

26 - Client-serveur : des sockets TCP aux requêtes HTTP

Prérequis : Modèle OSI + couches liaisons de données et réseaux

Niveau : L1

Motivation : Comprendre les couches réseaux hautes permettant à des développeurs de développer des applications simplement en s'appuyant sur les garanties offertes par les couches inférieures.

I) Introduction et rappels :

1.) Modes d'accès à l'information

Proposition 1 : Les différentes applications utilisant le réseau partagent des informations selon différents modèles, les principaux étant **client-serveur** et **pair-à-pair**.

Proposition 2 : Sur Internet, la plupart des communications utilisent le modèle **client-serveur** : une ou plusieurs machines clientes demandent des informations à une machine serveur qui les délivre.

Proposition 3 : Le modèle **pair-à-pair** (**peer-to-peer**), abrégé en P2P, identifie chaque machine à un client et un serveur : les utilisateurs forment un groupe informel au sein duquel chacun peut communiquer avec tous les autres.

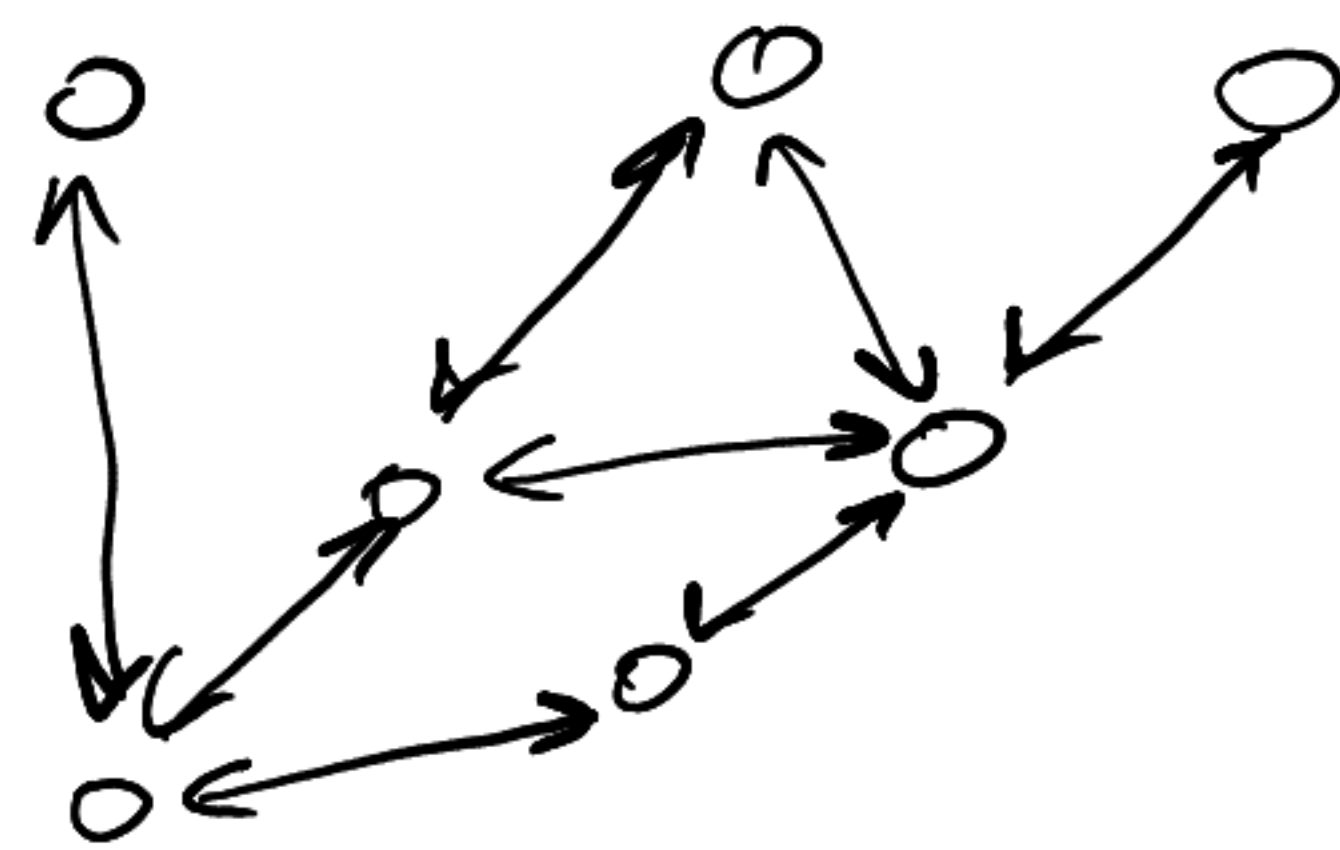
Exemples 4 :

- L'application WEB utilise le modèle client-serveur.
- L'application BitTorrent utilise le modèle pair-à-pair.

Schéma 5 : Comparaison des modèles d'accès à l'information



Modèle client-serveur



Modèle pair-à-pair

2.) Modèles en couches

Schéma 6 : Modèles OSI et TCP/IP

Application	Application
Présentation	
Session	
Transport	Transport
Réseau	Réseau
Liaison de données	Liaison de données
Physique	Physique

Modèle OSI

Modèle TCP/IP

II) Couche transport :

1.) Services fournis par la couche transport :

Proposition 10 : La vocation fondamentale de la couche transport est de fournir à la couche application un service de transfert de données efficace, fiable et économique.

Proposition 11 : Une communication est **avec connexion** (**connection-oriented**) si une session est établie avant l'envoi de données utiles. Elle assure alors l'arrivée dans le bon ordre des paquets transmis. Une communication est **sans connexion** (**connectionless**) où chaque paquet est remis indépendamment des autres.

Remarque 12 : Bien que les couches réseau et transport offrent des services qui semblent similaires, la distinction entre les deux est importante. La couche transport améliore la qualité de service en fournissant un service de transport plus fiable que la couche réseau.

2.) Primitives du service de transport

Proposition 13 : Chaque service de transport fournit une interface unique aux utilisateurs. Celle-ci s'appuie sur les primitives offertes par la couche réseau.

Activité 14 : Conception d'une application simple à partir de primitives pour un service de transport fictif.

Exemple 15 : Primitives d'un service de transport fictif

Primitive	Signification
LISTEN	Blocker jusqu'à ce qu'un processus tente de se connecter
CONNECT	Tenter activement d'établir une connexion
SEND	Envoyer des informations par un paquet "Data"
RECEIVED	Blocker jusqu'à ce que le paquet "Data" arrive
DISCONNECT	Libérer la connexion

3.) Éléments de protocoles de transport

Proposition 16 : Le service de transport est assuré par un **protocole de transport**. Ce dernier peut gérer, au même titre que les protocoles des couches inférieures, le contrôle d'erreur, le séquençage, le contrôle de débit...

Remarque 17 : Les deux protocoles de transport les plus utilisés sont **TCP** (Transmission Control Protocol) et **UDP** (User Datagram Protocol). Aujourd'hui, TCP représente ~30% du trafic Internet mondial.

Proposition 18 : Lors de l'établissement d'une connexion entre deux hôtes par un processus applicatif, un point terminal de transport doit être spécifié pour différencier les différents processus : on appelle ces point terminaux des **ports**.

Exemples 19: le protocole SSH utilise le port 22 et le protocole HTTP le port 80.

4.) TCP:

Proposition 20: **TCP** (**Transmission Control Protocol**) est un protocole de couche 4 (protocole de transport) conçu pour traiter de bout en bout des données de manière fiable sur un ensemble de réseaux non fiables.

Propriétés 21: TCP assure la bonne utilisation du réseau en évitant les congestions, la bonne réception du message et l'ordre dans lequel les messages sont reçus.

Proposition 22: Pour communiquer avec TCP, il faut créer deux points de connexions appelés **sockets**. Chaque socket est composé d'une adresse IP de l'ordinateur hôte et d'un port encodé sur 16 bits.

DEV: gestion de la connexion TCP

III) Couche applicative:

1.) Introduction

Proposition 23: Les couches applicatives (couches 5 à 7 du modèle OSI et 5 du modèle TCP/IP) fournissent aux utilisateurs des applications utilisant le réseau.

Remarque 24: Il existe beaucoup d'applications différentes avec pour chacune des protocoles associés, uniques ou non.

Exemples 25: Parmi les applications les plus utilisées sur Internet, on trouve le DNS permettant de résoudre les noms de domaine, ou encore le WEB.

2.) World Wide Web

Proposition 26: Le **Web** (ou **Toile** ou **World Wide Web**) est un cadre applicatif permettant d'échanger des contenus reliés au travers d'Internet sur des centaines de millions de machines. Il est basé sur le modèle client-serveur.

Proposition 27: Le **World Wide Web Consortium (W3C)** est une organisation dédiée au développement du Web, à la normalisation des protocoles et à la promotion de l'interopérabilité entre les sites.

Proposition 28: Le Web se présente sous la forme d'un ensemble de **pages (Web)** pouvant contenir des **liens hypertexte** vers d'autres pages. Les clients y accèdent par des **navigateurs internet**, communiquant par le protocole **HTTP** aux serveurs.

Application 29: Présentation d'une interaction Web entre un client avec telnet et un serveur avec apache2/nginx/...

DEV: Fonctionnement du protocole HTTP