

TCP/IP

MODULE 04


**COUCHE
TRANSPORT**

Plan de la séance

Protocoles de la couche Transport

- TCP ;
- UDP.

Couche Transport



Couche Transport

Présentation

La couche transport remplit également d'autres fonctions ...

- Elle permet à de **nombreuses applications de communiquer sur le réseau au même moment**, sur un même périphérique ;
- Elle **vérifie**, si cela est nécessaire, **que toutes les données sont reçues de façon fiable et dans l'ordre** par l'application voulue ;
- Elle **utilise des mécanismes de gestion des erreurs**.

Couche Transport

Rôles de la couche transport

La couche transport segmente les données et se charge du contrôle nécessaire au réassemblage de ces blocs de données dans les divers flux de communication. Pour ce faire, il doit ...

- effectuer un suivi des communications individuelles entre les applications résidant sur les hôtes source et de destination ;
- segmenter les données et gérer chaque bloc individuel ;
- réassembler les segments en flux de données d'application ;
- identifier les différentes applications.

Couche Transport

Fonctions de la couche Transport

Tous les protocoles de la couche Transport ont des fonctions essentielles communes ...

Segmentation et reconstitution

La plupart des réseaux limitent la quantité de données pouvant être incluses dans une même unité de données de protocole.

La couche Transport **divise les données d'application en blocs de données d'une taille adéquate.**

Une fois ces blocs parvenus à destination, la couche Transport **réassemble les données avant de les envoyer vers l'application ou le service de destination.**

Couche Transport

Fonctions de la couche Transport

Tous les protocoles de la couche Transport ont des fonctions essentielles communes ...

Multiplexage de conversations

De nombreux services ou applications peuvent s'exécuter sur chaque hôte sur le réseau. Une **adresse, appelée port, est affectée à chacun de ces services ou applications afin que la couche transport puisse déterminer à quel service ou application les données se rapportent.**

Établissement d'une session

La couche Transport est en mesure d'**orienter la connexion en créant des sessions entre les applications.** Ces connexions préparent les applications à communiquer entre elles avant le transfert des données. Dans ces sessions, il est possible de **gérer avec précision les données d'une communication entre deux applications.**

Couche Transport

Fonctions de la couche Transport

Tous les protocoles de la couche Transport ont des fonctions essentielles communes ...

Acheminement fiable

Bien des circonstances peuvent entraîner la corruption ou la perte d'un bloc de données lors de son transfert sur le réseau. La couche transport veille à ce que **tous les blocs atteignent leur destination en demandant au périphérique source de retransmettre les données qui ont pu se perdre.**

Livraison dans un ordre défini

Étant donné que les réseaux fournissent une multitude de routes dont les délais de transmission varient, **il se peut que les données arrivent dans le désordre.**

En numérotant et en ordonnant les segments, la couche transport s'assure que ces **segments sont réassemblés dans le bon ordre.**

Couche Transport

Fonctions de la couche Transport

Tous les protocoles de la couche Transport ont des fonctions essentielles communes ...

Contrôle du flux

Les hôtes du réseau disposent de ressources limitées, par exemple en ce qui concerne la mémoire ou la bande passante. Quand la couche Transport **détermine que ces ressources sont surexploitées, certains protocoles peuvent demander à l'application qui envoie les données d'en réduire le flux.**

Ceci s'effectue au niveau de la couche Transport en régulant la **quantité de données que la source transmet sous forme de groupe.**

Le contrôle du flux contribue à **prévenir la perte de segments sur le réseau et à rendre inutiles les retransmissions.**

Couche Transport

Protocole TCP

TCP (qui signifie *Transmission Control Protocol*, soit en français: Protocole de Contrôle de Transmission) est un des principaux protocoles de la couche Transport du modèle TCP-IP. Il permet, au niveau des applications, de **gérer les données en provenance (ou à destination) de la couche inférieure du modèle** (c'est-à-dire le protocole IP).

Lorsque les données sont fournies au protocole IP, celui-ci les **encapsule dans des datagrammes IP, en fixant le champ protocole à 6** (pour savoir que le protocole en amont est TCP).

TCP est un **protocole orienté connexion**, c'est-à-dire qu'il permet à **deux hôtes qui communiquent de contrôler l'état de la transmission**.

Couche Transport

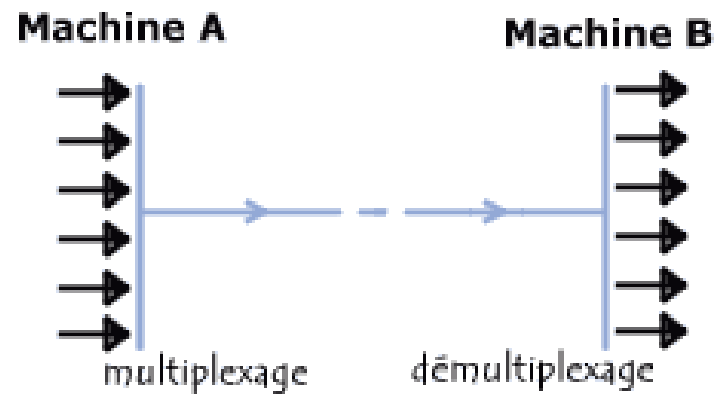
Fonction de multiplexage

TCP permet d'effectuer une tâche importante: le multiplexage/démultiplexage, c'est-à-dire faire **transiter sur une même ligne des données provenant d'applications diverses** ou en d'autres mots **mettre en série des informations arrivant en parallèle**.

Ces opérations sont **réalisées grâce au concept de ports** (ou sockets), c'est-à-dire un **numéro associé à un type d'application**, qui, **combiné à une adresse IP**, permet de **déterminer de façon unique une application** qui tourne sur une machine donnée.

Couche Transport

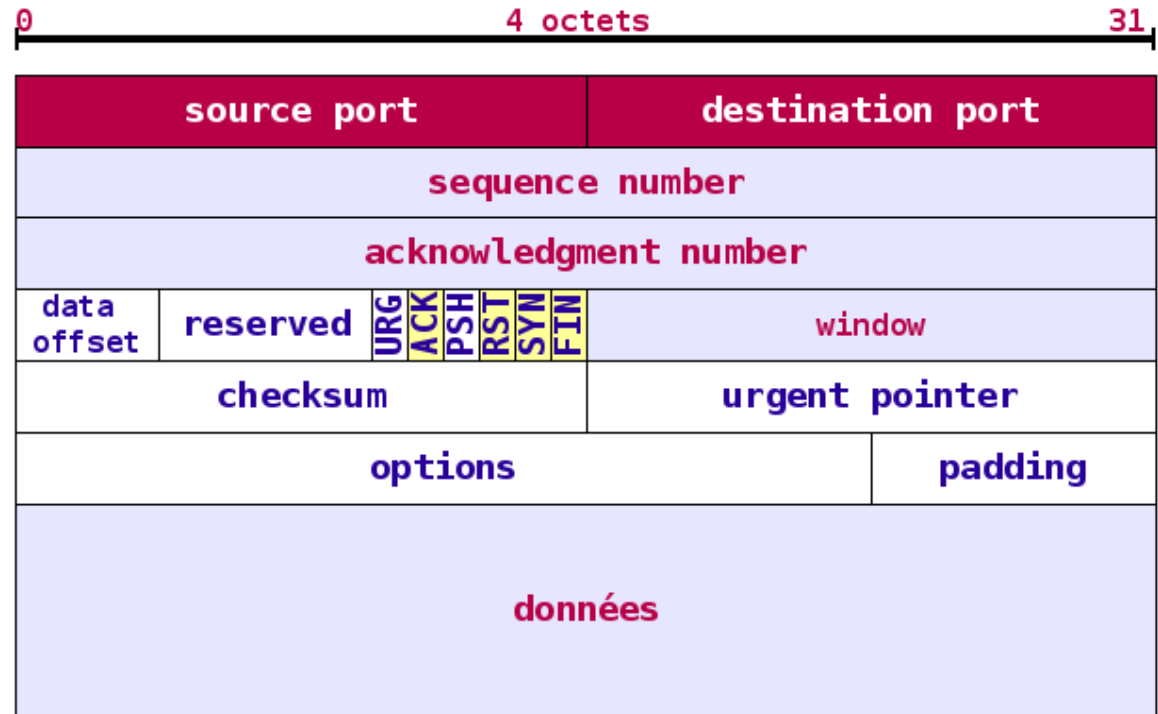
Fonction de multiplexage



Couche Transport

Format des données

Un segment TCP est constitué comme suit ...



Couche Transport

Format des données

Signification des différents champs ...

- **Port Source** (*Source port*) -- 16 bits
- **Port Destination** (*Destination port*) -- 16 bits
- **Numéro d'ordre** (*Sequence number*) -- 32 bits
- **Numéro d'accusé de réception** (*Acknowledgment number*) -- 32 bits
- **Décalage des données** (*Data offset*) -- 4 bits
- **Réservé** (*Reserved*) -- 6 bits
- **Drapeaux** (*flags*) (6x1 bit)
URG – ACK – PSH (*PUSH*) – RST (*RESET*) – SYN – FIN
- **Fenêtre** (*Window*) -- 16 bits
- **Somme de contrôle** (*checksum* ou **CRC**) -- 16 bits
- **Pointeur d'urgence** (*Urgent pointer*) -- 16 bits
- **Options** -- Taille variable
Bourrage (*Paddind*)
Fiabilité des transferts

Couche Transport

Format des données

En réalité, le protocole TCP possède un **système d'accusé de réception permettant au client et au serveur de s'assurer de la bonne réception mutuelle** des données.

Lors de l'émission d'un segment, un **numéro d'ordre (appelé aussi numéro de séquence) est associé**.

À la réception d'un segment de donnée, l'hôte **récepteur va retourner un segment de donnée dont l'indicateur ACK est à 1** (afin de signaler qu'il s'agit d'un accusé de réception) **accompagné d'un numéro d'accusé de réception égal au numéro d'ordre précédent**.

Couche Transport

Format des données

De plus, grâce à une **minuterie déclenchée** dès réception d'un segment au **niveau de l'hôte émetteur**, le segment est **réexpédié dès que le temps imparti est écoulé**, car dans ce cas l'hôte **émetteur considère que le segment est perdu**.

Toutefois, si le **segment n'est pas perdu** et qu'il arrive tout de même à destination, l'hôte **récepteur saura**, grâce au numéro d'ordre, qu'il s'agit d'un **doublon et ne conservera que le dernier segment arrivé à destination**.

Couche Transport

Établissement d'une connexion

L'établissement de la connexion entre deux applications se fait souvent selon le schéma suivant ...

- Les **ports TCP doivent être ouverts** ;
- L'**application sur le serveur est passive**, c'est-à-dire que l'**application est à l'écoute**, en **attente d'une connexion** ;
- L'**application sur le client fait une requête de connexion sur le serveur dont l'application est en ouverture passive**. L'**application du client** est dite en **ouverture active**.

Les **deux hôtes doivent donc synchroniser** leurs séquences grâce à un mécanisme communément appelé **poignée de main en trois temps** (*three ways handshake*) que l'on retrouve aussi lors de la clôture de session.

Couche Transport

Établissement d'une connexion

Ce dialogue permet d'initier la communication et se déroule en trois temps, comme sa dénomination l'indique :

- Dans un premier temps l'**hôte émetteur** (le client) **transmet un segment dont l'indicateur SYN est à 1** (pour signaler qu'il s'agit d'un segment de synchronisation), avec un **numéro d'ordre N, que l'on appelle numéro d'ordre initial du client** ;

Couche Transport

Établissement d'une connexion

- Dans un second temps l'**hôte récepteur** (le serveur) reçoit le **segment initial provenant du client**, puis lui envoie un **accusé de réception**, c'est-à-dire un segment dont l'**indicateur ACK est à 1** et l'**indicateur SYN est à 1** (car il s'agit là encore d'une synchronisation).
Ce segment contient le **numéro d'ordre de cet hôte** (du serveur) qui est le **numéro d'ordre initial du client**.
Le champ le plus important de ce segment est le **champ accusé de réception** qui contient le **numéro d'ordre initial du client, incrémenté de 1**.
- Enfin, le **client transmet au serveur un accusé de réception**, c'est-à-dire un segment dont l'indicateur ACK est à 1, **dont l'indicateur SYN est à 0** (il ne s'agit plus d'un segment de synchronisation).
Son **numéro d'ordre est incrémenté et le numéro d'accusé de réception représente le numéro d'ordre initial du serveur incrémenté de 1**.

Couche Transport

Méthode de la fenêtre glissante

Dans de nombreux cas, il est possible de **limiter le nombre d'accusés de réception**, afin de désengorger le réseau, en fixant un **nombre de séquence au bout duquel un accusé de réception est nécessaire**. Ce nombre est en fait stocké dans le **champ fenêtre** de l'en-tête TCP/IP.

On appelle effectivement cette **méthode de la fenêtre glissante** car on définit en quelque sorte une **fourchette de séquences n'ayant pas besoin d'accusé de réception**, et celle-ci se **déplace au fur et à mesure que les accusés de réception sont reçus**.

Couche Transport

Fin d'une connexion

Le client peut demander à mettre fin à une connexion au même titre que le serveur.

La fin de la connexion se fait de la manière suivante ...

- Un des hôtes **envoie un segment avec l'indicateur FIN à 1** et l'application se met en **état d'attente de fermeture de session**, c'est-à-dire qu'elle finit de recevoir le segment en cours et ignore les suivants
- Après réception de ce segment, l'autre **hôte envoie un accusé de réception avec l'indicateur FIN à 1** et **continue d'expédier les segments en cours**.

Suite à cela la **machine informe l'application qu'un segment FIN a été reçu**, puis **envoie un segment FIN à l'autre machine, ce qui clôture la connexion..**

Couche Transport

Protocole UDP

Le protocole UDP ou *User Datagram Protocol* est un des principaux protocoles de télécommunication utilisé par Internet. Il fait partie de la couche Transport de la pile de protocole TCP/IP: dans l'adaptation approximative de cette dernière au modèle OSI, il appartiendrait à la couche 4, comme TCP.

Il est **détailé dans la RFC 768**.

Couche Transport

Protocole UDP

Le rôle de ce protocole est de **permettre la transmission de paquets de manière très simple entre deux entités**, chacune étant définie par une **adresse IP et un numéro de port** (pour différencier différents utilisateurs sur le même hôte).

Contrairement au protocole TCP, il travaille en **mode non connecté**: il n'y a **pas de moyen de vérifier si tous les paquets envoyés sont bien arrivés à destination et dans quel ordre** (le séquençement peut cependant être assuré par un protocole réseau de couche inférieure).

Il n'est prévu **aucun contrôle de flux ni contrôle de congestion**.

C'est pour cela qu'il est souvent décrit comme étant un **protocole non fiable**.

En revanche, pour un paquet UDP donné, l'exactitude du **contenu des données est assurée grâce à une somme de contrôle** (*checksum*).

Couche Transport

Structure d'un segment UDP

L'**entête** (*header* en anglais) d'un segment UDP est bien **plus simple** que celui de TCP ...

Port Source (16 bits)	Port Destination (16 bits)
Longueur (16 bits)	Somme de contrôle (16 bits)
Données (longueur variable)	

Couche Transport

Structure d'un segment UDP

Il contient les 4 champs suivants ...

- **Port Source** (Source port) -- 16 bits
- **Port de Destination** (Destination port) -- 16 bits
- **Longueur** (Length) -- 16 bits
- **Somme de contrôle** (Checksum) -- 16 b

Couche Transport

Utilisation

Le protocole UDP est utilisé ...

- soit de **transmettre des données très rapidement** et où la **perte d'une partie de ces données n'a pas grande importance**,
- soit de **transmettre des petites quantités de données**, là où la connexion en trois phase de TCP serait trop lourde.

Couche Transport

Utilisation

Par exemple, dans le cas de la **transmission de la voix sur IP**, ce **n'est pas grave si l'un ou l'autre paquet se perd** (il existe des mécanismes de substitution des données manquantes), par contre la **rapidité de transmission est un critère primordial pour la qualité d'écoute**.

Exemples d'utilisation ...

- le **programme traceroute** ;
- les **protocoles DNS, TFTP, ...** ;
- les **jeux en réseau** ;
- la **diffusion** (*streaming*) :

Il est indispensable pour les **applications multimédias de par sa faible latence**.