Couche Transport

Le Protocole ICMP (1)

- Le protocole ICMP (Internet Control Message Protocol) permet d'envoyer des messages de contrôle ou d'erreur vers d'autres machines ou passerelles.

- Beaucoup d'erreurs sont causées par l'émetteur, mais d'autres sont dûes à des problèmes d'interconnexions rencontrées sur l'Internet :

  - Machine destination déconnectée,
  - Durée de vie du datagramme expirée,
  - Congestion de passerelles intermédiaires.

Le Protocole ICMP(2)

- Si une passerelle détecte un problème sur un datagramme IP, elle le détruit et émet un message ICMP pour informer l'émetteur initial.

- Les messages ICMP sont véhiculés à l'intérieur de datagrammes IP et sont routés comme n'importe quel datagramme IP sur l'internet.

- Une erreur engendrée par un message ICMP ne peut donner naissance à un autre message ICMP (évite l'effet cummulatif).

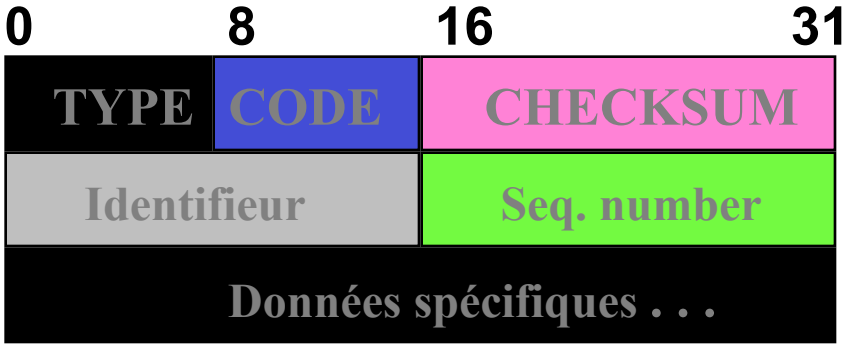
ICMP : format des messages

TYPE	8 bits; type de message
CODE	8 bits; informations complémentaires
CHECKSUM	16 bits; champ de contrôle
HEAD-DATA	en-tête datagramme + 64 premiers bits des données.

TYPE	Message ICMP
0	Echo Reply
3	Destination Unreachable
4	Source Quench
5	Redirect (change a route)
8	Echo Request
11	Time Exceeded (TTL)
12	Parameter Problem with a Datagram

TYPE	Message ICMP
13	Timestamp Request
14	Timestamp Reply
15	Information Request (obsolete)
16	Information Reply (obsolète)
17	Address Mask Request
18	Address Mask Reply

ICMP : format des commandes



IDENTIFIER et SEQUENCE NUMBER sont utilisés par l'émetteur pour contrôler les réponses aux requêtes, (CODE = 0).

Demande d'écho et réponse d'écho (*Echo Request, Echo Reply*)

- Permettent à une machine ou passerelle de déterminer la validité d'un chemin sur le réseau.
- Le champ de données spécifiques est composé de données optionnelles de longueur variable émises par la requête d'écho et devant être renvoyées par le destinataire si présentes.
- Utilisé par les outils applicatifs tels que: ping.

ICMP : les commandes

Synchronisation des Horloges et temps de transit

- Les horloges de deux machines qui diffèrent de manière importante peuvent poser des problèmes pour des logiciels distribués.

- Une machine peut émettre une demande d'horodatage (*timestamp request*) à une autre machine susceptible de lui répondre (*timestamp reply*) en donnant l'heure d'arrivée de la demande et l'heure de départ de la réponse.

- L'émetteur peut alors estimer le temps de transit ainsi que la différence entre les horloges locale et distante.

- Le champ de données spécifiques comprend l'heure originale (*originate timestamp*) émis par le demandeur, l'heure de réception (*receive timestamp*) du destinataire, et l'heure de départ (*transmit timestamp*) de la réponse.

ICMP : les commandes

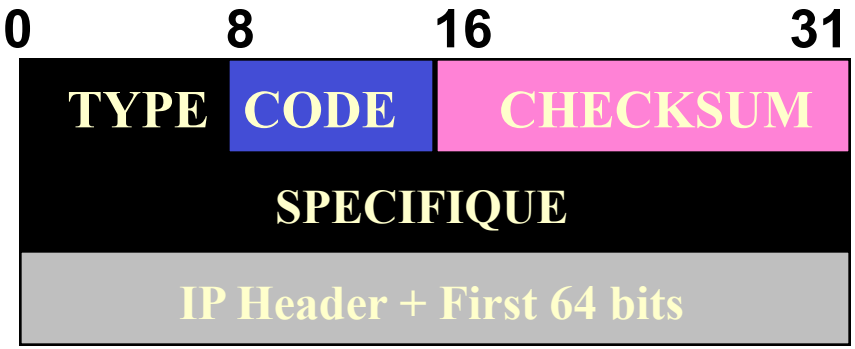
Demande et réponse d'information (*Information Request + Reply*)

- Ces messages étaient initialement utilisés pour permettre aux machines de connaître leur adresse IP au démarrage du système.
- Ces commandes sont aujourd'hui remplacées par les protocoles RARP et BOOTP.

Obtention de masque de sous-réseau

- Une machine peut émettre une demande de masque de sous-réseau (*Subnet Mask Request*) vers une passerelle gérant le sous-réseau en question.
- La passerelle transmet par une "*Subnet Mask Reply*", l'adresse de masque de sous-réseau dans le champ de donnée spécifique.

ICMP : les messages d'erreur



Format des messages d'erreur ICMP

- **CODE** indique le codage de l'erreur rapportée et est spécifique à chaque type d'erreur,
- **SPECIFIQUE** est un champ de données spécifique au type d'erreur,
- IP **HEADER + FIRST 64 bits** contient l'en-tête IP + les premiers 64 bits de données du datagramme pour lequel le message est émis.

ICMP : les messages d'erreur

- Lorsqu'une passerelle émet un message ICMP de type destination inaccessible, le champ code décrit la nature de l'erreur :
  - 0 Network Unreachable
  - 1 Host Unreachable
  - 2 Protocol Unreachable
  - 3 Port Unreachable
  - 4 Fragmentation Needed
  - 5 Source Route Failed
  - 6 Destination Network Unknown
  - 7 Destination Host Unknown
  - 8 Source Host Isolated
  - 9 Communication with desination network administratively prohibited
  - 10 Communication with desination host administratively prohibited
  - 11 Network Unreachable for type of Service
  - 12 Host Unreachable for type of Service

ICMP : contrôle de congestion

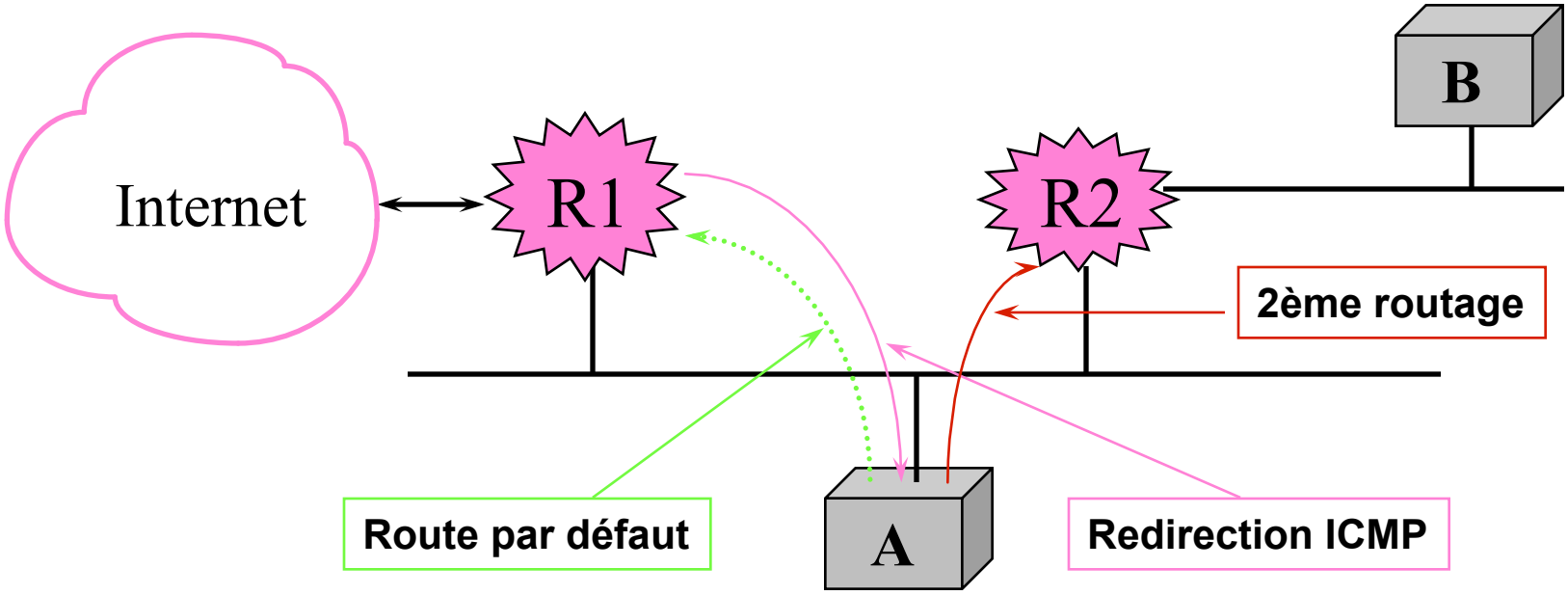
- Une situation de congestion se produit :
  - lorsqu'une passerelle est connectée à deux réseaux aux débits différents (elle ne peut écouler au rythme imposé par le réseau le plus rapide),
  - lorsque de nombreuses machines émettent simultanément des datagrammes vers une passerelle.

- Pour palier ce problème, la machine peut émettre un message ICMP de limitation de débit de la source (***Source Quench***) vers l'émetteur.

- Il n'existe pas de message d'annulation de limitation de débit. La source diminue le débit, puis l'augmente progressivement tant qu'elle ne reçoit pas de nouvelle demande de limitation.

ICMP : modification de route

Un message ICMP de redirection de route peut être transmis par une passerelle vers une machine reliée au même réseau pour lui signaler que la route n'est pas optimale.



Une fois la redirection effectuée, les datagrammes seront acheminés vers la passerelle appropriée.

ICMP : modification de route

- Dans le bloc de commande, le champ SPECIFIQUE indique l'adresse de la passerelle que la machine doit utiliser pour router le datagramme;

CODE spécifie la redirection :

CODE	SIGNIFICATION
0	Redirect datagrams for the Network
1	Redirect datagrams for the Host
2	Redirect datagrams for the Type of Service and Network
3	Redirect datagrams for the Type of Service and Host

ICMP : autres compte-rendus

- Lorsqu'une passerelle ou une machine détecte un problème avec un datagramme (en-tête incorrecte) non couvert par les messages ICMP prédéfinis, elle émet un message "*Parameter Problem on a Datagram*" vers l'émetteur du datagramme.
- Le problème rencontré consiste soit en une option manquante (dans le datagramme), soit en une donnée erronée.
- Dans le bloc de commande, le champ CODE indique la nature du pb:

CODE	SIGNIFICATION
0	erreonous data
1	missing option

- Le champ spécifique comprend un pointeur (codé sur les 8 premiers bits, les 24 restants étant à 0) servant à identifier l'octet erroné dans le datagramme; il est non significatif lorsque CODE = 1.

Couche Transport

- **TCP : Transmission Control Protocol**
- **UDP : User Datagram Protocol**

UDP : User Datagram Protocol

- **UDP** : protocole de transport sans connexion de service applicatif :
  - émission de messages applicatifs : sans établissement de connexion au préalable
  - l'arrivée des messages ainsi que l'ordonnancement ne sont pas garantis.
- **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 s'adresser au processus exécuté sur la machine réceptrice.

UDP : les Ports

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

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

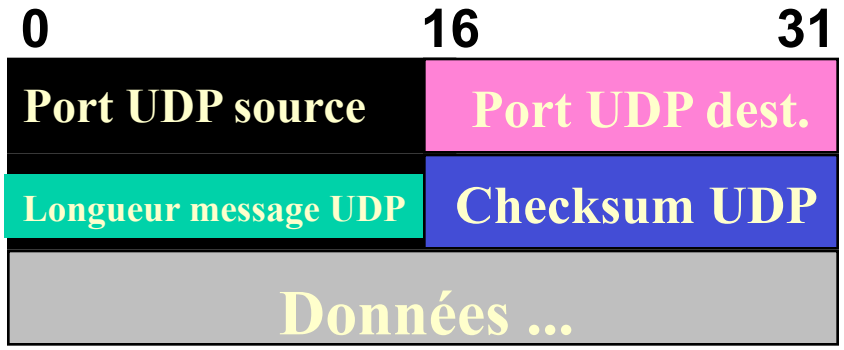
- Les processus disposent d'une interface système leur permettant de spécifier un port ou d'y accéder (socket, ...).

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

UDP : Format des messages

Les messages UDP sont également appelés des datagrammes UDP.

Ils contiennent deux parties : un en-tête UDP et les données UDP.



Format des messages UDP

- Les ports source et destination contiennent les numéros de port utilisés par UDP pour démultiplexer les datagrammes destinés aux processus en attente de les recevoir.

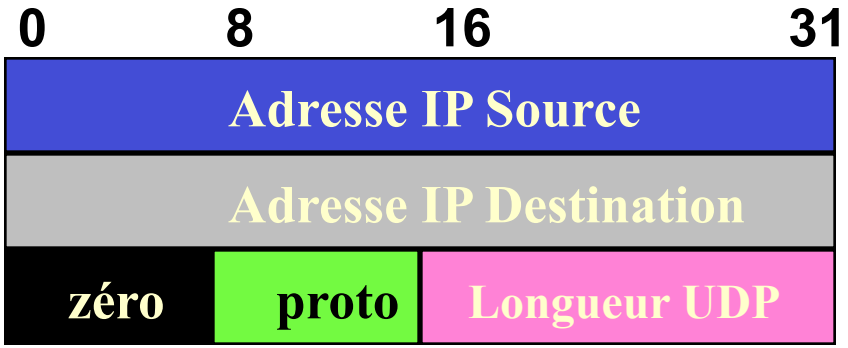
- Le port source est facultatif (égal à zéro si non utilisé).

- La longueur du message est exprimée en octets (8 au minimum) ,

- Le champ de contrôle est optionnel (0 si non utilisé).

UDP : pseudo en-tête

- Lorsqu'il est utilisé, le champ de contrôle couvre plus d'informations que celles contenue dans le datagramme UDP;
- En effet, le checksum est calculé avec un pseudo-en-tête non transmis dans le datagramme:



Format du pseudo en-tête

•Le champ PROTO indique l'identificateur de protocole pour IP (17= UDP)

•Le champ LONGUEUR UDP spécifie la longueur du datagramme UDP sans le pseudo-en-tête.

UDP : Multiplexage

- UDP multiplexe et démultiplexe les datagrammes en sélectionnant les numéros de ports :
  - une application obtient un numéro de port de la machine locale; dès que l'application émet un message via ce port, le champ PORT SOURCE du datagramme UDP contient ce numéro de port,
  - une application connaît un numéro de port distant afin de communiquer avec le service désiré.
- Lorsque UDP reçoit un datagramme, il vérifie que celui-ci est un des ports actuellement actifs (associé à une application) et le délivre à l'application responsable
- Si ce n'est pas le cas, il émet un message ICMP *port unreachable*, et détruit le datagramme.

UDP : les ports standards

- **Certains ports sont réservés (*well-kown port assignements*) :**

No port	Mot-clé	Description
7	ECHO	Echo
11	USERS Active	Users
13	DAYTIME	Daytime
37	TIME	Time
42	NAMESERVER	Host Name Server
53	DOMAIN	Domain Name Server
67	BOOTPS	Boot protocol server
68	BOOTPC	Boot protocol client
69	TFTP	Trivial File transfert protocol
123	NTP	Network Time Protocol
161	SNMP	Simple Network Management prot.

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

TCP : Transmission Control Protocol

- Transport fiable de la technologie TCP/IP.
  - Transferts tamponés : découpage en segments
  - Connexions bidirectionnelles et simultanées
- Service en mode connecté
- Garantie de non perte de messages

TCP : La connexion

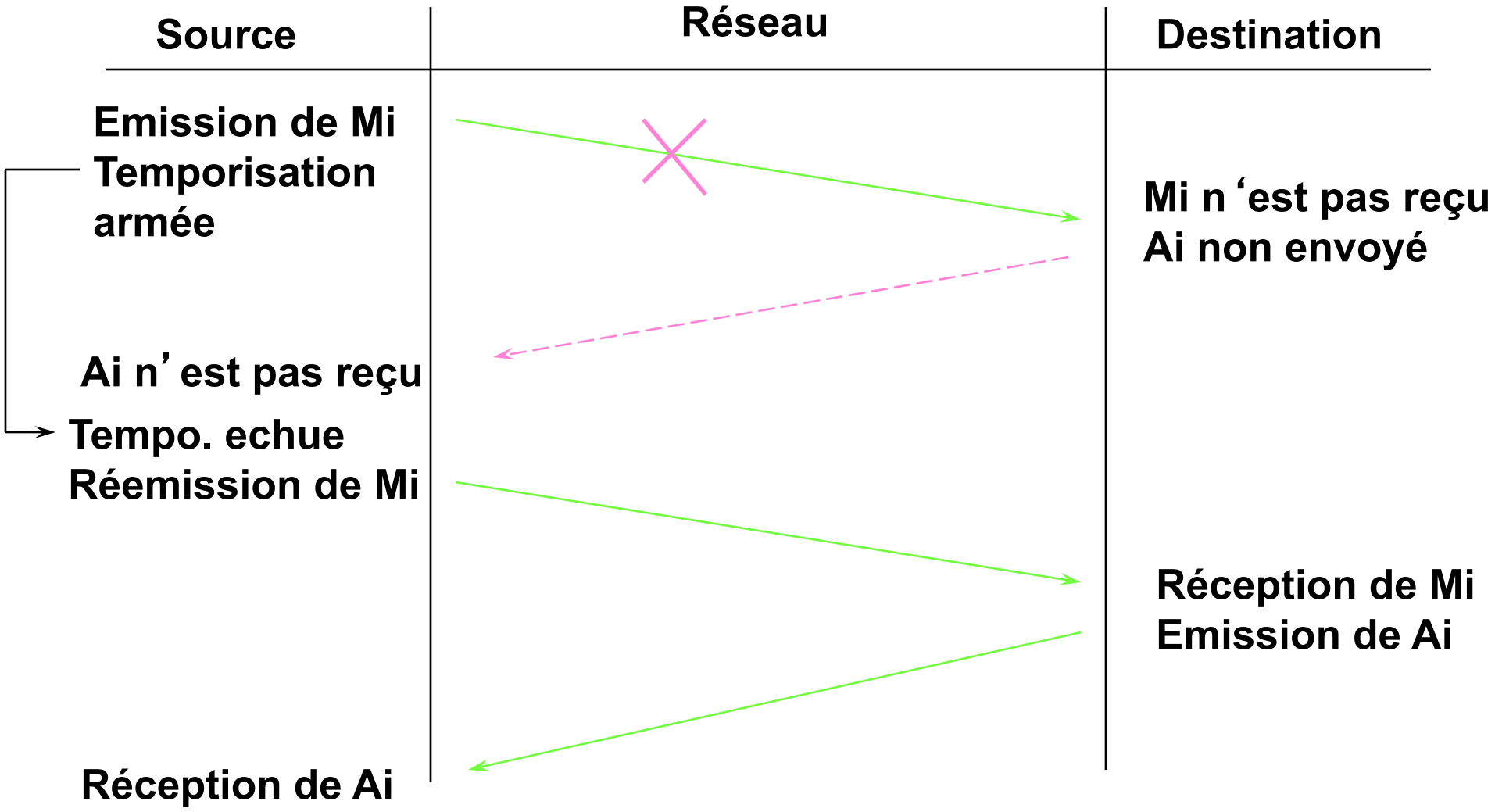
- une connexion de type circuit virtuel est établie avant que les données ne soient échangées : appel + négociation + transferts
- Une connexion = une paire d'extrémités de connexion
- Une extrémité de connexion = couple (adresse IP, port)
- **Exemple** de connexion : ((124.32.12.1, 1034), (19.24.67.2, 21))
- Une extrémité de connexion peut être partagée par plusieurs autres extrémités de connexions
- 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.

TCP : Segmentation

- **Segmentation, contrôle de flux**
  - Les données transmises à TCP constituent un flot d'octets de longueur variable.
  - TCP divise ce flot de données en segments en utilisant un mécanisme de fenêtrage.
  - Un segment est émis dans un datagramme IP.

- **Acquittement de messages**
  - Contrairement à UDP, TCP garantit l'arrivée des messages, c'est à dire qu'en cas de perte, les deux extrémités sont prévenues.
  - Ce concept repose sur les techniques d'acquittement de message : lorsqu'une source S émet un message Mi vers une destination D, S attend un acquittement Ai de D avant d'émettre le message suivant Mi+1.
  - Si l'acquittement Ai ne parvient pas à S, S considère au bout d'un certain temps que le message est perdu et reémet Mi :

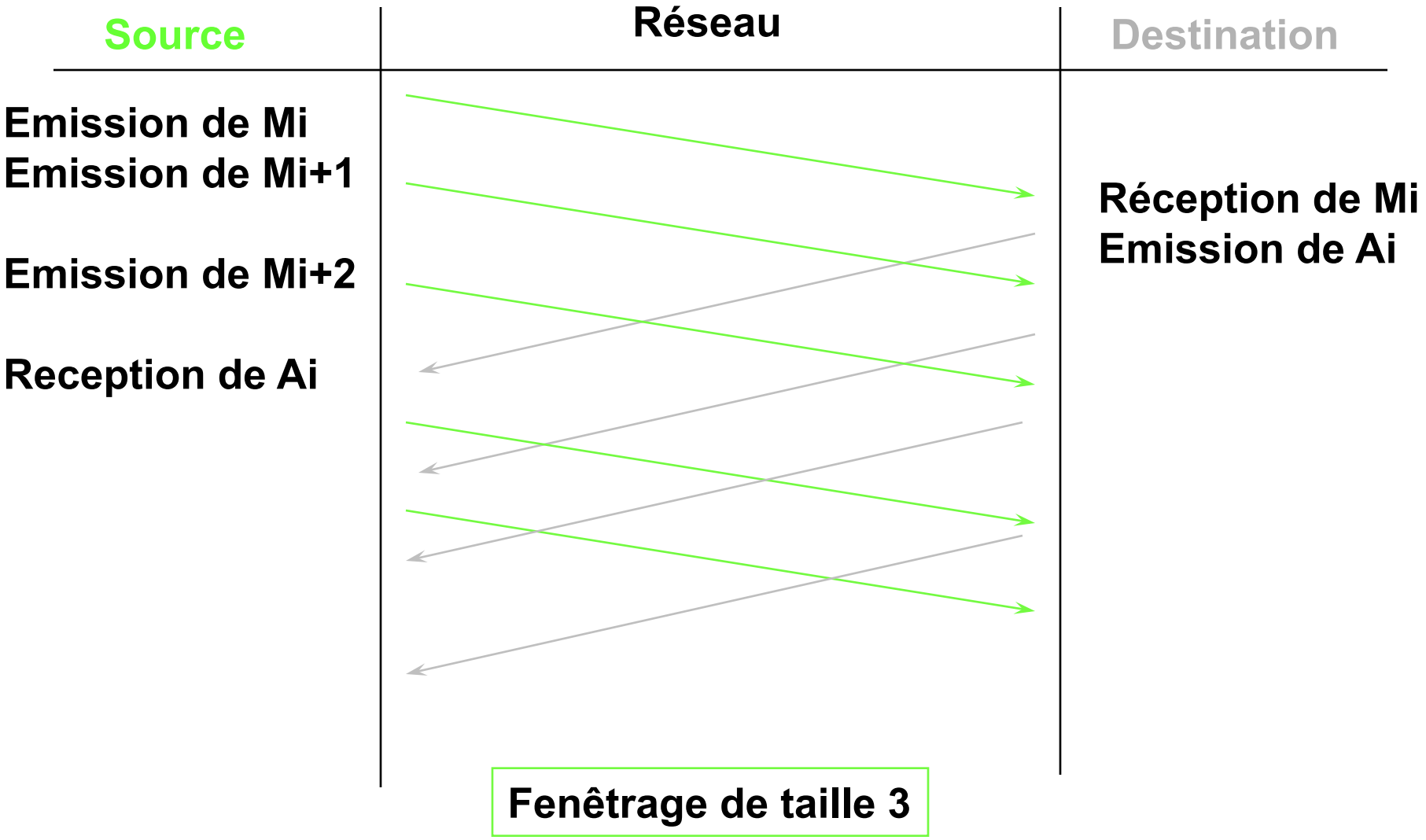
TCP : Acquittements



TCP : le fenêtrage

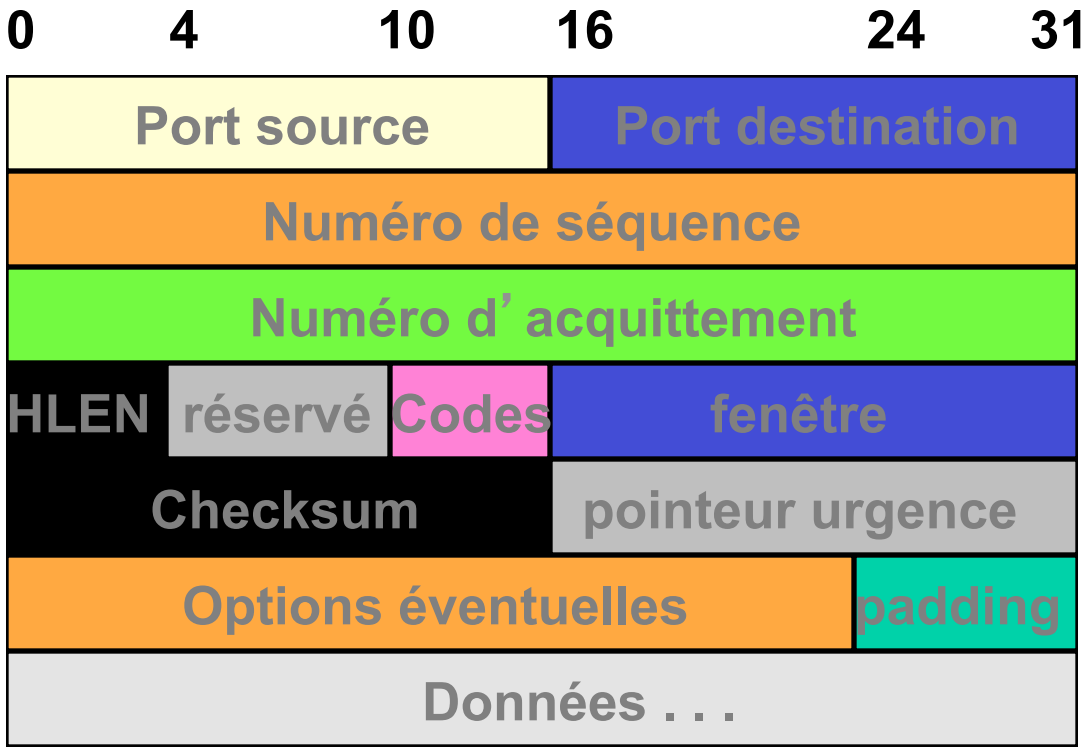
- La technique acquittement simple pénalise les performances puisqu'il faut attendre un acquittement avant d'émettre un nouveau message.
- Le fenêtrage améliore le rendement des réseaux.
- La technique du fenêtrage : une fenêtre de taille T, permet l'émission d'au plus T messages "*non acquittés*" avant de ne plus pouvoir émettre

TCP : le fenêtrage



TCP : format du segment

- **Segment** : unité de transfert du protocole TCP.
  - Echangés pour établir les connexions,
  - Transférer les données,
  - Emettre des acquittements,
  - Fermer les connexions;



TCP : format du segment

- Numéro de séquence : le numéro de séquence du premier octet (NSP) de ce segment. Généralement à la suite d'octets O1, O2, ..., On (données du message) est associée la suite de numéro de séquence NSP, NSP+1, ..., NSP +n.

- Numéro d'acquittement : le prochain numéro de séquence NS attendu par l'émetteur de cet acquittement. Acquitte implicitement les octets NS-1, NS-2, etc.

- Fenêtre: la quantité de données que l'émetteur de ce segment est capable de recevoir; ceci est mentionné dans chaque segment (données ou acquittement).

TCP : format du segment

- CODE BITS : indique la nature du segment :
  - URG : le pointeur de données urgentes est valide, les données sont émises sans délai, les données reçues sont remises sans délai.
  - SYN : utilisé à l'initialisation de la connexion pour indiquer où la numérotation séquentielle commence. Syn occupe lui-même un numéro de séquence bien que ne figurant pas dans le champ de données.
  - FIN : utilisé lors de la libération de la connexion;

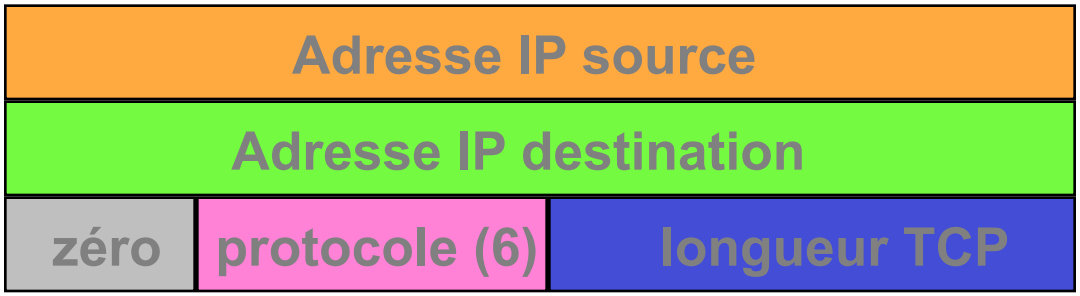
TCP : format du segment

- PSH : fonction push. Normalement, en émission, TCP reçoit les données depuis l'applicatif , les transforme en segments à sa guise puis transfère les segments sur le réseau; Un récepteur TCP décodant le bit PSH, transmet à l'application réceptrice les données correspondantes sans attendre plus de données de l'émetteur.

- RST : utilisé par une extrémité pour indiquer à l'autre extrémité qu'elle doit réinitialiser la connexion. Ceci est utilisé lorsque les extrémités sont désynchronisées.

TCP : format du segment

- **CHECKSUM** : calcul du champ de contrôle : utilise un pseudo-en-tête et s'applique à la totalité du segment obtenu (PROTO =6) :



TCP : format du segment

OPTIONS

- Permet de négocier la taille maximale des segments échangés. Cette option n'est présente que dans les segments d'initialisation de connexion (avec bit SYN).
- TCP calcule une taille maximale de segment de manière à ce que le datagramme IP résultant corresponde au MTU du réseau. La recommandation est de 536 octets.
- La taille optimale du segment correspond au cas où le datagramme IP n'est pas fragmenté mais :
  - il n'existe pas de mécanisme pour connaître le MTU,
  - le routage peut entraîner des variations de MTU,
  - la taille optimale dépend de la taille des en-têtes (options).

TCP : acquittements

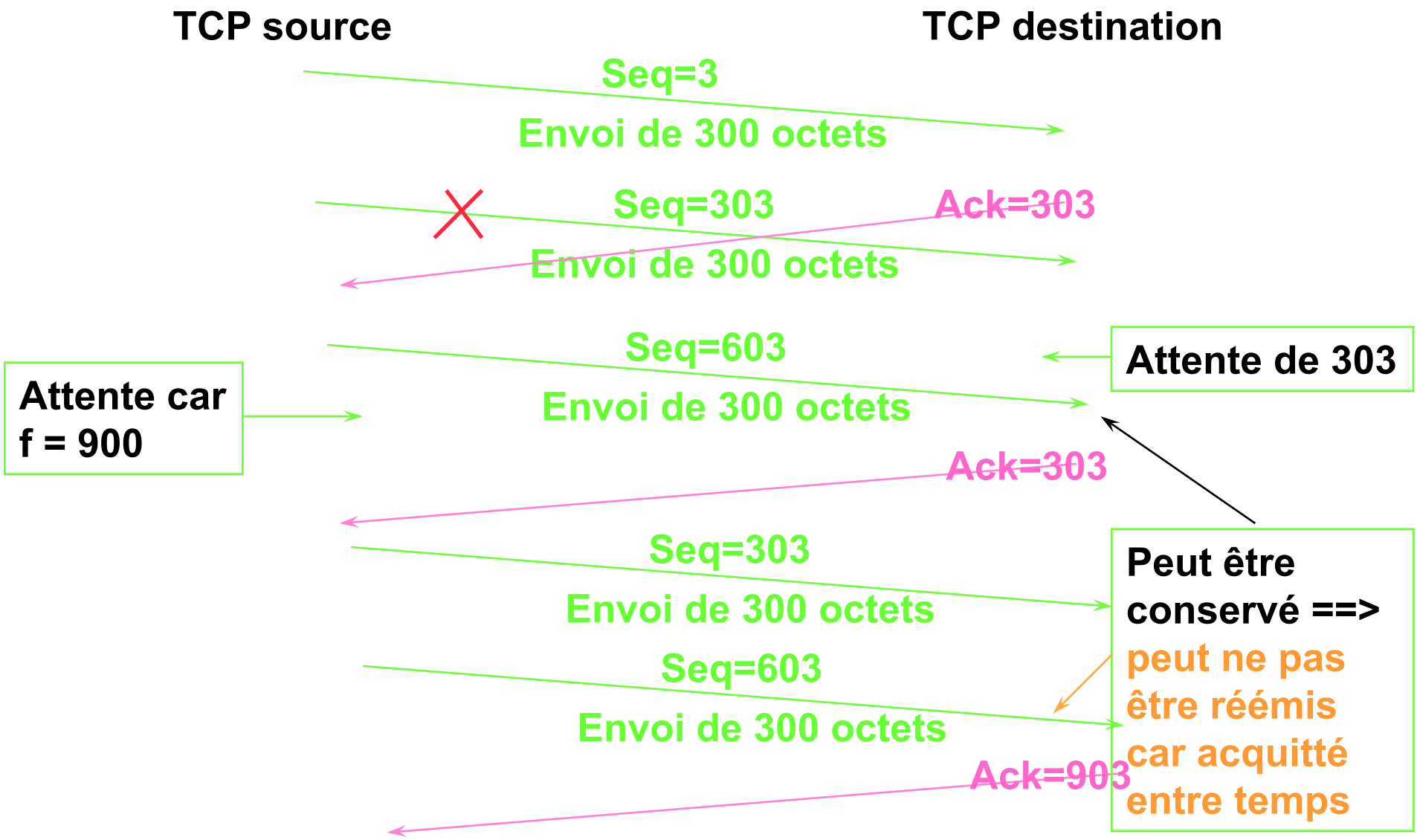
Acquittements et retransmissions

- Le mécanisme d'acquittement de TCP est cumulatif :
  - 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.
- Pour tout segment émis, TCP s'attend à recevoir un acquittement:
  - Si le segment n'est pas acquitté, le segment est considéré comme perdu et TCP le retransmet.
  - Or 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

TCP : acquittements

Fenêtre=900

Segment=300



TCP : retransmissions

algorithme de retransmission adaptative

- enregistre la date d'émission d'un segment,
- enregistre la date de réception de l'acquittement correspondant,
- calcule l'échantillon de temps de boucle A/R écoulé,
- détermine le temps de boucle moyen RTT (Round Trip Time) :

  RTT = (a * anc_RTT) + ((1-a) * NOU_RTT))

  avec 0<= a < 1

  a proche de 1 : RTT insensible aux variations brèves,

  a proche de 0 : RTT très sensible aux variations rapides,
- calcule la valeur du temporisateur en fonction de RTT.
- Les premières implémentations de TCP ont choisi un coefficient constant B pour déterminer cette valeur : Temporisation = B * RTT avec B >1 (généralement B=2).
- Aujourd'hui de nouvelles techniques sont appliquées pour affiner la mesure du RTT : l'algorithme de Karn.

TCP : retransmissions

L'algorithme de Karn repose sur les constatations suivantes :

- en cas de retransmission d'un segment, l'émetteur ne peut savoir si l'acquittement s'adresse au segment initial ou retransmis (ambiguïté des acquittements), =>l'échantillon RTT ne peut donc être calculé correctement,
- TCP ne doit pas mettre à jour le RTT pour les segments retransmis.

- L'algorithme de Karn combine les retransmissions avec l'augmentation des temporisations associées (***timer backoff***):
  - une valeur initiale de temporisation est calculée
  - si une retransmission est effectuée, la temporisation est augmentée (généralement le double de la précédente, jusqu'à une valeur plafond).
- Cet algorithme fonctionne bien même avec des réseaux qui perdent des paquets.

TCP : la congestion

Gestion de la congestion

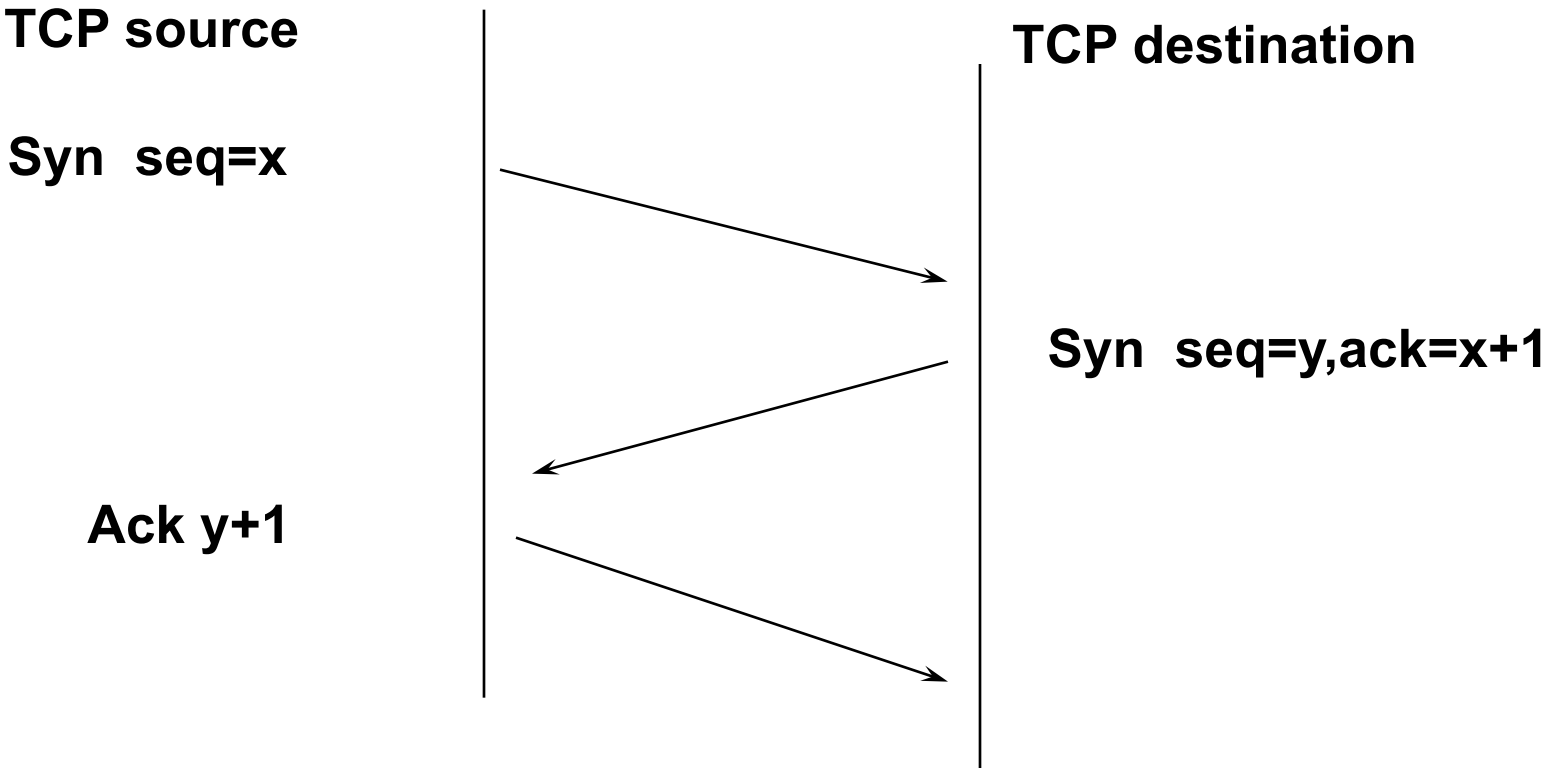
- TCP gère le contrôle de flux de bout en bout mais également les problèmes de congestion liés à l'interconnexion.

- La congestion correspond à la saturation de noeud(s) dans le réseau provoquant des délais d'acheminement de datagrammes jusqu 'a leur pertes éventuelles.

Dans la technologie TCP/IP, les passerelles (niveau IP) utilisent la réduction du débit de la source mais TCP participe également à la gestion de la congestion en diminuant le débit lorsque les délais s'allongent.

TCP : connexion

Une connexion TCP est établie en trois temps de manière à assurer la synchronisation nécessaire entre les extrémités :



Ce schéma fonctionne lorsque les deux extrémités effectuent une demande d'établissement simultanément.

TCP ignore toute demande de connexion, si cette connexion est déjà établie.

TCP retransmissions

- Si la congestion disparaît, TCP définit une fenêtre de congestion égale à 1 segment et l'incrémente de 1 chaque fois qu'un acquittement est reçu; ce mécanisme permet un démarrage lent et progressif :

  Fenêtre_congestion = 1,
  émission du 1er segment,
  attente acquittement,
  réception acquittement,

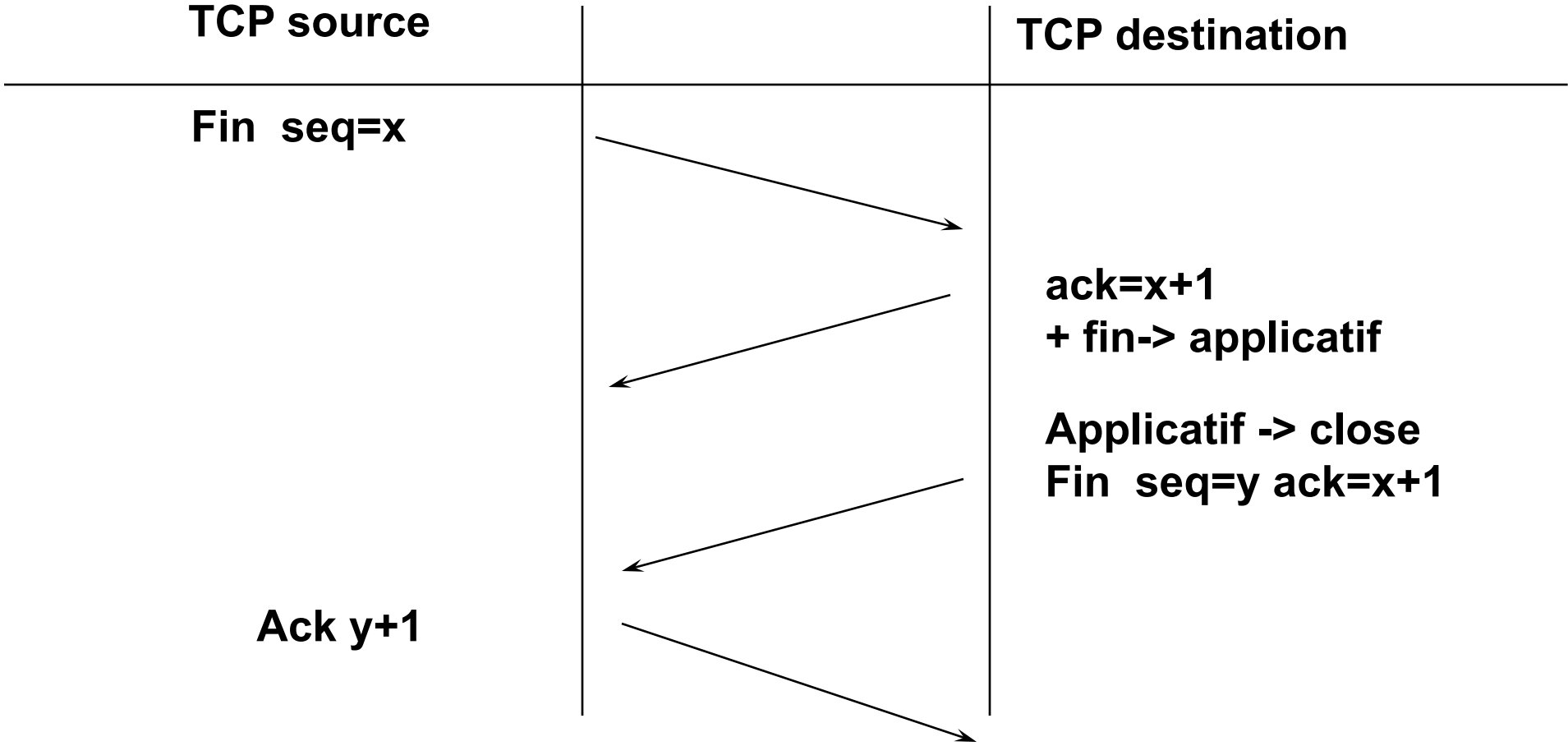
  Fenêtre_congestion = 2,
  émission des 2 segments,
  attente des acquittements,
  réception des 2 acquittements,

  Fenêtre_congestion = 4,
  émission des 4 segments, ...

Log2 N itérations pour envoyer N segments. Lorsque la fenêtre atteint une fois et demie sa taille initiale, l'incrément est limité à 1 pour tous les segments acquittés de la fenêtre.

TCP : déconnexion

- Une connexion TCP est libérée en un processus dit "trois temps modifié":



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	WWW
110	POP3	Post Office Protocol - Version 3
111	SUNRPC	SUN Remote Procedure Call

TCP : L'automate