

## Attendus

- adressage IP
  - principes (hiérarchie, unicité)
  - limites du système des classes
  - calculs sur les adresses et masques
- relaying IP
  - principes de la table de routage
  - recherche dans la table (calcul)
- interface avec la couche liaison
  - lien entre l'adressage à la couche liaison et l'adressage à la couche réseau
  - principe du protocole ARP
- contrôle IP
  - les principales erreurs (ICMP)
- couche transport
  - distinction UDP et TCP
  - notion de ports
- TCP
  - ouverture et fermeture d'une connexion
    - SYN et FIN, rôle des numéros de séquence et d'acquittements
  - principe des mécanismes pour la fiabilité (numéros de séquences, acquittements)
  - principe des mécanismes de contrôle de flux (fenêtre de réception)
- décodage
  - savoir décoder une trace hexadécimale à l'aide du rappel des formats des trames, paquets etc
- interprétation d'un relevé de trafic
  - à partir d'un relevé de trafic (décodé), savoir expliquer ce qui se passe
  - pouvoir vérifier la cohérence d'un relevé de trafic, ou compléter un relevé partiel
- outils
  - savoir lire et configurer des tables de routage sous Linux
  - savoir déboguer le routage (commandes ping et traceroute)
  - usage basique de l'analyseur de trafic wireshark

# Le protocole IP

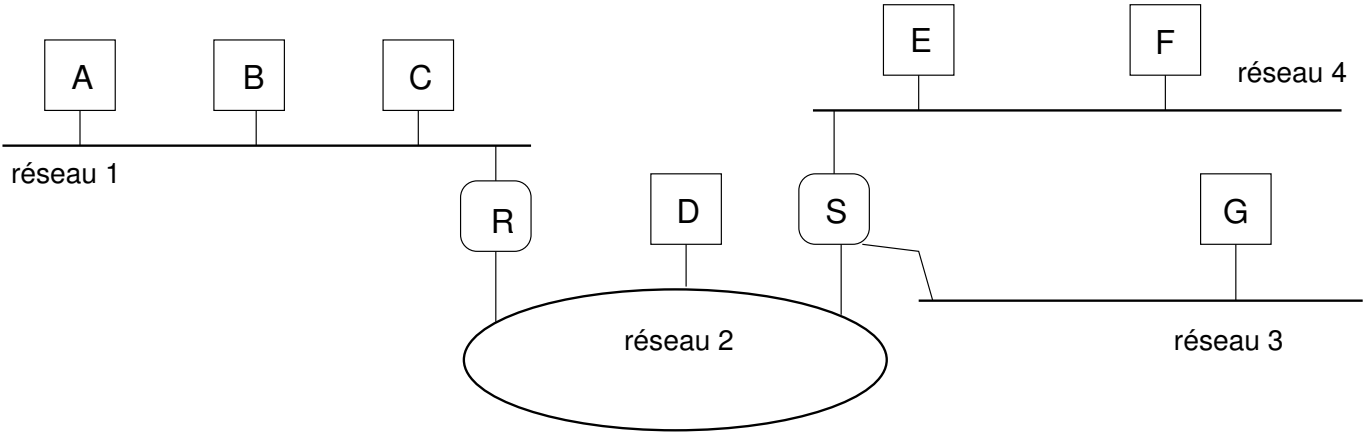


FIGURE 1 – Notre internet

On donne un nom à chaque *interface* (connexion d’une machine à un réseau) :

A : a, B, b... G : g

R : r1 sur le réseau 1, r2 sur le réseau 2

S : s2, s3, s4

Adresses liaison nommées d’après l’interface :

A : La, B : Lb, ... G : Lg

R : Lr1 sur le réseau 1, Lr2 sur le réseau 2

S : Ls2, Ls3, Ls4

FIGURE 2 – Adresses liaison de notre internet

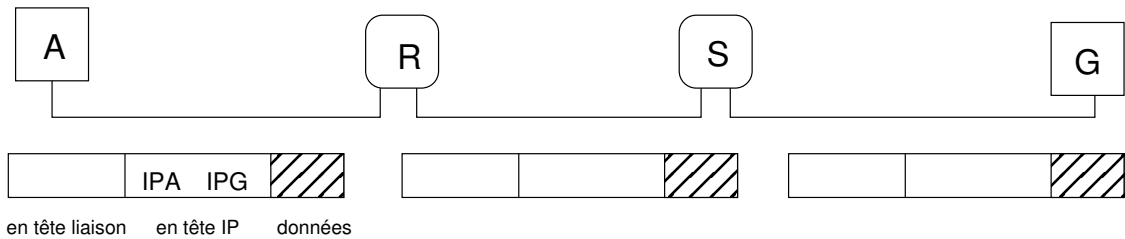


FIGURE 3 – Communication de A vers G

2 octets	2 octets	1	1	2 octets	6 octets	4 octets	6 octets	4 octets
0001	0800	06	04	Opér- ation	adr. L source	adr IP src	adr. L cible	adr. IP cible

↓

0001 = requête : who–has IPcible? Tell IPsrc, Lsrc

0002 = réponse : IPsrc is–at Lsrc

FIGURE 4 – Format des messages ARP

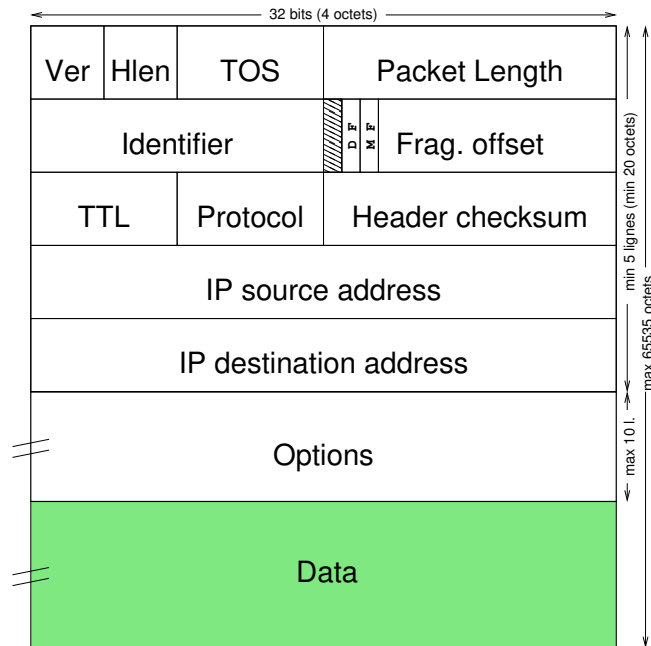


FIGURE 5 – Format d'un paquet IP

Valeur	Protocole
1	ICMP
4	IP in IP
6	TCP
17	UDP
89	OSPF

FIGURE 6 – Quelques valeurs possibles pour le champ Protocol

```

0x0000:  a0f3 c110 8776 a0f3 c110 8779 0800 4500
0x0010:  002c 814d 4000 4006 352c c0a8 0101 c0a8
0x0020:  0201 1000 2000 d6b4 525a 0000 0000 6002
0x0030:  16d0 a3f4 0000 0204 05b4

```

FIGURE 7 – Capture d'un paquet IP dans une trame ethernet (troisième trame du fichier capture1.cap)

	0	8	16	24	31
classe A	0	réseau		hôte	
classe B	1 0	réseau		hôte	
classe C	1 1 0	réseau		hôte	
classe D	1 1 1 0		adresse multicast		
classe E	1 1 1 1		réserve		

FIGURE 8 – Classes d'adresses IP (A, B, C sont obsolètes)

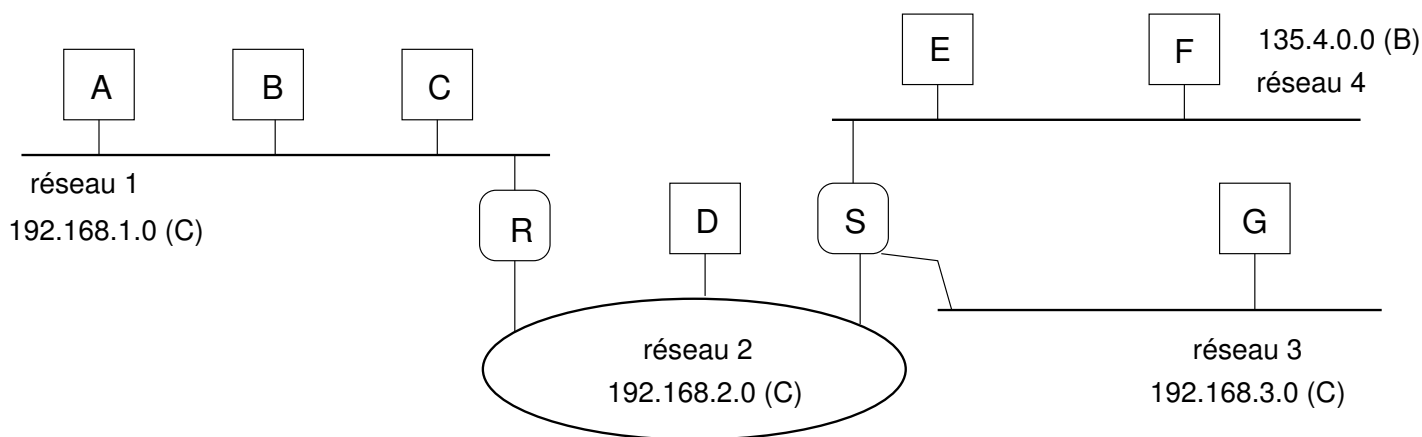


FIGURE 9 – Notre internet avec adresses en classes

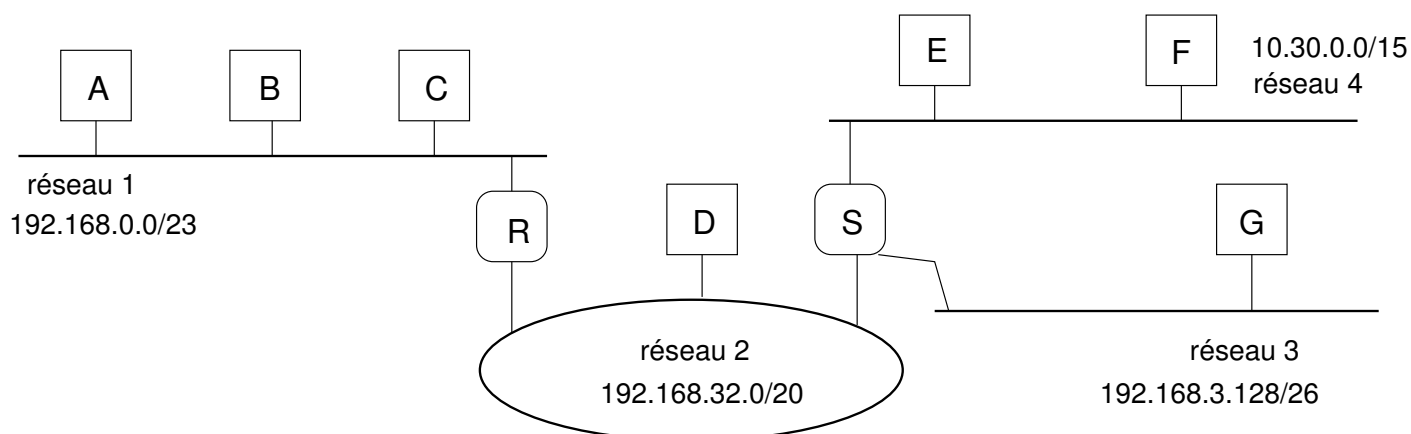


FIGURE 10 – Notre internet avec adresses *classless*

type	signification
Echo request	demande d'écho
Echo reply	... et réponse
Destination unreachable	réseau non joignable hôte non joignable port non disponible fragmentation nécessaire blocage administratif
Time exceeded	TTL arrivé à 0 ou réassemblage échoué
Redirect	je connais une meilleure route

FIGURE 11 – Principaux messages ICMP

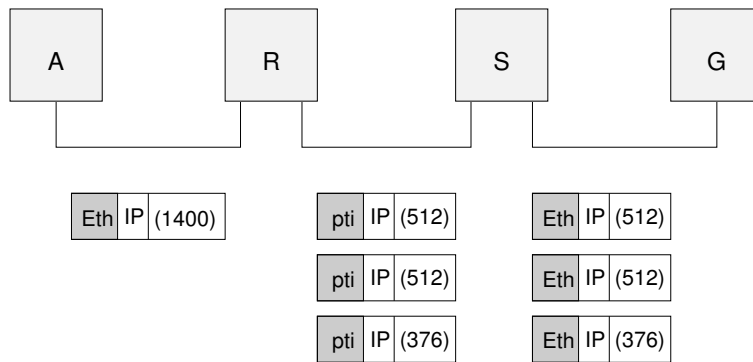


FIGURE 12 – Fragmentation d'un paquet

Début de l'en tête	Début de l'en tête	Début de l'en tête	Début de l'en tête
Ident = x    0 0    Offset = 0	Ident = x    0 1    Offset = 0	Ident = x    0 1    Off=64	Ident = x    0 0    Off=128
Fin de l'en tête	Fin de l'en tête	Fin de l'en tête	Fin de l'en tête
1400 octets de données	512 octets de données	512 octets de données	376 octets de données

FIGURE 13 – Fragmentation d'un paquet : détail des en-têtes

## Protocoles de transport : UDP et TCP

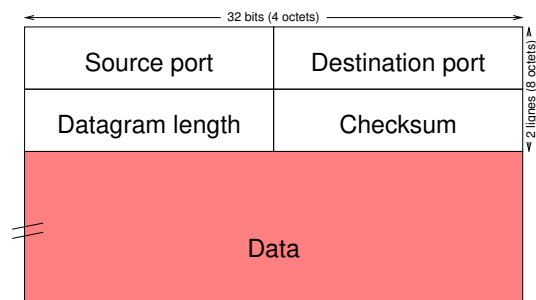


FIGURE 14 – Format du datagramme UDP

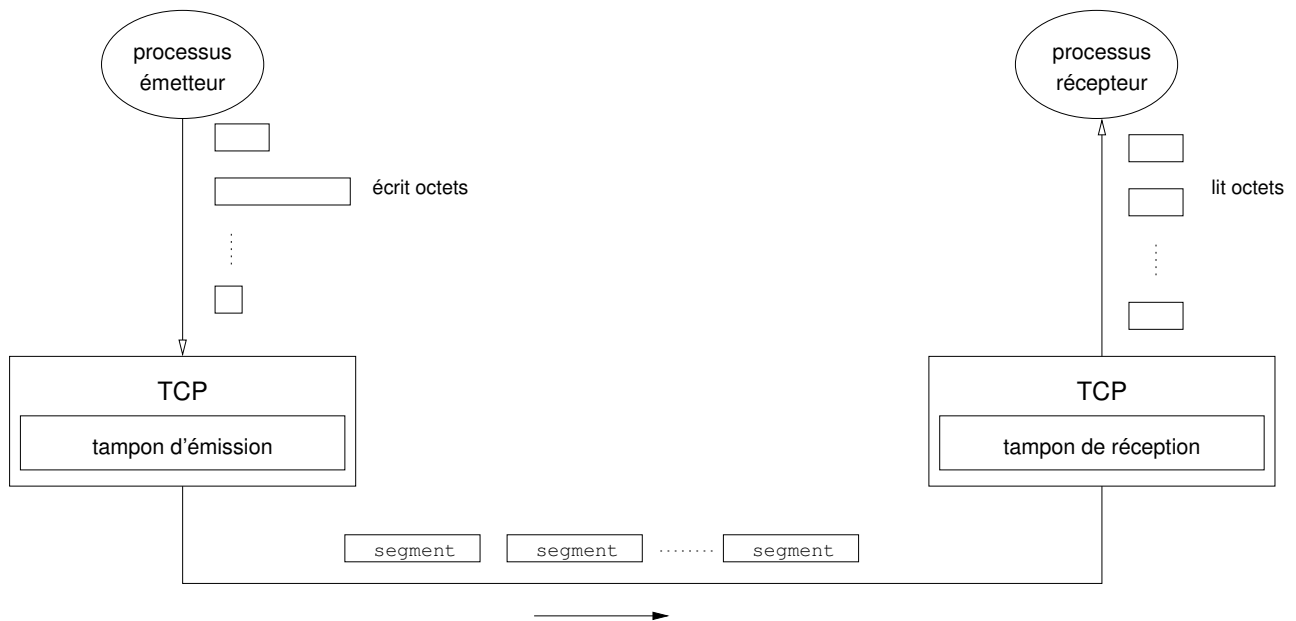


FIGURE 15 – Principes de TCP

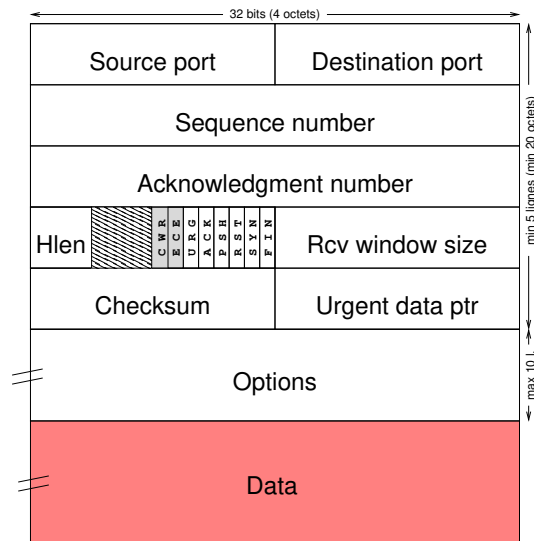


FIGURE 16 – Format du segment TCP

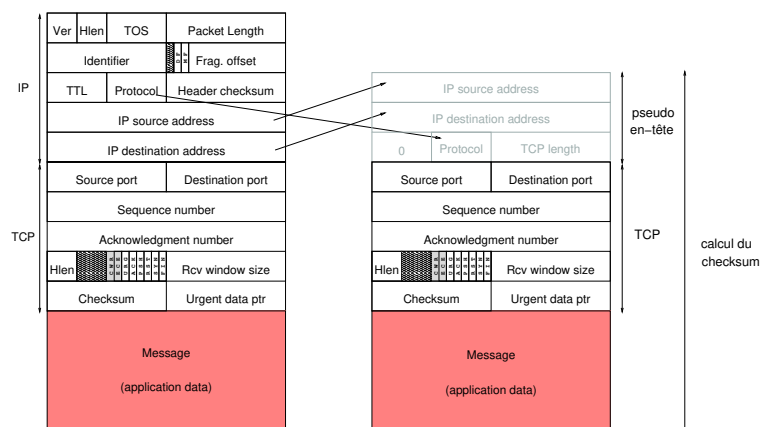


FIGURE 17 – Pseudo-entête pour le calcul du *checksum* TCP

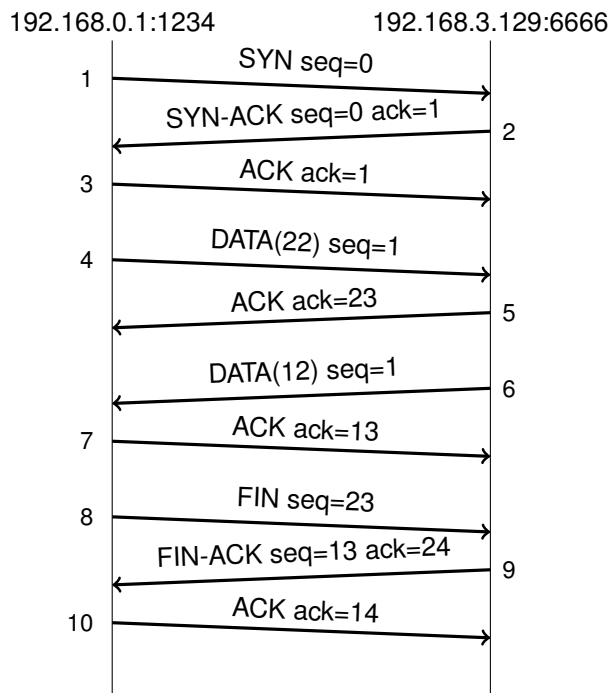


FIGURE 18 – Diagramme d'échanges de la capture 5

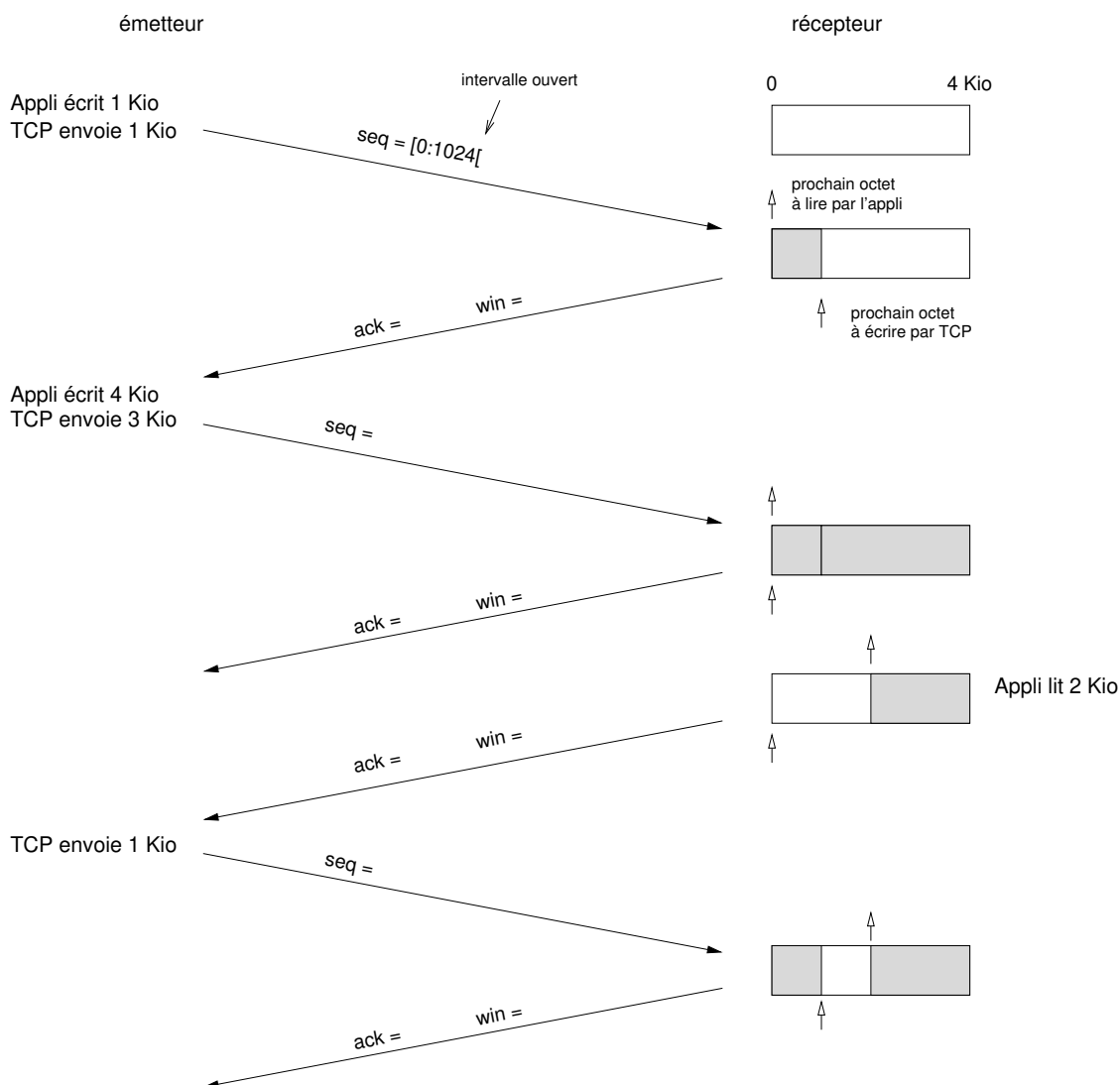


FIGURE 19 – Contrôle de flot TCP

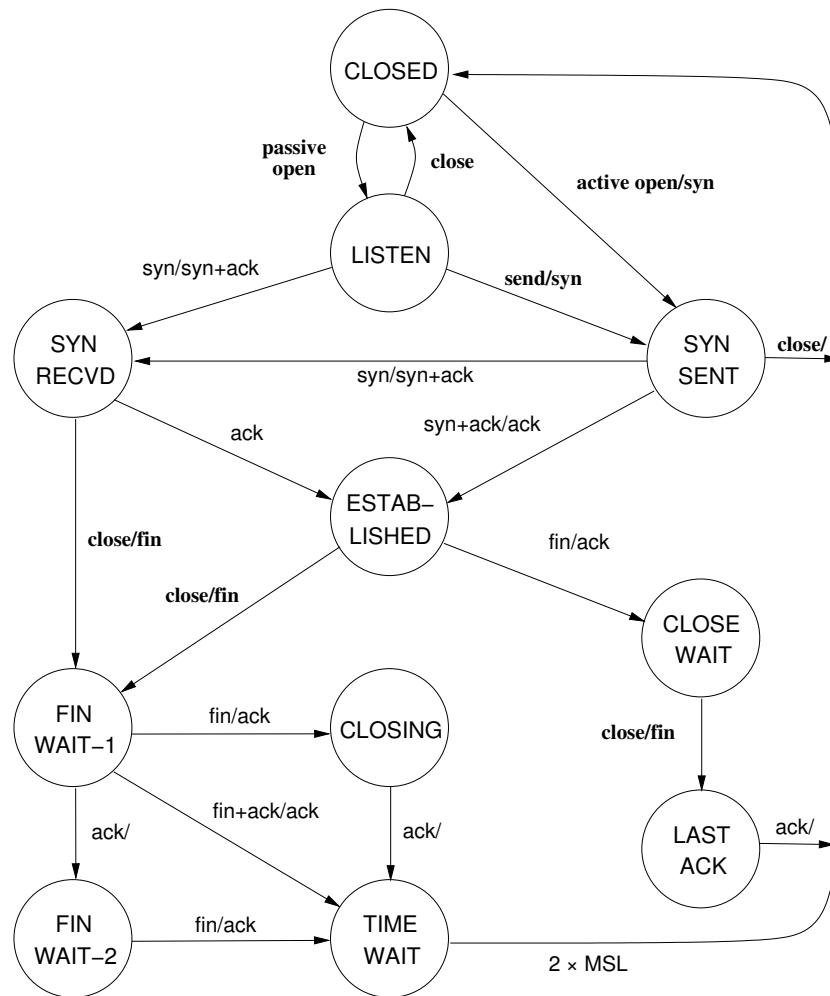


FIGURE 20 – Automate d'états fini de TCP (simplifié)