveur**
  - **Demande d'une information via une méthode**
  - **Renvoi de l' information, erreur ou information sur le document**
  - **Déconnexion**
- **Code de retour**
  - **1xx :Information**
  - **2xx : Succès**
  - **3xx : Redirection**
  - **4xx : Erreurs (p.ex. 404)**
  - **5xx : Erreurs venant du serveur HTTP**
- **Méthodes**
  - **GET : Requête de la ressource située à l'URL spécifiée**
  - **HEAD : Requête de l'en-tête de la ressource située à l'URL spécifiée**
  - **POST : Envoi de données au programme situé à l'URL spécifiée**
  - **PUT : Envoi de données à l'URL spécifiée**
  - **DELETE: Suppression de la ressource située à l'URL spécifiée**

Le protocole HTTP
Requête HTTP

Une requête HTTP est un ensemble de lignes envoyé au serveur par le navigateur. Elle comprend :

- **<u>Une ligne de requête:</u> c'est une ligne précisant le type de document demandé, la méthode qui doit être appliquée, et la version du protocole utilisée. La ligne comprend trois éléments devant être séparés par un espace :**
  - **La méthode**
  - **L'URL**
  - **La version du protocole utilisé par le client (généralement *HTTP/1.0*)**
- **<u>Les champs d'en-tête de la requête:</u> il s'agit d'un ensemble de lignes facultatives permettant de donner des informations supplémentaires sur la requête et/ou le client (Navigateur, système d'exploitation, ...). Chacune de ces lignes est composée d'un nom qualifiant le type d'en-tête, suivi de deux points (:) et de la valeur de l'en-tête**
- **<u>Le corps de la requête:</u> c'est un ensemble de lignes optionnelles devant être séparées des lignes précédentes par une ligne vide et permettant par exemple un envoi de données par une commande POST lors de l'envoi de données au serveur par un formulaire.**

Le protocole HTTP
Réponse HTTP

Une réponse HTTP est un ensemble de lignes envoyées au navigateur par le serveur. Elle comprend :

- **<u>Une ligne de statut:</u> c'est une ligne précisant la version du protocole utilisé et l'état du traitement de la requête à l'aide d'un code et d'un texte explicatif. La ligne comprend trois éléments devant être séparés par un espace :**
  - **La version du protocole utilisé**
  - **Le code de statut**
  - **La signification du code**
- **<u>Les champs d'en-tête de la réponse:</u> il s'agit d'un ensemble de lignes facultatives permettant de donner des informations supplémentaires sur la réponse et/ou le serveur. Chacune de ces lignes est composée d'un nom qualifiant le type d'en-tête, suivi de deux points (:) et de la valeur de l'en-tête**
- **<u>Le corps de la réponse:</u> il contient le document demandé**

Serveurs Web

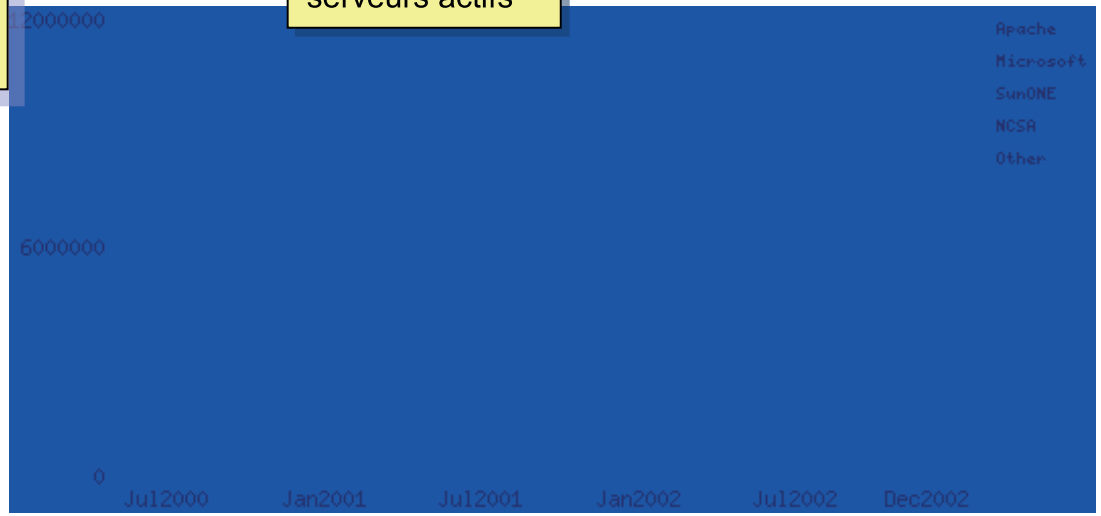
Serveurs Web

Principaux serveurs :

- **Apache**
- **Microsoft : Internet Information Server (IIS)**

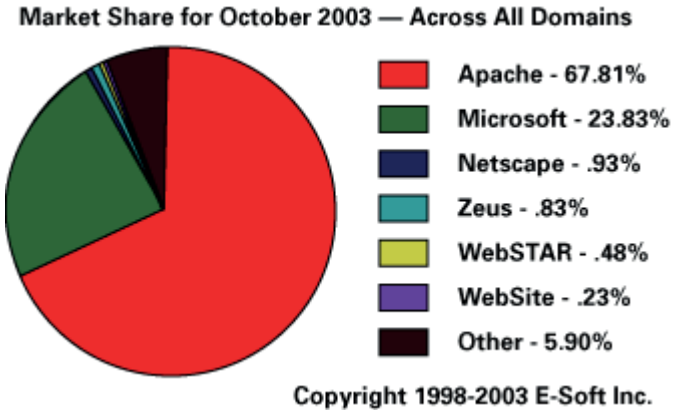
Serveurs Web en ligne 2003

serveurs actifs



Parts de marchés 2003:

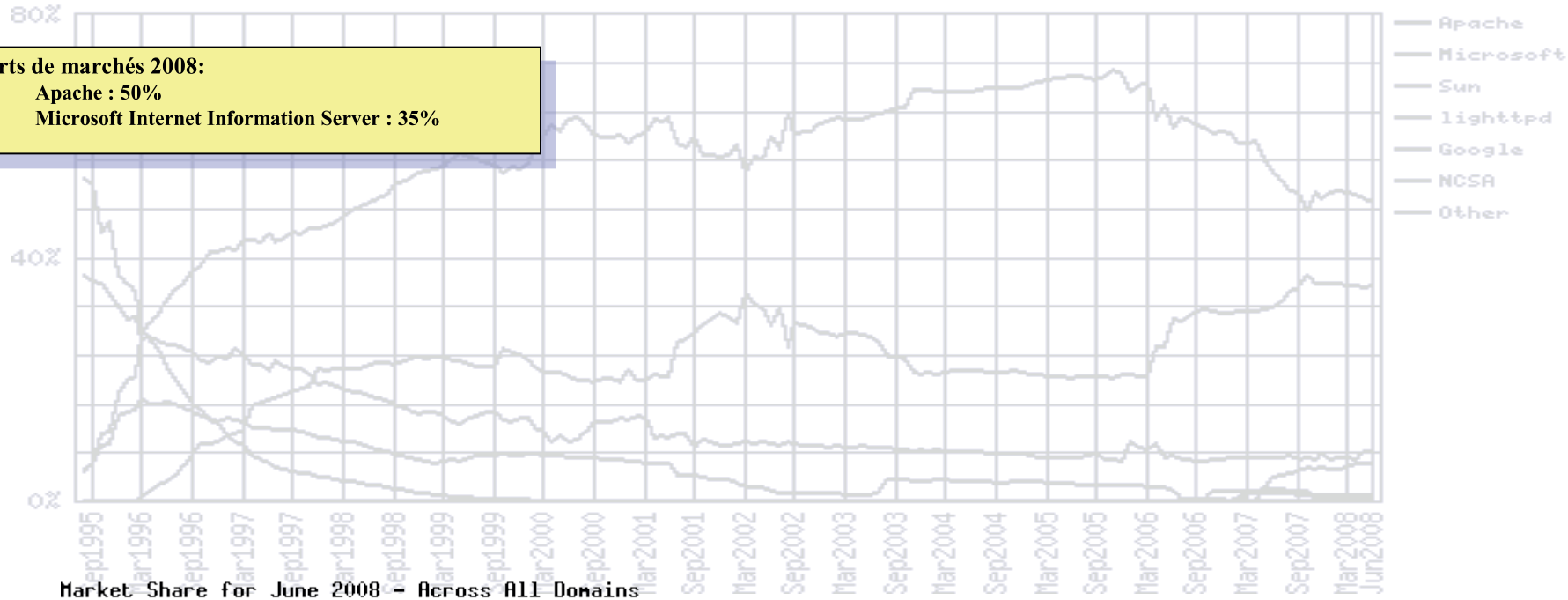
- **Apache : 63%**
- **Microsoft Internet Information Server : 23%**



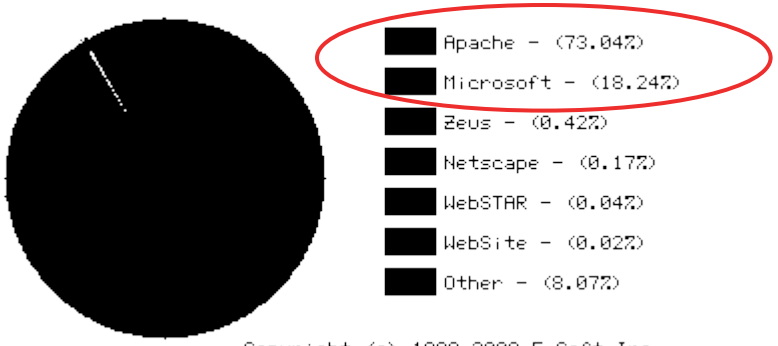
(source http://survey.netcraft.com)

Serveurs Web en ligne 2008

Parts de marchés 2008:
- **Apache : 50%**
- **Microsoft Internet Information Server : 35%**



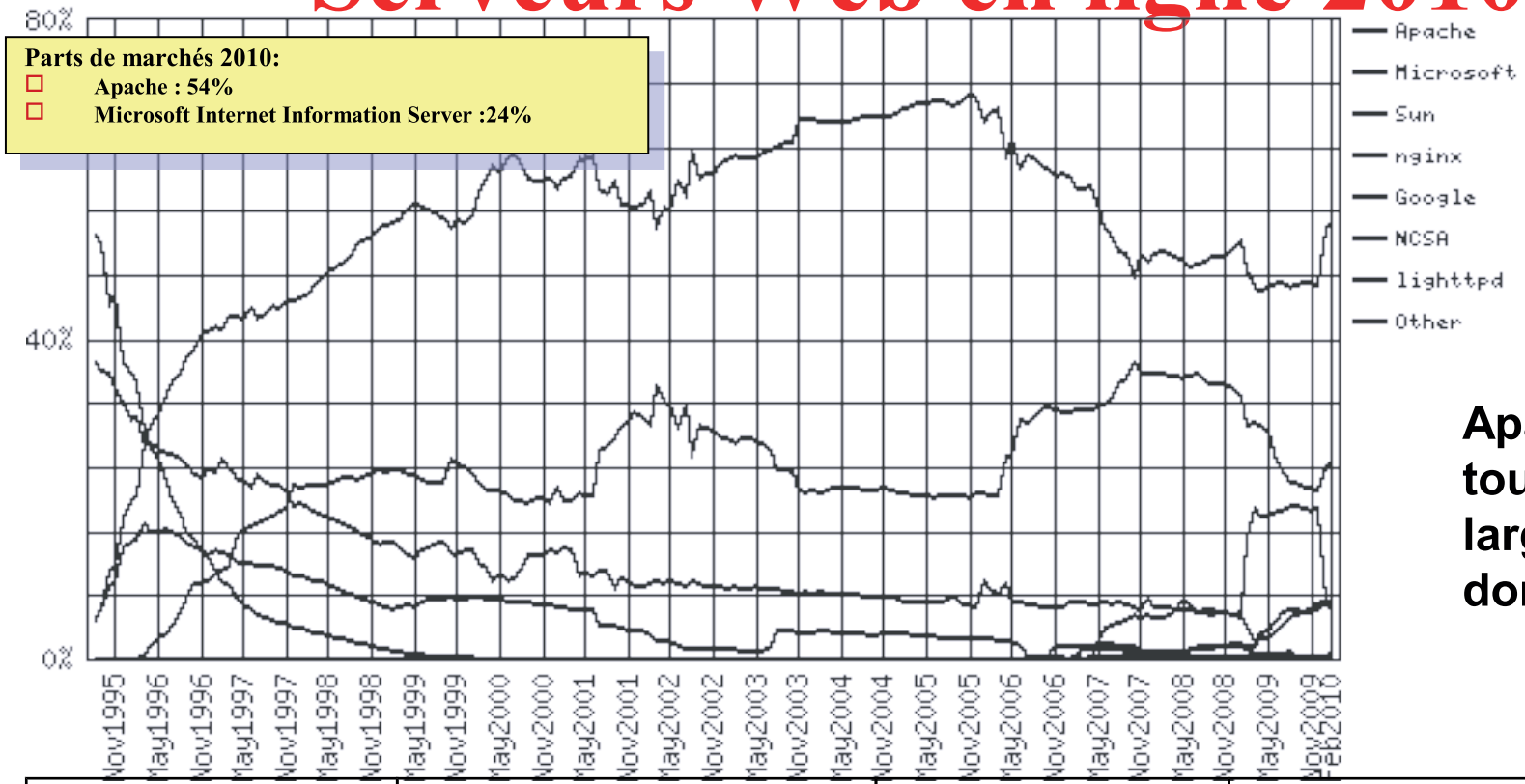
Market Share for June 2008 - Across All Domains



(source http://survey.netcraft.com)

Serveurs Web en ligne 2010

Parts de marchés 2010:
- **Apache : 54%**
- **Microsoft Internet Information Server :24%**

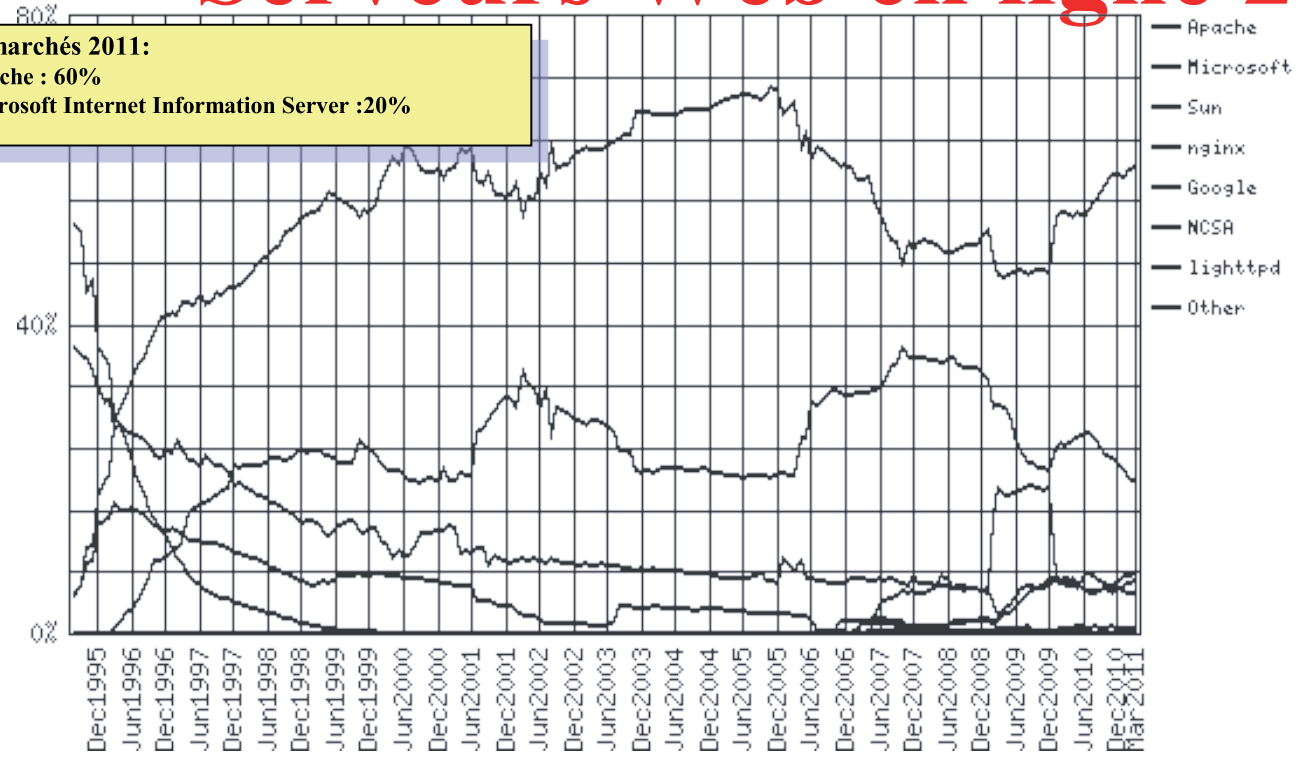


Apache est toujours largement dominant…

Développeur	Janvier 2010	Pourcentage	février 2010	Pourcentage
Apache	111,307,941	**53.84%**	112,903,926	**54.46%**
Microsoft	49,792,844	24.08%	50,928,226	24.57%
Google	14,550,011	7.04%	14,315,464	6.91%
nginx	15,568,224	7.53%	13,978,719	6.74%
lighttpd	955,146	0.46%	1,097,685	0.53%

Serveurs Web en ligne 2011

Parts de marchés 2011:
- **Apache : 60%**
- **Microsoft Internet Information Server :20%**



Apache est toujours largement dominant…

Développeur	01/02/11	Pourcentage	01/03/11	Pourcentage
Apache	171,195,554	60.10%	179,720,332	60.31%
Microsoft	57,084,126	20.04%	57,644,692	19.34%
nginx	21,570,463	7.57%	7.57%1	7.65%
Google	14,454,484	5.07%	15,161,530	5.09%
lighttpd	1,953,966	0.69%	1,796,471	0.60%

Apache

Généralités

- **Apache est le serveur web le plus populaire sur Internet. Il est robuste et extensible. Il est maintenu est développé par la fondation ASF (Apache Software Foundation)**
- **Existe depuis 1995, à l'origine basé sur le serveur NCSA httpd 1.3 (Université de l'Illinois Etats-Unis)**
- **Site officiel : http://www.apache.org**
- **Distribué sous une licence "Open source" (Licence Apache).**

Pourquoi Apache?

- **Gratuit.**
- **Code source disponible et modifiable permet un développement rapide du serveur.**
- **Très grande réactivité dans la correction de tout bogue identifié.**
- **Très grande flexibilité du serveur grâce à sa structure modulaire l'ajout d'un nouveau module permet d'ajouter de nouvelles fonctionnalités.**
- **Disponibles sur plusieurs plateformes (Linux, windows, …)**

Pré-requis systèmes

- **Apache s'exécute sur n'importe quel type de machine.**
- **Pour un serveur de sites WEB peu exigeant, un simple processeur fera l'affaire.**
- **Pour des sites très exigeants utilisant de nombreuses bases de données, un multiprocesseur doit être envisagé.**
- **Concernant la mémoire : plus il y a de mémoire vive et plus la quantité de donnée en mémoire est importante ce qui permet d'accélérer les accès.**

Pré-requis systèmes

- **Concernant le disque dur : un disque dur rapide permet d'améliorer les performances d'accès aux données des sites WEB.**
- **Pour les sites à grande audience, il est préférable d'utiliser plusieurs disques de tailles moyennes plutôt qu'un seul disque à grande capacité (un disque dur ne pouvant lire qu'à un seul endroit à la fois!).**

Installation et configuration sous un système de type Debian

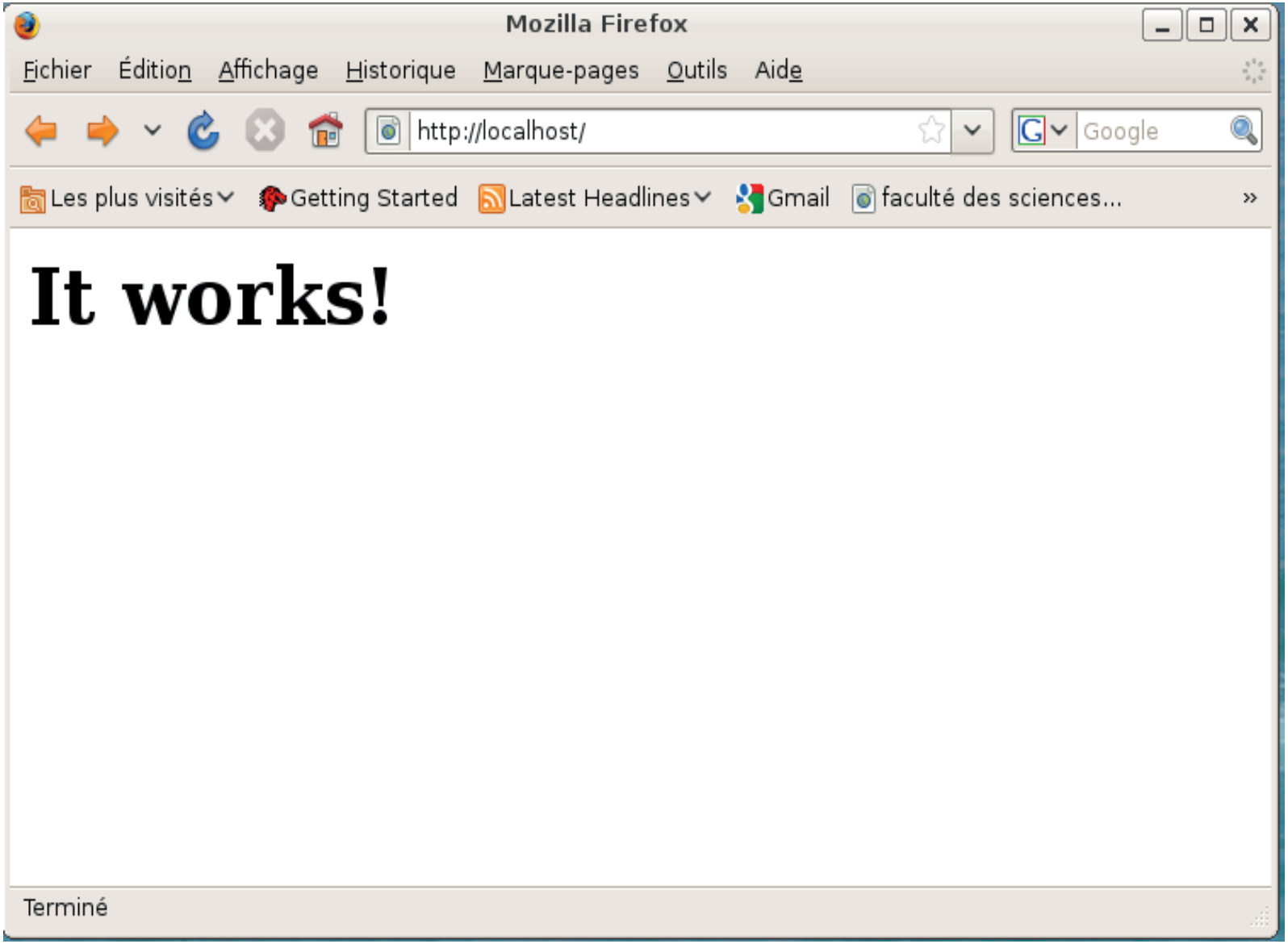
Installation minimale

Installation en mode console :

#apt-get install apache2

Ou utilisation d'un gestionnaire graphique (synaptic).

Vérification de l'installation



Fichiers et répertoires d'apache dans /etc/apache2

apache2.conf : fichier de configuration principale.

httpd.conf : fichier vide.

envars : contient les variables d'environnement.

conf.d/ : contient des fichiers de configuration additionnels. Il sont utilisés dans apache2.conf, par la ligne :

Include /etc/apache2/conf.d

ports.conf : directives de configuration pour

Fichiers et répertoires d'apache dans /etc/apache2

mods-available/ : contient une série de fichiers .load et .conf. Un fichier .load contient les paramètres de configuration nécessaires pour charger un module in question. Le fichier .conf correspondant, es paramètres de configuration nécessaires pour utiliser le module en question.

mods-enabled/ : pour utilisé un module (autorisé), il faut mettre un lien symbolique vers le fichier .load (et .conf, s'il existe) du module associé dans le dossier mods-available

Fichiers et répertoires d'apache dans /etc/apache2

sites-available/ : même chose que mods-available/, mais cette fois pour les sites virtuels.

Ce n'est obligé d'avoir le même nom pour le site et le fichier.

sites-enabled/ : même chose que mods-enabled/.

Par défaut, un seul serveur est disponible (le serveur par défaut). Il est disponible dans apache2.conf par la ligne :

Outils

Les commandes a2enmod et a2dismod sont disponibles pour permettre ou ne pas permettre un module.

ha2enmod userdir

pour permettre aux utilisateurs d'avoir leurs propres pages web disponibles via un lien de type :

http://NomSite/~utilisateur

http://localhost/~smi6

Exemple d'une page personnelle

Outils

Les commandes a2ensite et a2dissite sont disponibles pour permettre ou ne pas permettre un site.

Configuration de base

Avant de commencer la configuration, il faut faire une sauvegarde des fichiers que vous voulez modifier :

#cp apache2.conf apache2.conf.save

Racine du serveur :

ServerRoot "/etc/apache2"

Adresse électronique de l'administrateur du serveur :

ServerAdmin webmaster@localhost

Configuration de base

Port à écouter (ports.conf):

Listen 80

Listen 192.168.100.1:80

Emplacement par défaut des pages html :

DocumentRoot /var/www/

Pages par défaut (mods-enabled/dir.conf):

DirectoryIndex index.htm index.html index.php

Configuration de base

Journal d'erreur par défaut :

ErrorLog /var/log/apache2/error.log

PHP

- **PHP est un Langage de script interprété (non compilé) spécialement conçu pour le développement d'applications web. Il peut être intégré facilement au HTML (voir php.net).**
- **Pour installer la version 5 de PHP, il faut exécuter la commande :**

  # apt-get install php5 libapache2-mod-php5
- **Il faut redémarrer apache :**

  # /etc/init.d/apache2 restart

Vérification de l'installation

Dans /var/www, créer le script info.php, contenant les lignes :

```
<?php
   phpinfo();
FZ
```

Dans votre navigateur, tapez l'adresse http://localhost/info.php

il fournira un ensemble d'informations et de paramètres de configuration.

MySQL

MySQL est un système de gestion de bases de données relationnel (SGBDR) libre, open-source et gratuit. Il est performant et très populaire. Il est multi-utilisateurs.

Installation :

apt-get install mysql-server

Utilisation

Si vous n'avez pas défini de mot de passe (déconseillé) :

mysql -u root

Si vous avez défini un mot de passe :

mysql -u root -p

et tapez votre mot de passe.

Vous arriverez alors sur un prompt du type :

mysql>

Vous pouvez alors taper des requêtes MySQL. N'oubliez pas le point-virgule à la fin de la requête.

Ajouter ou changer le mot de passe de root

Pour affecter un mot de passe à l'utilisateur root, tapez la commande suivante :

 # mysqladmin -u root password MotPasse

Pour changer le mot de passe de l'utilisateur root, tapez la commande :

 # mysqladmin -u root -p password MotPasse

Enter password: (ancien mot passe)

Sélectionner la base de données

Soit directement lorsque vous lancez le client mysql en ligne de commande :

mysql -u root -p votre_base

Soit une fois connecté à mysql en tapant :

mysql> use votre_base;

Quelques requêtes SQL

Créer une base de données :

```
CREATE DATABASE nom_de_la_base;
```

Liste des bases de données :

```
SHOW DATABASES;
```

Liste des tables de la base de données active

```
SHOW TABLES;
```

Structure d'une table :

```
DESCRIBE nom_table;
```

Renommer une table :

```
ALTER TABLE nom_table RENAME AS
nouveau_nom;
```

Quelques requêtes SQL

Créer un un utilisateur

```
CREATE user
"nom_utilisateur"@"localhost";
```

Supprimer un utilisateur

```
DROP user nom_utilisateur@localhost;
```

Donner tous les droits sur une base de données à un utilisateur précis

```
GRANT ALL ON nom_base.*  TO
nom_utilisateur@localhost;
```

Supprimer les droits sur une base de données à un utilisateur précis

```
REVOKE ALL privileges ON nom  base.*
```

PhpMyAdmin

phpMyAdmin est une application écrite en PHP, pour l'administration de MySQL. Elle est accessible via un navigateur.

Avant l'installation, il faut s'assurer, que php et MySQL sont déjà installés.

Installation :

apt-get install phpmyadmin

Utilisation :

dans votre navigateur, saisir l'adresse

http://localhost/phpmyadmin