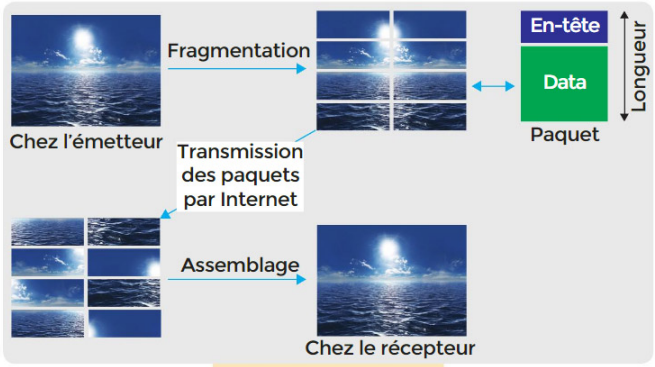
DES PAQUETS ET DES PROTOCOLES

(≈ 5 min)

https://youtu.be/dCknqcjcItU

p.18 :

DOCUMENT 2 Fragmentation et assemblage des paquets



Envoi d'un fichier sur Internet

Sur Internet, les données sont transférées sous forme de **paquets** (ou **datagrammes**). Chaque paquet est formé de deux grandes parties :
– **l'en-tête** où sont inscrites toutes les informations permettant l'acheminement ;
– **la partie « data »** qui contient une partie de la donnée à transférer.
Un paquet moyen a une taille de 128 ou 256 octets. Avec l'arrivée des technologies à très hauts débits, il est possible de transmettre des paquets de tailles plus importantes, aux alentours des 1 500 octets.
Un émetteur veut transmettre une photo par Internet, sous forme d'un fichier Jpeg de taille 200 ko (200 000 octets). La longueur maximale de transfert d'informations par câble électrique Ethernet étant de l'ordre de 1 500 octets, cette photo devra être décomposée en plusieurs paquets. Le fichier coupé en morceaux sera réassemblé chez le récepteur.
La réception des paquets ne se fait pas dans l'ordre dans lequel ils ont été émis : c'est l'une des caractéristiques d'Internet.

Cela signifie que, pour une photo de 5 Mo, il y aura environ 3 334 paquets (5 000 000 ÷ 1 500 ≈ 3 333,33) lors du passage par un câble Ethernet !

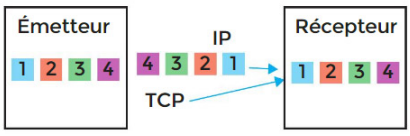
p.22 :

DOCUMENT 3 Protocole TCP/IP

Le rôle des protocoles IP et TCP est de permettre la fragmentation en paquets des données à transmettre et de les reconstituer ensuite dans l'ordre.
Le **protocole IP** (*Internet Protocol*) donne une adresse à toutes les machines du réseau. Ses principales fonctions sont :
- de définir le format des données (datagramme) ;
- d'assurer l'adressage et le routage de ces datagrammes jusqu'à leur adresse de destination ;
- de fragmenter et réassembler les datagrammes si nécessaire.

Le **protocole TCP** (*Transmission Control Protocol*) ou protocole de contrôle de la transmission préserve l'ordre des paquets. Pour cela TCP intègre des mécanismes :
- de détection d'erreurs ;
- de détection de perte ;
- de duplication de paquets ;
- de réémission automatique des paquets perdus.

Mais TCP ne garantit pas un délai précis. La durée d'envoi d'un paquet dépend des débits disponibles à l'instant du transfert.



On divise généralement les différentes tâches de communication en quatre couches superposées, chacune ayant un rôle spécifique :

- couche d'accès réseau / couche de liaison de données : elle est responsable de la transmission physique des données sur le réseau. Elle correspond aux câbles, aux cartes réseau et aux aspects matériels.
- couche internet : c'est ici que le protocole IP intervient. Son rôle est d'acheminer les paquets de données à travers différents réseaux jusqu'à leur destination finale, en utilisant une adresse IP unique.
- couche de transport : elle est divisée en deux protocoles (normes régissant la façon dont les données doivent être structurées et transmises à travers un réseau) principaux, TCP et UDP.
  - → TCP est utilisé pour des transmissions fiables et ordonnées
  - → UDP est utilisé pour des transmissions rapides et sans connexion
- couche application : elle contient tous les protocoles de haut niveau tels que HTTP[1] (pour le web), SMTP[2] (pour l'email), FTP[3] (pour le transfert de fichiers), etc.

DIFFÉRENCES ENTRE TCP ET UDP

TCP est un **protocole basé sur la connexion**, tandis que UDP fonctionne sans connexion établie.
TCP se caractérise par une plus grande **fiabilité**, mais avec un transfert de données **plus lent**.
UDP, bien que moins fiable, offre une rapidité supérieure.
TCP établit une voie de **communication** pour garantir la **transmission intégrale** et **fiable** de toutes les données. Lorsqu'un message est envoyé via TCP, sa **réception** est **systématiquement vérifiée** pour confirmer que l'intégralité des données a été correctement reçue.
UDP envoie des données sans établir de connexion préalable et sans confirmer leur réception ou vérifier l'existence d'éventuelles erreurs. En conséquence, cela signifie qu'une partie, voire la totalité, des données peut se perdre durant leur transmission.
Ainsi, chaque protocole est spécifiquement adapté à différents types de transfert de données.

	TCP	UDP
Connexion avant la transmission des données	Oui	Sans
Séquence les données	Oui : préserve l'ordre d'arrivée	Non
Retransmet les données en cas de perte	Oui	Non
Accusé de réception	Oui : livraison garantie	Non
Vérification des erreurs et correction	Oui	Non
Rapidité	Transmission moins rapide	Transmission plus rapide

TCP est donc particulièrement adapté à des applications telles que l'envoi d'**e-mails**, l'envoi de **SMS**, le **streaming de contenus préenregistrés** comme ceux disponibles sur Netflix, le **transfert de fichiers**, ou encore la **navigation sur internet**.

1 Hypertext Transfer Protocol
2 Simple Mail Transfer Protocol
3 File Transfer Protocol

UDP excelle dans des situations nécessitant une transmission rapide et en temps réel, comme le **streaming en direct**, les **jeux vidéo en ligne**, les **messageries instantanées**, les **conférences** vidéo, la **VoIP**[4] pour les appels vocaux via applications, ou encore pour le **système DNS**.

TCP présente un inconvénient notable : en cas de perte de données, il peut interrompre le chargement d'informations jusqu'à ce que les données manquantes soient récupérées et transmises. Cela peut se manifester, par exemple, lors du chargement d'une page web, où TCP n'affichera pas les images ou d'autres éléments de la page tant que l'ensemble des données nécessaires n'aura pas été complètement reçu.

TCP préserver l'ordre des paquets ? Voyons cela sur un exemple

Prenons l'exemple d'une donnée fragmentée en 4 paquets ordonnés : P1, P2, P3 et P4.

Supposons l'utilisation de TCP/IP et l'arrivée des paquets dans cet ordre : P3, P2, P1, P4.

Lorsque le paquet P3 arrive en premier, le destinataire TCP reconnaît qu'il s'agit d'un paquet hors séquence, car il s'attendait à recevoir P1 en premier : TCP met P3 en attente dans un tampon, car il sait qu'il fait partie d'une séquence de données mais n'est pas le premier paquet attendu.

Lorsque P2 arrive ensuite, TCP le place également dans le tampon, car P1, qui devrait être le premier paquet de la séquence, manque toujours.

Quand P1 arrive enfin, TCP l'accepte, puis vérifie son tampon pour voir s'il y a des paquets en attente qui peuvent maintenant être placés dans l'ordre.

TCP trouve P2 et P3 dans le tampon et les place dans l'ordre correct : P1, P2, P3.

Enfin, lorsque P4 arrive, il est placé dans la séquence, complétant ainsi la série de données : P1, P2, P3, P4.

Après chaque réception de séquence continue de paquets, TCP envoie un accusé de réception (ACK[5]). Dans ce cas, après avoir reçu P1 (et ayant P2 et P3 dans le tampon), il envoie un ACK pour P3. Cet ACK signifie que le destinataire a reçu toutes les données jusqu'à P3 inclus.

De même, après la réception de P4, il envoie un ACK pour P4, indiquant que les données jusqu'à P4 sont reçues.

Cette méthode « ACK cumulatif » permet de réduire le nombre de messages de contrôle sur le réseau, ce qui est une optimisation importante pour la performance du réseau.

COMPLÉMENTS (FACULTATIF)

Pour en savoir un peu plus sur le TCP/IP, vous pouvez regarder cette vidéo (≈ 15 min) :
youtu.be/_0thnFumSdA

4 Voice over IP

5 Abbréviation de *acknowledgement* qui signifie « accusé de réception ».