

Réseaux

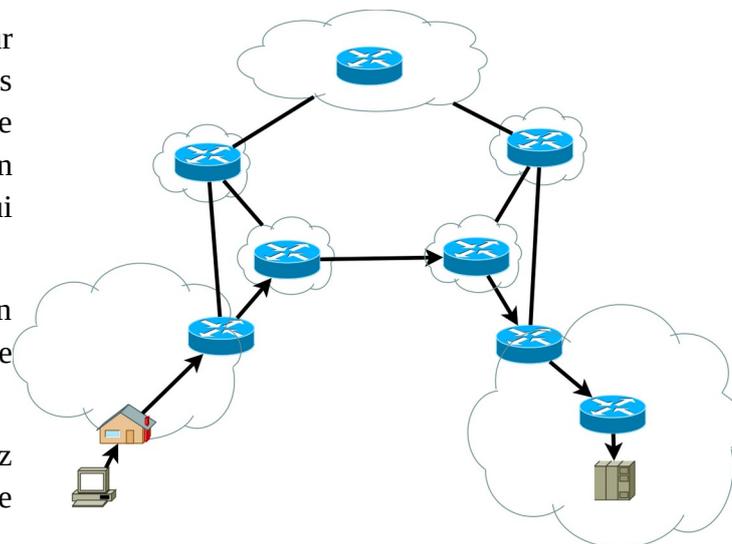
routage, encapsulation et bit alterné

I Le routage

Lorsque vous envoyez des données sur Internet, par exemple, lorsque vous faites une requête sur votre moteur de recherche favori, vos données sont découpées en sous-messages, on parle de paquets qui sont envoyés à travers Internet.

On s'intéresse donc au chemin pris par un paquet pour aller entre votre machine et le serveur auquel vous voulez l'envoyer.

Dans le schéma ci-contre, vous pouvez constater qu'il y a plusieurs chemins entre l'ordinateur domestique et le serveur.



Le routage consiste à mener les paquets à destination tout en essayant de prendre le chemin le plus rapide.

Dans ce cours nous nous limiterons à observer le chemin pris par des paquets de données.

À faire vous-même 1

- Ouvrir une console de commandes puis taper : `tracert flaustriat.frama.io` sous Windows ou `traceroute flaustriat.frama.io` sous Linux. Vous pouvez interrompre cette commande avec `Ctrl + C` si cela prend trop de temps.
- Observez le résultat.
La première ligne contient l'adresse IP de destination.
Une des deux lignes suivantes correspond à l'adresse IP de votre box où à celle de votre smartphone. Les autres lignes donnent les serveurs par lesquels sont passés les paquets.
- Ouvrir dans votre navigateur l'URL : <https://geotraceroute.com/new.php?host=&node=1768> . Puis dans le champ « Site » entrez « flaustriat.frama.io », « wikipedia.com » le site international de Wikipédia puis « government.ru » le site du gouvernement russe. Vous pouvez observer le chemin pris depuis un serveur à Paris vers ces sites. Vous pouvez bien sûr vous amuser à changer la localisation du serveur. Le logiciel libre et gratuit <https://visualtraceroute.net/> permet de voir sur une carte le chemin emprunté par des paquets depuis votre machine.

II Encapsulation des données

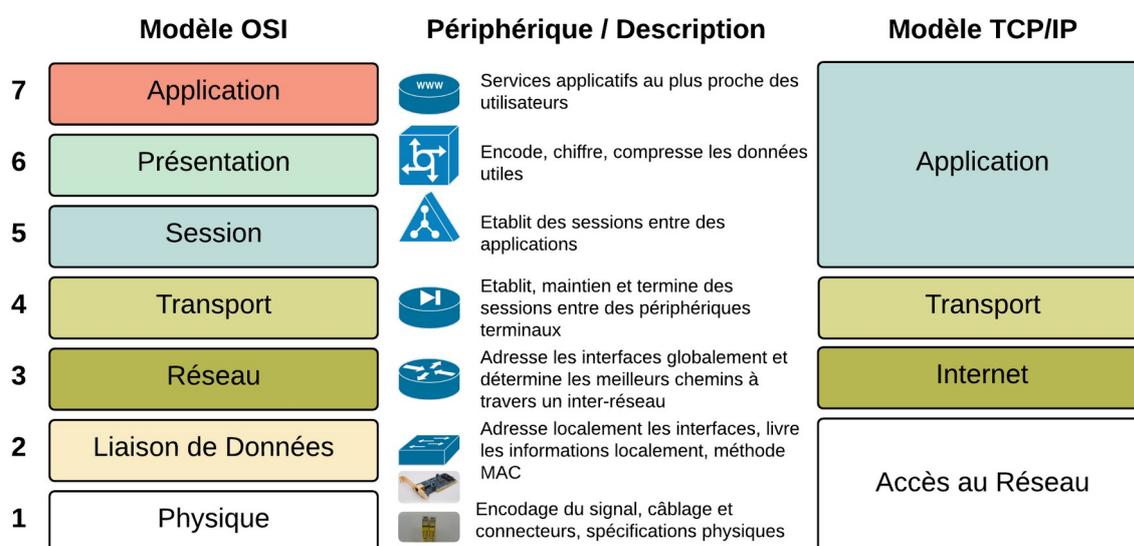
À faire vous-même 2

Regarder la vidéo :

<https://www.youtube.com/watch?v=0thnFumSdA>

Détaillons le modèle TCP/IP

Tableau rôle des différentes couches :

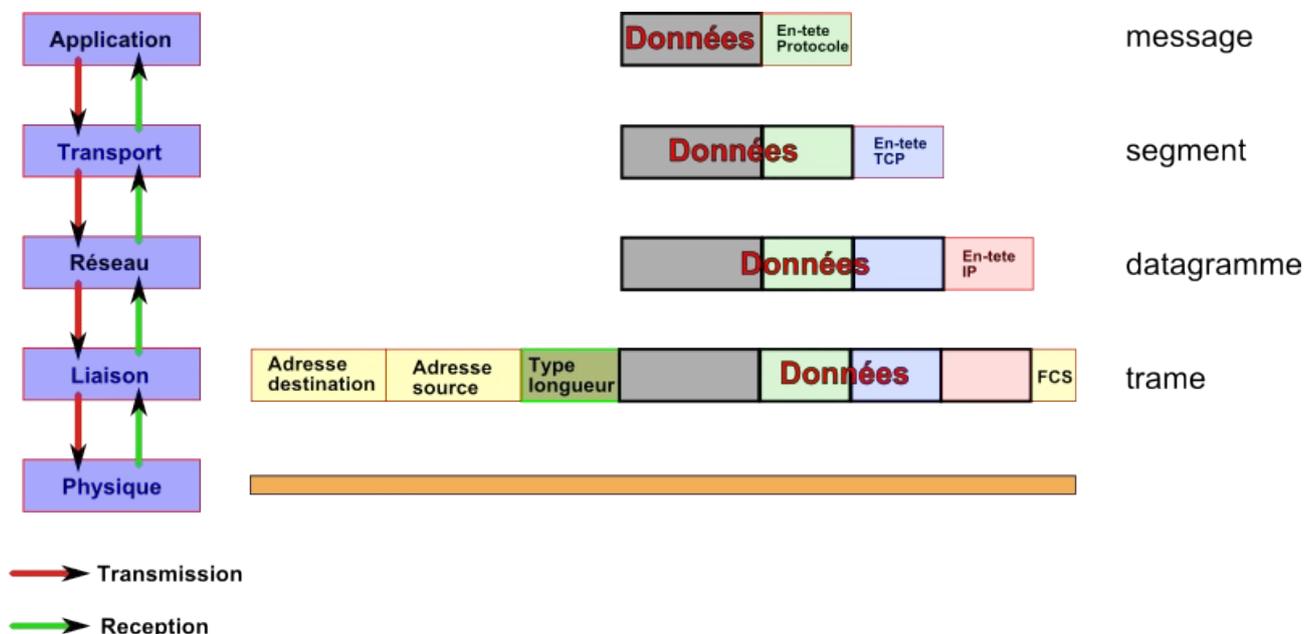


- **La couche 4 application** : cette couche regroupe les protocoles nécessaires à la bonne mise en forme d'un message (au sens large) avant sa transmission. Ces protocoles peuvent être de nature très différente : protocole HTTP pour la transmission de pages web, protocole FTP pour le transfert de fichiers, protocoles POP ou IMAP pour le courrier électronique...
- **La couche 3 transport** : Le protocole majeur de cette couche est le protocole TCP : Il s'assure par SYN-ACK que l'émetteur et le récepteur sont prêts à échanger des messages. Il découpe en segments numérotés le message à transmettre (côté émetteur) ou bien recompose le message total en remettant les segments dans l'ordre (côté récepteur). Les éléments échangés avec la couche inférieure sont des **segments**.
- **La couche 2 réseau** : C'est la couche où chaque segment numéroté est encapsulé dans un paquet qui, suivant le protocole IP, va contenir son adresse source et son adresse de destination. C'est à ce niveau que se décide si le message doit rester dans le réseau local ou être envoyé sur un autre réseau via la passerelle du routeur. Les éléments échangés avec la couche inférieure sont des **paquets**.
- La couche 1 **couche liaison/physique** : C'est l'encapsulation finale du message. Suivant le protocole Ethernet, les informations sont transmises d'une carte réseau à une autre, grâce à leur adresse MAC (Media Access Controller). Les éléments échangés avec la couche

inférieure sont des **trames**.

La couche physique où le message est transmis physiquement d'un point à un autre. Par signal lumineux (fibre optique), par ondes (wifi), par courant électrique (Ethernet), etc. Les éléments transmis sont les **bits**.

Tableau montrant l'encapsulation des données



L'encapsulation des données consiste lors de l'envoi de données et après avoir découpé en sous message les données à ajouter des entêtes qui correspondent aux protocoles des différentes couches (de 4 à 1), puis d'enlever les entêtes des couches 1 à 4 à la réception.

Lors de son parcours, une trame peut être partiellement décapsulée et remonter à la couche 3, avant de redescendre et de continuer son chemin. C'est le cas notamment lors du passage dans un routeur. Mais jamais, lors de son acheminement, le contenu réel du message n'est ouvert : les paquets transmis sont acheminés de manière identique, qu'ils contiennent les éléments constitutifs d'une vidéo YouTube ou d'un email à votre cousin.

Ce principe fondateur, actuellement menacé par certains acteurs politiques et industriels, est connu sous l'expression « **la neutralité du net** ». Cette neutralité garantie que vos données ne seront pas lues en chemin, comme lorsque vous postez une lettre son contenu n'est pas lu lors de son traitement par la Poste.

Tableau des identifiants par couche

Couche	Nom de l'encapsulation	Identifiant	Exemple
4 application	données	Un protocole et un nom de domaine	http:// suivi de www.wikipedia.fr
3 transport	segments	Port TCP ou UDP	TCP80 comme port par défaut pour HTTP
2 réseau	paquets/datagrammes	Adresse IPv4 et/ou IPv6	182.168.0.18
1 liaison	trames/bits	adresse physique (MAC 802)	70 : 56 : 81 : bf : 7c : 37

Remarques :

- Le protocole UDP est un protocole de communication plus rapide et moins fiable que TCP. Il n'y a pas de vérification de bonne réception (phase acknowledge ACK). Il est utilisé pour le streaming de voix ou d'images. En effet, même avec une perte éventuelle de segments de données, le message restera compréhensible : l'image sera pixelisée, ou bien il y aura de la friture sur la ligne, mais cela restera compréhensible.
- Les deux protocoles TCP et UDP peuvent être utilisés simultanément. C'est ce qui se passe lorsque l'on joue en réseau (en protocole TCP) tout en étant connecté à un serveur de discussion, un tchat (en protocole UDP). Ainsi le jeu peut laguer, alors que le tchat ne le fera pas : TCP est un protocole plus lent que UDP. Certains jeux de tirs à la première personne utilisent UDP pour palier à ce problème (ce qui entraîne d'autres, comme des délocalisations intempestives).

III Le protocole du bit alterné

Ce protocole est un exemple simple de fiabilisation du transfert de données. En effet, il faut s'assurer que les données ont bien été reçues. Un câble peut être rompu, le Wifi ne plus capter passé une certaine distance où passer sous un pont en utilisant la 4G de votre smartphone sont autant de raisons qui font que des données peuvent être perdues.

Contexte :

- Alice veut envoyer à Bob un message M, qu'elle a prédécoupé en sous-messages M0, M1, M2,...
- Alice envoie ses sous-messages à une cadence Δt fixée (en pratique, les sous-messages partent quand leur acquittement a été reçu ou qu'on a attendu celui-ci trop longtemps : on parle alors de *timeout*)

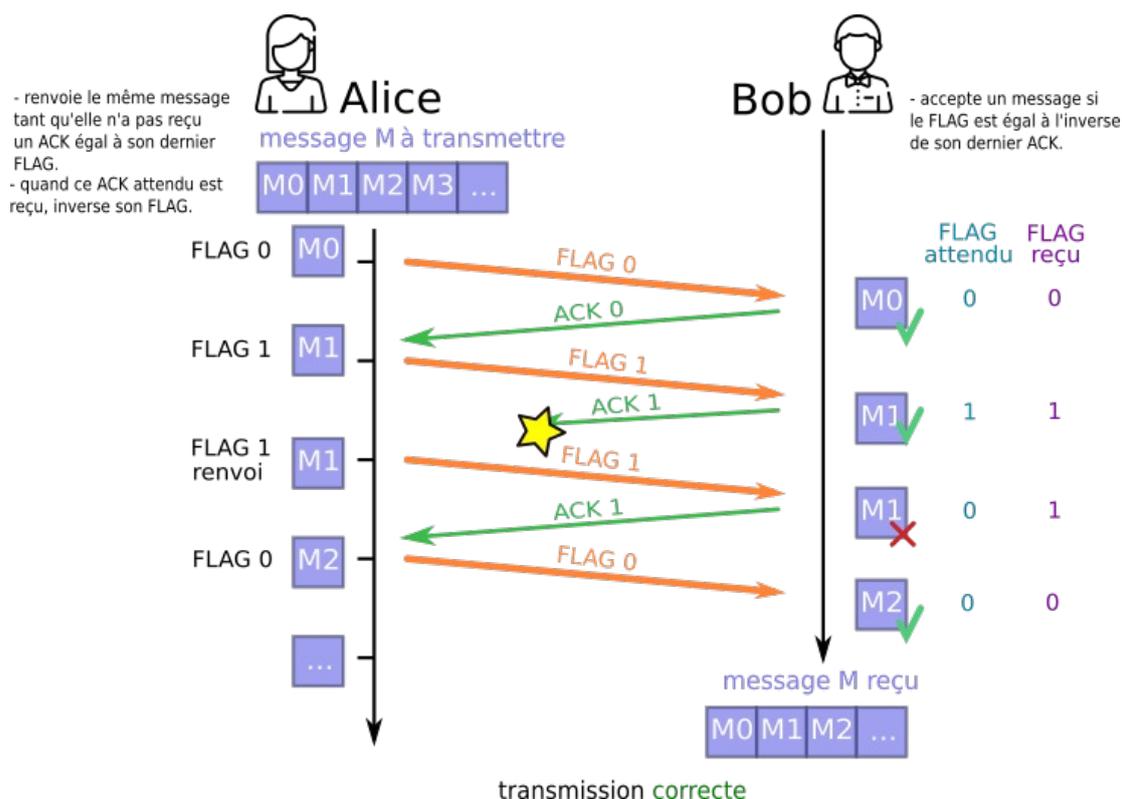
Le protocole du bit alterné est un protocole simple de la couche 1b (liaison) de TCP/IP, qui permet la retransmission de trames perdues ou corrompues.

Une fois la connexion établie, lors de l'envoi de données d'un serveur B à un client A, B rajoute un bit de séquence SEQ aux données, soit SEQ0 soit SEQ1. A dispose de deux ACK : ACK0 et ACK1. Ces bits uniques sont appelés drapeaux (flag en anglais).

- B envoie une trame de données continuellement avec par exemple SEQ0, jusqu'à ce qu'il reçoive un ACK0. Une horloge interne permet de déterminer la fréquence d'envoi de la trame.
- Quand A reçoit les données avec SEQ0, il envoie continuellement ACK0, jusqu'à ce qu'il reçoive des données avec SEQ1. Si les données sont corrompues, il ne renvoie pas ACK0. Là aussi une horloge donne la fréquence d'envoi du ACK.
- Quand B reçoit un ACK0, il change son numéro de séquence en SEQ1, et envoie continuellement la trame suivante de données jusqu'à recevoir un ACK1.
- Tout message reçu avec le mauvais SEQ ou le mauvais ACK sera considéré comme un « négative acknowledge », et ignoré.

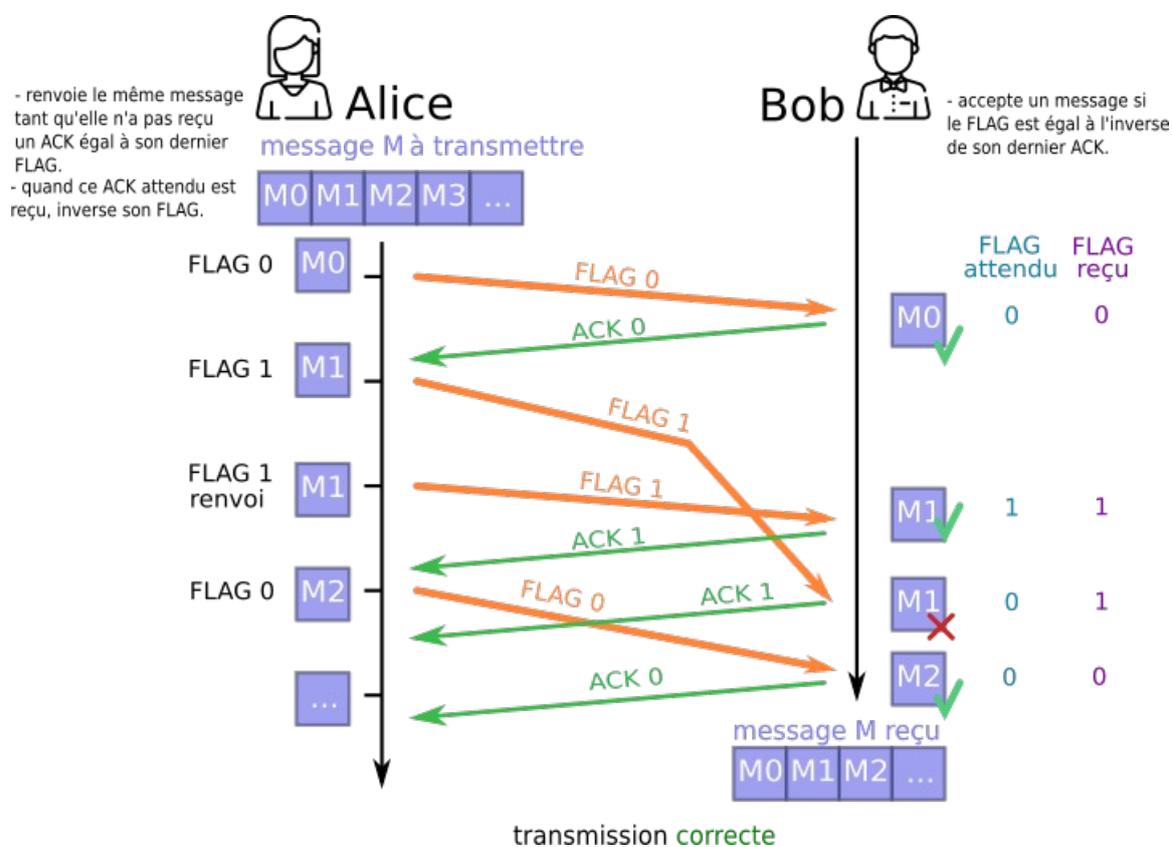
Le protocole peut être initialisé en envoyant des messages (données ou ACK) sans réel contenu, avec SEQ1 et ACK1. Le premier message SEQ0 sera considéré comme un vrai message.

Exemple :



À faire vous-même 3

Voici le cas où le sous-message M1 arrive en retard.



Que se passerait-il si le message M1 était encore plus en retard ?

Remarque :

Il n'est plus utilisé actuellement car remplacé par des protocoles plus efficaces mais aussi plus complexes.

Sources :

- https://github.com/glassus/nsi/blob/master/Premiere/Theme04_Architecture_materielle/04_Protocoles_de_communication.md
- https://pixees.fr/informatiquelycee/n_site/snt_internet_routage.html
- <https://cisco.goffinet.org/ccna/fondamentaux/modeles-tcp-ip-osi/>