Plan

Services Transport, protocole TCP

Généralités

- le rôle d'une Couche Transport
- les Services
- les Protocoles

Le Service TCP

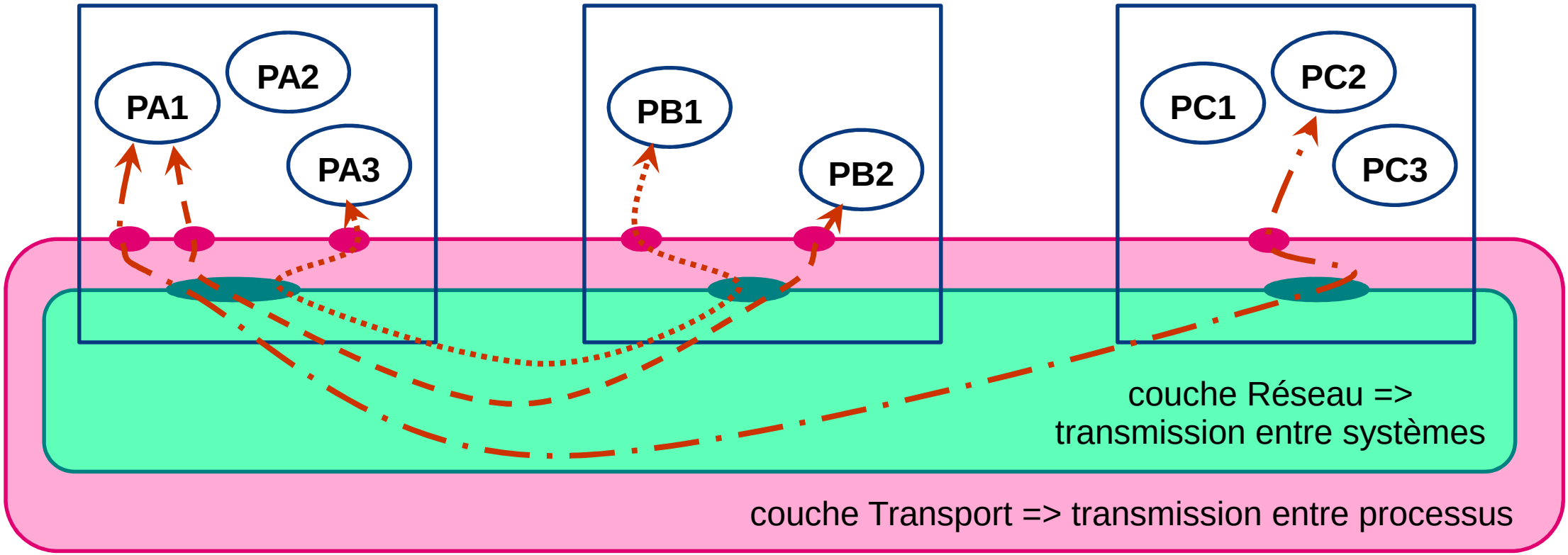
- Service, facilités, primitives de service, adressage

Le Protocole TCP

- transmission de données
  - détection et traitement des pertes et duplications
  - contrôle de flux
- établissement et terminaisons de connexion

Couche Transport

Le rôle de la couche Transport est d'acheminer des octets entre des processus s'exécutant dans des systèmes appartenant à la même couche Réseau, sans qu'ils aient à se préoccuper des problèmes de médium, de routage, etc...



Couche Transport - Généralités

Service

- transmission de données entre processus (de bout en bout)
- données normales, données express
- transparence
- mode connecté, mode non connecté

Protocole

- classes de protocole, selon la qualité du Service Réseau
- optimisation des ressources Réseau

Les couches Transport les plus utilisées

- ISO
  - 1 Service en mode connecté, 5 classes de protocole
  - 1 Service en mode non connecté, 1 protocole
- technologie TCP/IP
  - 1 Service en mode connecté (TCP), 1 protocole
  - 1 Service en mode non connecté (UDP), 1 protocole

Les protocoles de Transport

	Transport ISO						technologie TCP/IP	
	mode connecté					mode non connecté	TCP	UDP
	0	1	2	3	4			
connexion	oui	oui	oui	oui	oui	non	oui	non
transfert de données	oui	oui	oui	oui	oui	oui	oui	oui
correction des erreurs signalées	non	oui	non	oui	oui	non	oui	non
contrôle de flux	non	non	oui	oui	oui	non	oui	non
données express	non	non	oui	oui	oui	non	oui	non
détection et correction des erreurs non signalées	non	non	non	non	oui	non	oui	non
multiplexage	non	non	oui	oui	oui	non	non	non
éclatement sur plusieurs connexions Réseau	non	non	non	non	oui	non	non	non

Classes (de qualité) de Service Réseau →	**A**	**B**	**C**
détecte les erreurs	oui	oui	non
taux d'erreurs détectées	fort	fort	nul
tente de corriger les erreurs détectées	oui	non	non
taux d'erreurs corrigées	fort	faible	nul
signale les erreurs non corrigées	oui	oui	non

Les Services de Transport
TCP et UDP

TCP
Service en mode connecté

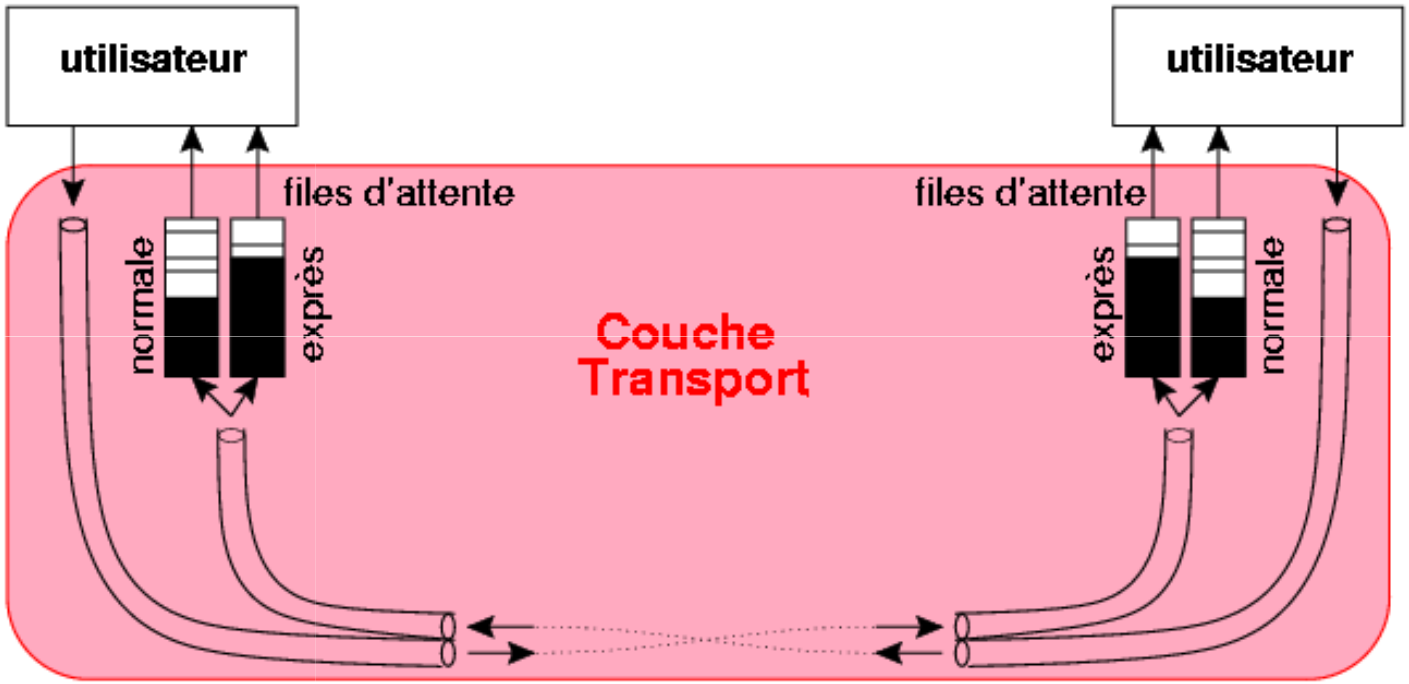
- jusqu'à 65534 points d'accès par système
- détection et tentative de correction des pertes et duplications
- signalement des erreurs
- livraison dans l'ordre d'émission
- contrôle de flux

UDP
Service en mode non connecté

- jusqu'à 65534 points d'accès par système
- un ou plusieurs destinataires

Le Service TCP

- mode connecté → canaux de transmission indépendants
- files d'attente des octets en attente de livraison
- établissement de connexion TCP = création des 2 canaux
- terminaison de connexion = destruction des canaux
- 3 facilités
  - établissement
  - transmission
  - terminaison



Les points d'accès aux Services TCP et UDP

Le concept de point d'accès au Service TCP (resp. UDP) est traduit par le concept de socket des systèmes d'exploitation

- création de *socket* selon les besoins des processus
  - par la fonction **`socket()`**
  - une *socket* appartient à un processus
  - une *socket* est détruite au plus tard à la mort du processus

- attribution d'une adresse de point d'accès (numéro de port)
  - par la fonction **`bind()`**
  - des restrictions :
    - numéros de port privilégiés : 1..1023
    - numéros de port bien connus
    - numéros de port enregistrés

voir http://www.iana.org/assignments/port-numbers

Service TCP : Primitives du Service

Etablissement de connexion

- appelé **`listen`** déclaration du rôle "appelé"
  `accept` attente et acceptation d'appel
- appelant **`connect`** appel

Transmission d'octets

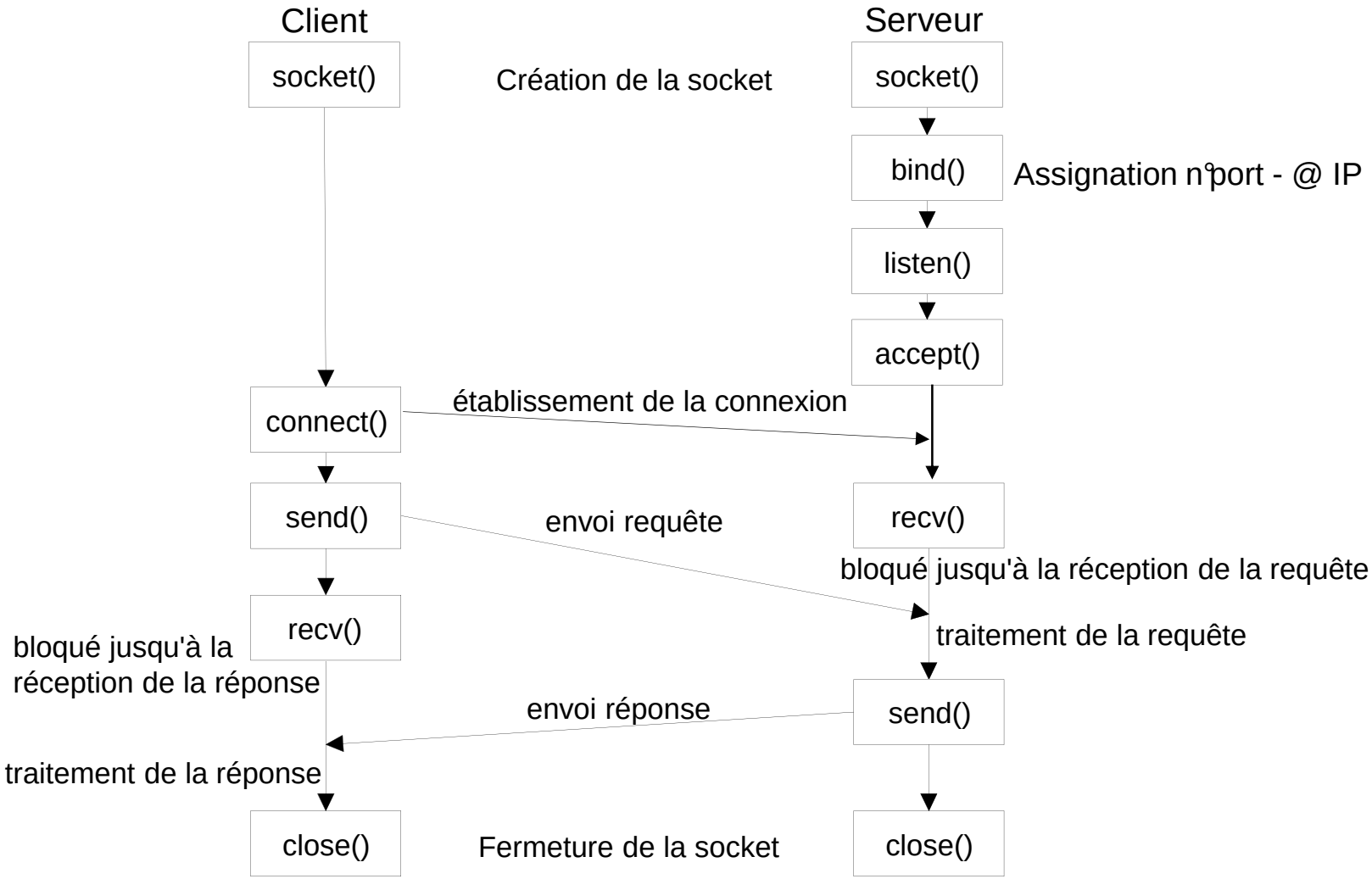
- demande d'envoi **`write`** ou **`send`**
- livraison **`read`** ou **`recv`**

Terminaison de connexion

- ordonnée **`shutdown`** (pour chaque voie)
- brutale **`close`** (la plus utilisée)

(explications dans le manuel Unix)

Mode connecté



Adresse des points d'accès au Service TCP

adresse d'un TCP-SAP

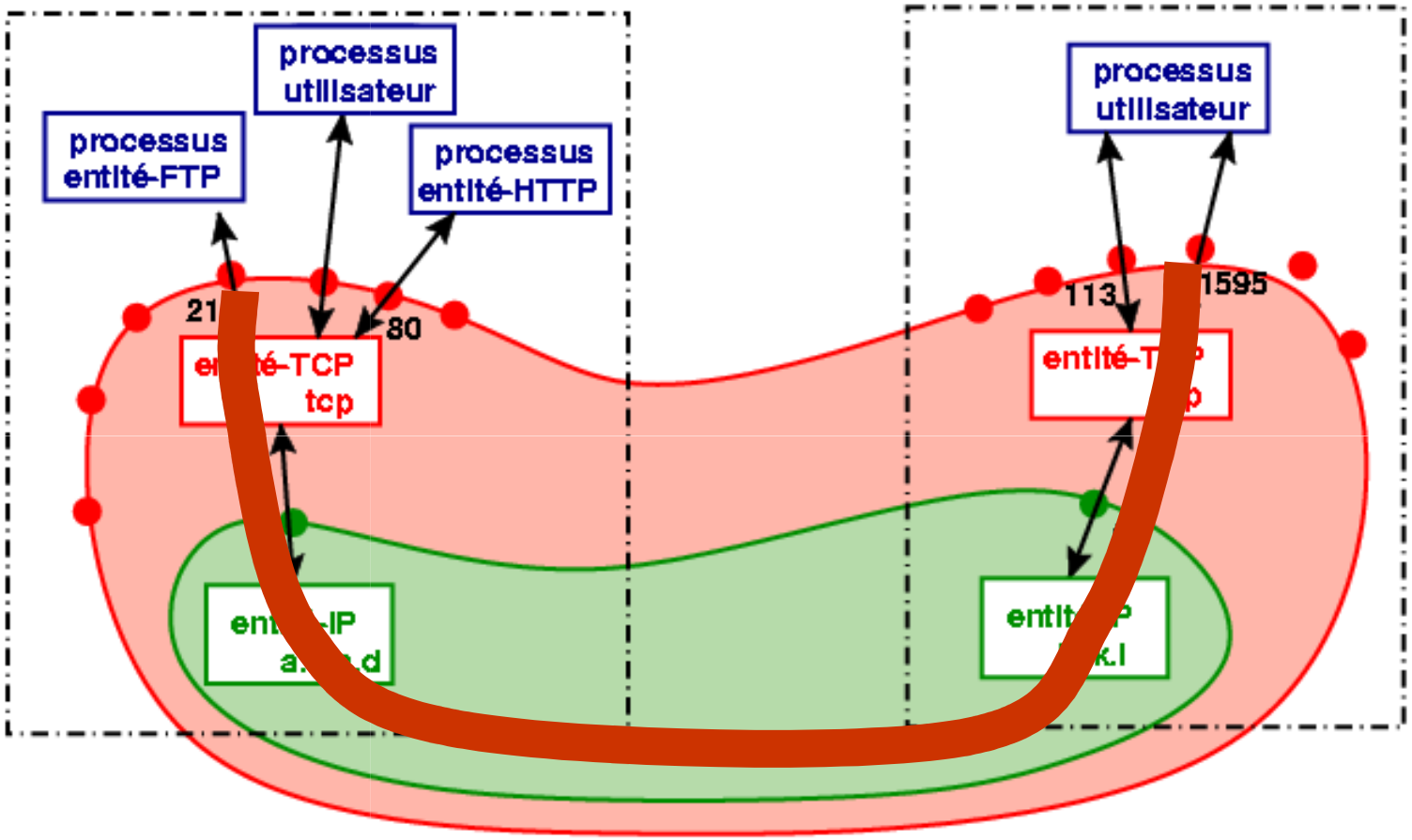
- identification de l'entité-IP du système (adresse-IP)
- adresse du point d'accès au Service-IP pour les entités-TCP [6]
- adresse du point d'accès au Service-TCP (port)
  - privilégié
  - bien connu
  - enregistré

a.b.c.d-tcp/21

identification d'une connexion-TCP

•les TCP-SAP des deux extrémités de la connexion-TCP

a.b.c.d/21 – i.j.k.l/1595



Transmission de données par UDP ou TCP

Les données sont transmises dans des PDU-TCP (resp.PDU-UDP), transportées par le Service IP qui livre ... peut-être, une seule fois

Les risques :

- perte de PDU-IP → perte de la PDU-TCP ou PDU-UDP encapsulée et des données qu'elle contient
- livraisons multiples de la même PDU (duplication), donc de données
- livraison d'une suite d'octets dans un ordre différent de celui de la soumission

- UDP ne fait ni détection, ni correction
- **TCP tente de détecter tous les problèmes et de les corriger**

Protocole TCP - détection des pertes et duplications

Numérotation lors de l'expédition

- les octets de l'utilisateur sont transmis par bloc dans des PDU-TCP
- l'entité-TCP calcule le n° du 1er de ces octets et le place dans la PDU (seq)
- initialisation de `seq` : à l'établissement de connexion
- incrémentation : n°bloc = n° bloc précédent + lg.bloc précédent

Vérification lors de la réception

- n°reçu > n°attendu → une perte
- n°reçu < n°attendu → une duplication
- n°reçu = n°attendu → OK

Correction

- duplication : l'entité-TCP ignore la PDU-TCP en doublon
- perte : retransmission de la PDU contenant le bloc (quand ?)
  - l'entité-TCP qui reçoit indique dans chaque PDU qu'elle envoie le n°du prochain octet qu'elle attend (`ack`)
  - l'entité-TCP qui expédie conserve en mémoire les blocs d'octets qu'elle envoie; si elle n'a pas reçu `ack` d'un bloc dans le temps défini (RTT), elle renvoie le bloc

Protocole TCP - contrôle de flux

Une entité-TCP place les octets reçus pour l'utilisateur dans des files d'attente où il peut en prendre livraison

- place dans la file d'attente = capacité de réception
- file d'attente pleine ⇨impossible d'ajouter de nouveaux octets reçus et consommation inutile de ressources de (re)transmission

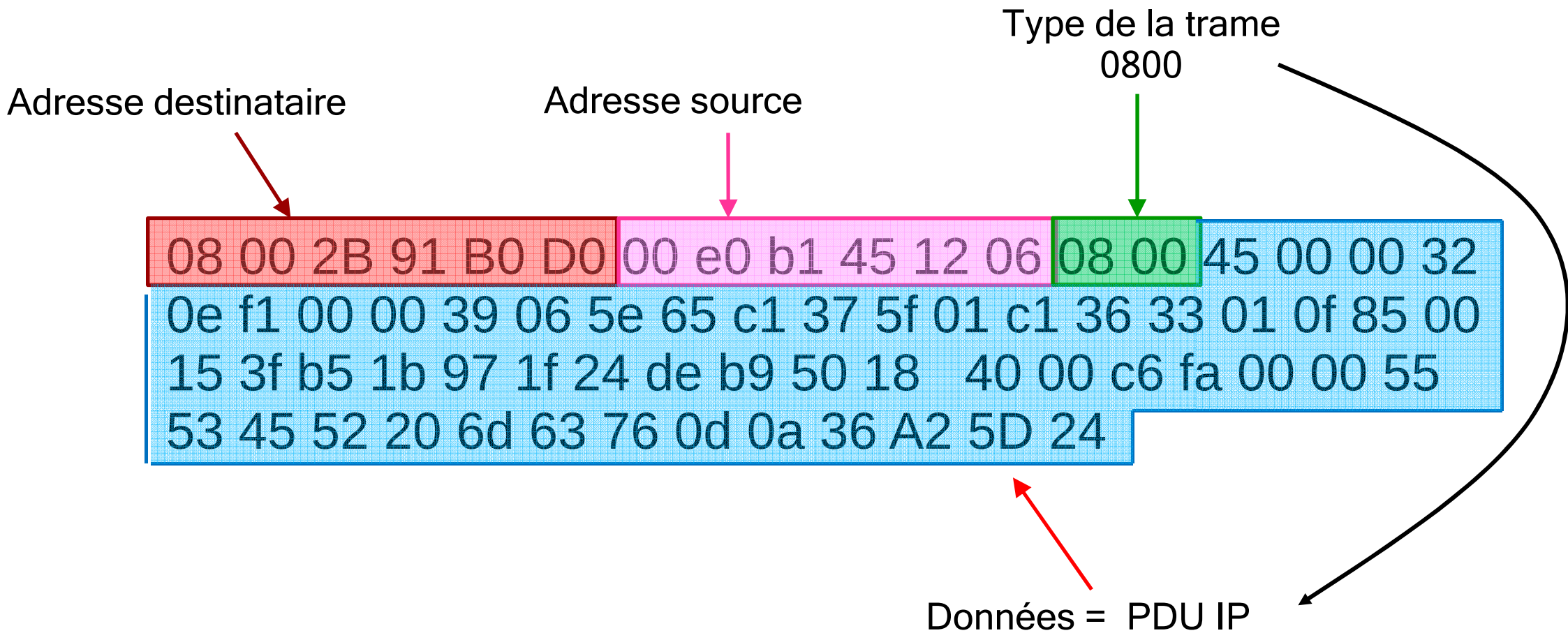
Le contrôle de flux permet d'éviter à l'entité-TCP expéditrice d'envoyer plus que l'entité-TCP réceptrice peut mémoriser dans les files d'attente

- chaque entité-TCP place dans chaque PDU qu'elle envoie le nb d'octets libres dans la file d'attente (champ `win`)

Remarque : autre technique : "stop and go"

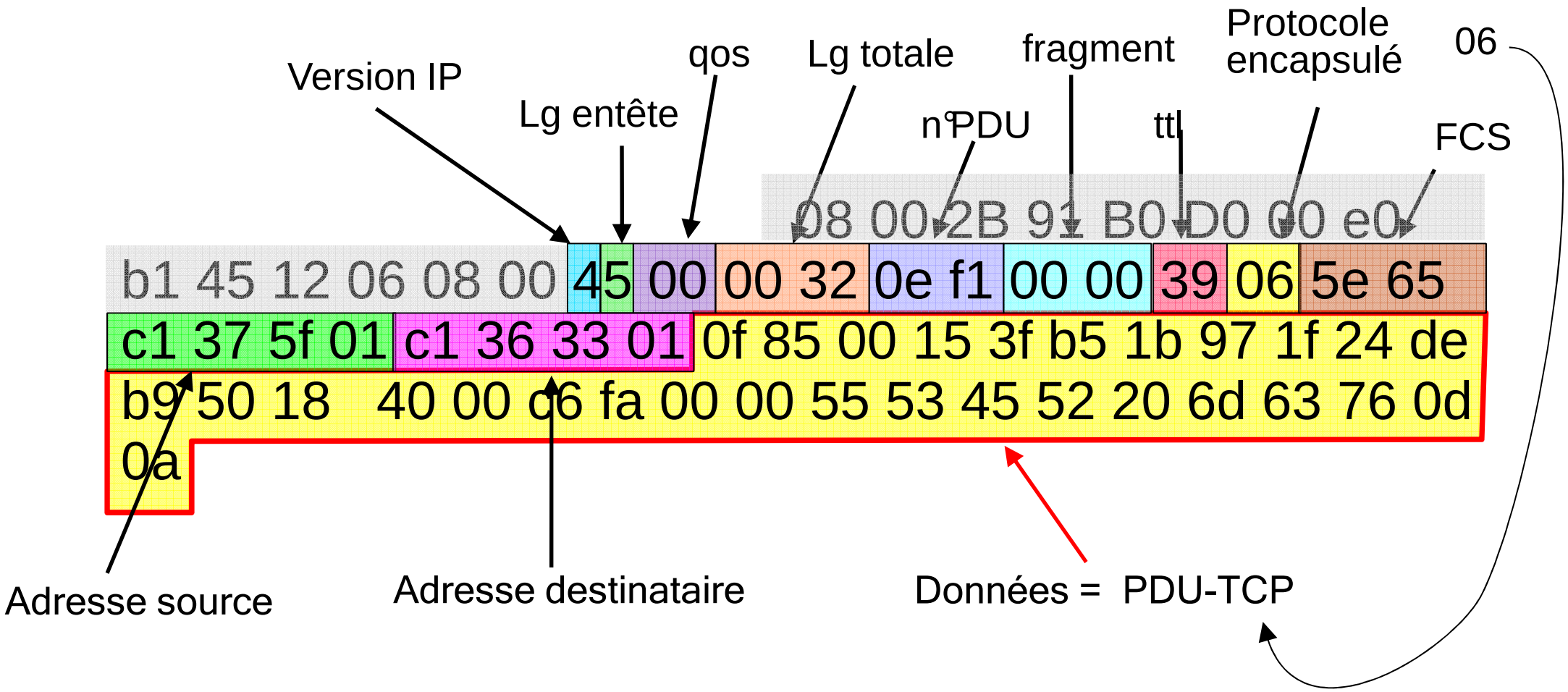
Exemple de trame

1) la PDU MAC-Ethernet

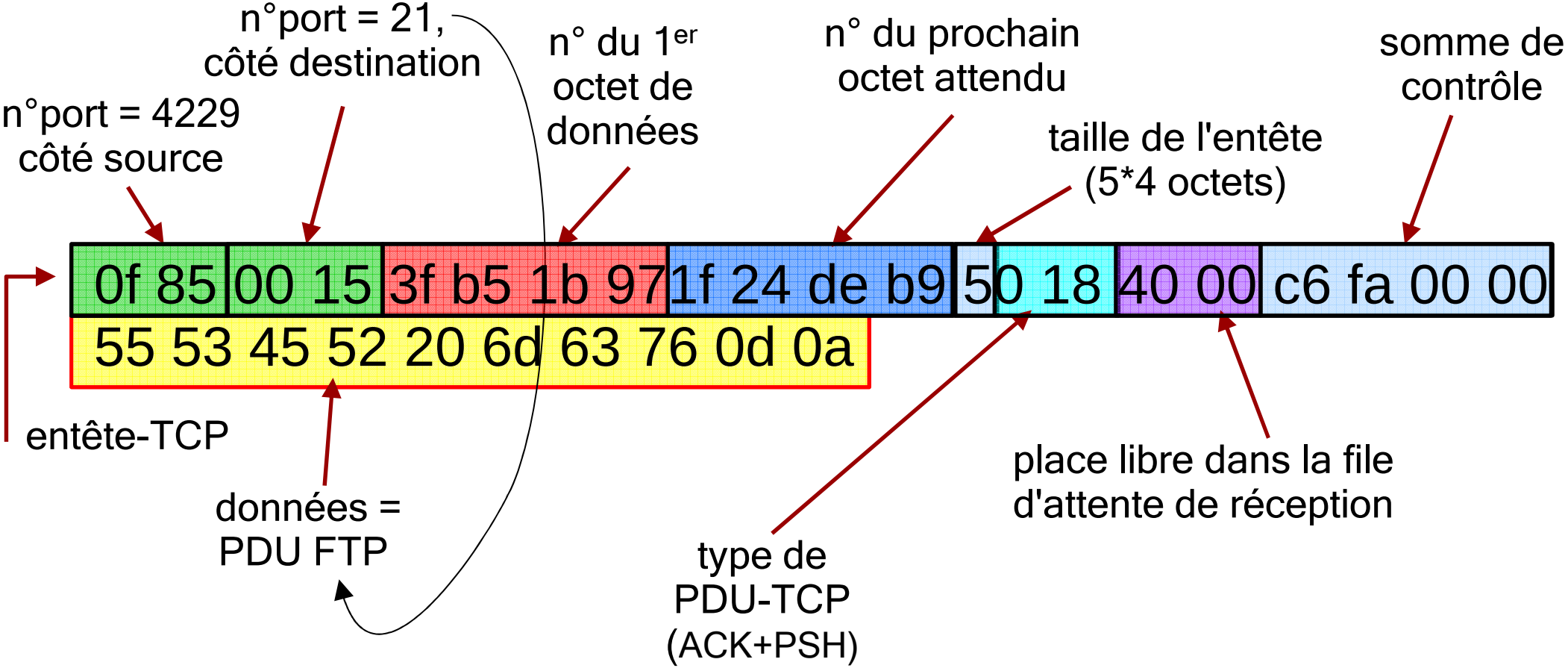


Exemple de trame

2) la PDU IP



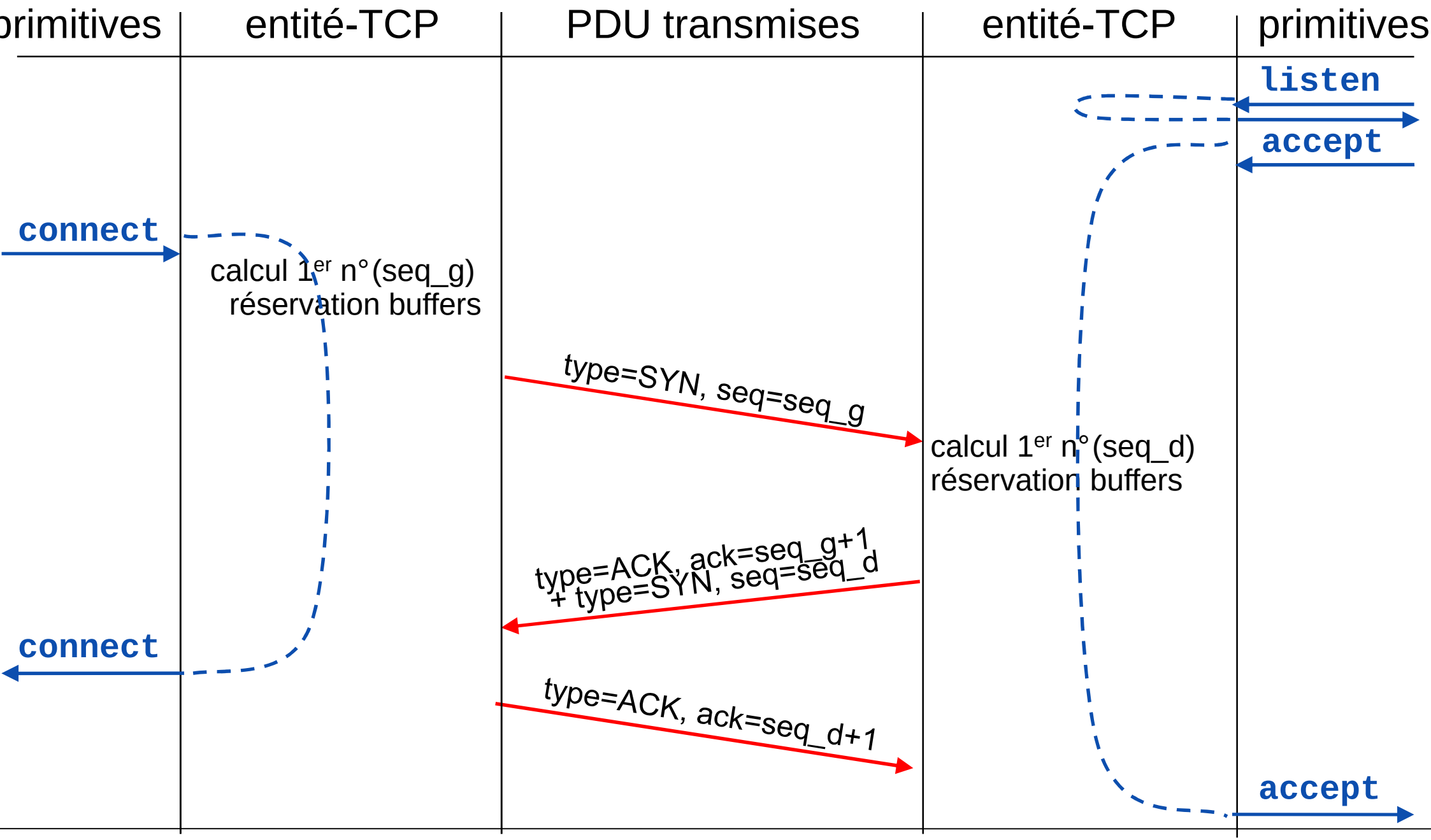
Exemple de PDU-TCP



type de PDU-TCP

URG	ACK	PSH	RST	SYN	FIN
20	10	8	4	2	1

Protocole TCP
établissement de connexion



Protocole TCP - terminaison de connexion

Terminaison ordonnée

- sur primitive **`shutdown`**
- l'entité-TCP avertit l'autre entité-TCP qu'il n'y aura plus de transmission par une PDU-TCP de type FIN,
- l'autre acquitte par une PDU-TCP de type FIN+ACK
- 2 fois (pour chaque sens de transmission)

Terminaison brutale

- sur primitive **`close`**
- l'entité-TCP envoie une PDU-TCP de type RST,
  et détruit les buffers (les données non livrées sont perdues)
- l'autre entité-TCP détruit les buffers (les données non livrées sont perdues),
  et avertit l'utilisateur par une erreur sur la prochaine primitive

Le Pare-feu

- Un pare-feu (firewall en anglais), est un système qui permet de protéger un réseau local des intrusions de personnes en provenance d'Internet, donc du réseau extérieur à l'entreprise.
C'est un système permettant de bloquer des ports et des adresses IP, c'est-à-dire en interdire l'accès aux personnes provenant de l'extérieur

- Fonctionnement

Filtrage des paquets:

-> analyse de l'entête des paquets TCP (UDP) et de l'entête du paquet IP

exemple filtre suivant :

L'adresse IP de la machine émettrice

L'adresse IP de la machine réceptrice

Le protocole (TCP, UDP, ICMP, ARP, ...)

Le numéro de port, …