Le modèle TCP/IP

Fiche de synthèse

1 Un modèle en couches

Faire communiquer deux machines pose plusieurs problèmes : transmettre les bits, trouver le destinataire à travers plusieurs réseaux, vérifier que rien n'est perdu, comprendre le message. On les répartit en **couches** : chaque couche résout un problème et s'appuie sur la couche du dessous. Une couche peut changer sans toucher aux autres : on passe du Wi-Fi à Ethernet sans changer la façon de demander une page.

Couche	Rôle	Protocoles
4. Application	ce que veut faire le logiciel	HTTP, DNS, SMTP
3. Transport	acheminement de bout en bout, fiable ou non	TCP, UDP
2. Internet	adressage et routage entre réseaux	IP
1. Accès réseau	transmission sur le support	Ethernet, Wi-Fi

2 TCP et UDP

— **TCP** garantit que les données arrivent **dans l'ordre et sans perte** : il découpe le message en morceaux numérotés, attend un accusé de réception pour chacun et retransmet ce qui est perdu. Le Web et le courrier électronique l'utilisent.

— **UDP** envoie sans vérification : plus rapide, il accepte de perdre des données. Appels vidéo et jeux en ligne l'utilisent.

— Un **numéro de port** désigne le programme destinataire sur la machine : 80 pour HTTP, 443 pour HTTPS, 53 pour DNS.

3 Le protocole du bit alterné

Protocole du bit alterné

1. L'émetteur envoie un paquet avec un bit de contrôle, 0 pour le premier.
2. Le récepteur renvoie un **accusé de réception** (ACK) portant le même bit.
3. À sa réception, l'émetteur envoie le paquet suivant avec le bit inversé.
4. Sans accusé avant la fin d'un délai, fixé par un **temporisateur**, l'émetteur renvoie le même paquet, avec le même bit.

Un récepteur qui reçoit deux fois de suite le même bit sait que c'est un doublon, son accusé s'étant perdu : il l'ignore et renvoie l'accusé. Chaque paquet arrive une fois et dans l'ordre ; en contrepartie, attendre chaque accusé limite le débit.

Émetteur	Réseau	Récepteur
envoie A, bit 0	→	reçoit A ; ACK 0
reçoit ACK 0 ; envoie B, bit 1	perdu	
délai écoulé ; renvoie B, bit 1	→	reçoit B ; ACK 1
reçoit ACK 1 ; envoie C, bit 0	→	reçoit C ; ACK 0

4 L'encapsulation

À l'envoi, chaque couche ajoute son **en-tête** autour de ce que lui donne la couche du dessus : c'est l'**encapsulation**, comme une lettre dans une enveloppe, dans un carton, dans un conteneur. À l'arrivée, la **désencapsulation** retire les en-têtes un à un.

Couche	En-tête	Résultat	Repère
Application	HTTP	données	nom de domaine
Transport	TCP ou UDP	segment	port
Internet	IP	paquet	adresse IP
Accès réseau	Ethernet	trame	adresse MAC

5 DNS, URL et HTTP

— Le **DNS** traduit un nom de domaine, `google.com`, en adresse IP : c'est l'annuaire d'Internet.

— Une **URL** donne l'adresse complète d'une ressource : protocole, nom de domaine, chemin, et éventuellement des paramètres.

— **HTTP** suit le schéma requête puis réponse : `GET` demande une page, `POST` envoie des données ; la réponse contient un code, 200 pour un succès, 404 pour une page introuvable. **HTTPS** chiffre la communication.

À retenir

— Quatre couches : application, transport, internet, accès réseau.
— TCP est fiable, UDP est rapide sans garantie.
— Chaque couche ajoute un en-tête : segment, paquet, trame.
— Le DNS traduit les noms de domaine en adresses IP.