

7: Couche applicative

Point de contact avec les services réseaux.

FTP, Telnet, SMTP, HTTP, ...

Elle fait office d'interface pour donner accès aux services réseaux (transfère de fichiers, rédaction de mail, établissement d'une session à distance, de visualisation de page web...)

Plusieurs protocoles assurent ces services, dont FTP (pour le transfert des fichiers), Telnet (pour l'établissement des sessions à distance), SMTP (pour l'envoi d'un mail)...

6: Présentation

Elle s'occupe de tout aspect lié à la présentation des données : format, cryptage, encodage, etc.

types **MIME**, **base64**, **charset** compression (gzip) et chiffrement SSL/TLS

Elle s'occupe de la sémantique, de la syntaxe, du cryptage/décryptage et de tout aspect « visuel » de l'information. Un des services de cette couche, entre autres : la conversion d'un fichier codé en EBCDIC (Extended Binary Coded Decimal Interchange Code) vers un fichier codé en ASCII.

Le cryptage peut être pris en charge par une autre couche que la couche de présentation. En effet, il peut s'effectuer dans la couche application, transport, session, et même réseau. Chaque niveau de cryptage a ses avantages.

5: Session

Responsable de l'initialisation de la session de communication, de sa gestion et de sa fermeture.

session, cookie, ...

Cette couche organise et synchronise les échanges entre tâches distantes. Elle réalise le lien entre les adresses logiques et les adresses physiques des tâches réparties. Elle établit également une liaison entre deux programmes d'application devant coopérer et commande leur dialogue (qui doit parler, qui parle...). La couche session permet aussi de resynchroniser l'échange en cas d'interruption

4: Transport

Choix du protocole de transmission et préparation de l'envoi des données. Elle spécifie le numéro de port utilisé par l'application réceptrice. Elle fragmente les données en plusieurs séquences (ou segments).

TCP, UDP

Une fois la session établie, l'information doit être envoyée. La couche de transport se charge de préparer l'information à l'envoi. Le nom de cette couche peut prêter à confusion: elle n'est pas responsable du transport des données proprement dit, mais elle y contribue. En fait, ce sont les quatre dernières couches (transport, réseau, liaison de données et physique) qui toutes ensemble réalisent le transport des données. Cependant, chaque couche se spécialise. La couche de transport divise les données en plusieurs segments (ou séquences) et les réunit dans la couche transport de l'hôte récepteur. Cette couche permet de choisir, en fonction des contraintes de communication, la meilleure façon d'envoyer une information. «Devrai-je m'assurer que la transmission a réussi, ou devrai-je juste l'envoyer et espérer que tout se passe bien? Quel port devrai-je utiliser?» La couche de transport modifie également l'en-tête des données en y ajoutant plusieurs informations, parmi lesquelles les numéros de ports de la source et de la destination. Le protocole TCP (Transmission Control Protocol) est le plus utilisé dans la couche de transport.

3: Réseau

Connexion logique entre les hôtes. Elle traite de tout ce qui concerne l'identification et le routage dans le réseau.

IP

3: Réseau (cont)

Maintenant que nous savons quel numéro de port utiliser, il faut aussi préciser l'adresse IP du récepteur. La couche réseau se charge du routage (ou relai) des données du point A au point B et de l'adressage. Ici aussi, l'en-tête subit une modification. Il comprend désormais l'en-tête ajouté par la couche de transport, l'adresse IP source et l'adresse IP du destinataire. Se fait également dans cette couche le choix du mode de transport. Le protocole le plus utilisé à ce niveau est le protocole IP.

2: Liaison

Etablissement d'une liaison physique entre les hôtes. Fragmente les données en plusieurs trames.

Ethernet, PPP (Point to Point Protocol), HDLC (High-Level Data Link Control), etc...

Transforme la couche physique en une liaison à priori exempte d'erreur de transmission pour la couche réseau. Elle fractionne les données en trames et les transmet au destinataire. La couche liaison doit donc être capable de reconnaître les frontières des trames. Elle doit également être capable de renvoyer une trame pour signaler un problème. Une autre fonction de la couche liaison est le contrôle de flux, c'est-à-dire d'éviter d'inonder un récepteur plus lent que l'émetteur.

1: Physique

Conversion des trames en bits et transmission physique des informations.

Interface, câblage, signaux

S'occupe de la transmission en données binaires de façon brute sur un circuit de communication. Les tensions transmises doivent être suffisamment claires pour reconnaître un 1 ou un 0. Elle définit si les données peuvent être bidirectionnelles ou pas, le nombre et le rôle de chaque broche du connecteur. Elle ne connaît pas la signification des données transmises.

Modèle de référence OSI

Open Systems Interconnection. Ce modèle a été mis en place par l'ISO afin de mettre en place un standard de communications entre les ordinateurs d'un réseau, c'est-à-dire les règles qui gèrent les communications entre des ordinateurs.

Chaque couche du modèle communique avec une couche adjacente. Chaque couche utilise ainsi les services des couches inférieures et en fournit à celle de niveau supérieur.

But:

Assurer l'interopérabilité entre instances et entre couches

Permettre le développement de protocoles s'intégrant dans une hiérarchie cohérente. Sans la définition des couches et de leur rôle, ceci n'est pas possible. C'est un cadre pour la définition de

Similarité des couches 2 et 4

Elles ont des rôles semblables: assurer un transport sûr des données.

La couche 2 relie deux appareils (via une ligne, un bus, un "réseau").

la couche 4 relie deux applications (via un réseau).

Portée des couches

Noeuds: Contiennent que les couches 1,2,3.

Terminaux: Contiennent toutes les couches.

Protocoles des couches 1 et 2 Peuvent varier d'un tronçon à un autre.

Le protocole de la couche 3 Doit être uniforme dans tout le réseau (sauf en présence de plusieurs couches 3 différentes interconnectées par des passerelles en couche 7 p.ex. via un proxy).

*En pratique, on retrouvera des adresses à portée **locale** dans les couches 2 et des adresses à portée **globale** dans la couche 3.*

Interaction entre les couches

Instance A		Instance B	
↓	Protocoles	↑	
Application	🗨️ ↔ 🗨️	Application	
📄 ↓	APDU	📄 ↑	
Présentation	🗨️ ↔ 🗨️	Présentation	
📄 ↓	PPDU	📄 ↑	
Session	🗨️ ↔ 🗨️	Session	
📄 ↓	SPDU	📄 ↑	
Transport	🗨️ ↔ 🗨️	Transport	
📄 ↓	TPDU	📄 ↑	
Réseau	🗨️ ↔ 🗨️	Réseau	
✉️ ↓	Paquet	✉️ ↑	
Liaison	🗨️ ↔ 🗨️	Liaison	
✉️ ↓	Trame	✉️ ↑	
Physique	0101 → bit →	Physique	

Construction des enveloppes	Traitement des entêtes
-----------------------------	------------------------

🗨️ ↔ 🗨️ *communication logique entre instances (Messages)*

0101 → bit *communication réelle (physique)*

Chaque couche physique est séparée en (au moins) deux parties appelées **instances**.

Les instances collaborent pour la réalisation des services de la couche grâce à des **PDU** (Protocol Data Unit).

Interaction entre les couches (cont)

Afin de pouvoir réaliser les services de la couche pour l'information transmise, les instances de chaque couche doivent ajouter au message des informations destinées à leur instance partenaire. Chaque couche ajoute donc son en-tête et parfois sont post-tête (couche 2) aux données de la couche précédente.

Principe de l'interaction

Chaque couche fournit des services à la couche du dessus et utilise les services de la couche en dessous. La communication se fait via les interfaces à travers un point d'accès dit **SAP**(Service Acces Point).

Chaque couche reçoit de la couche supérieure des données opaques (la couche liaison ne peut pas lire les adresses IP indiquées dans les données venant de la couche de réseau IP) qu'elle doit transférer à la couche inférieure (dans le cas d'une émission).

Chaque couche ajoute à la donnée opaque qui est le **SDU** (Donnée effective dans un **PDU**) des informations de contrôle de protocole dit **PCI** pour obtenir un **PDU**.

Lors de l'émission d'une information, chaque couche N du modèle OSI encapsule les données qu'elle reçoit de la couche supérieure N+1 en lui rajoutant des informations de contrôle relatives à la couche N (entête de protocole ou PCI). Cette encapsulation au niveau d'une couche N est donc la concaténation du SDU et d'un entête (et/ou en queue) pour former l'unité de données de protocole PDU qui deviendra le SDU de la couche inférieure.

Lors de la réception d'une information, chaque couche N dés-encapsule le PDU (reçut par la couche inférieur N-1) pour séparer l'information de protocole du SDU qui sera envoyée à la couche supérieure N+1

Protocoles

Méthode standard qui permet la communication entre des processus (s'exécutant éventuellement sur différentes machines)

Ensemble de règles et de procédures à respecter pour émettre et recevoir des données sur un réseau.

De plusieurs types selon ce que l'on attend de la communication. Certains protocoles seront spécialisés dans l'échange de fichiers, d'autres servent à gérer l'état de la transmission et des erreurs...

Protocole avec ou sans connexion :

Protocoles **orientés** connexion:

Opèrent un contrôle de transmission des données pendant une communication établie entre deux machines. La machine réceptrice envoie des accusés de réception lors de la communication, ainsi la machine émettrice est garante de la validité des données qu'elle envoie. Les données sont envoyées sous forme de flot. **TCP** est un protocole **orienté**.

Protocoles **non orientés** connexion:

Mode de communication dans lequel la machine émettrice envoie des données sans prévenir la machine réceptrice et la machine réceptrice reçoit les données sans envoyer d'accusé de réception à la première. Les données sont ainsi envoyées sous forme de blocs (datagrammes). **UDP** est un protocole **non orienté** connexion.

Services

Un **service** est une unité de travail qu'une couche supérieure peut demander à une couche inférieure:

- établir une connexion avec son instance distante

- transporter des données jusqu'à cette instance distante

- etc ...

Services (cont)

Nécessite plusieurs interactions entre la couche inférieure et la couche supérieure. Lorsque le service n'est pas confirmé, on a deux interactions:

- la demande d'un côté

- la confirmation d'exécution de l'autre côté

Si le service est confirmé on a alors 4 interactions.

Ces interactions peuvent être organisées sous forme de 4 classes de service:

REQUEST, INDICATION, RESPONSE, CONFIRM

On parle aussi de primitives. Un transfert confirmé de donnée par la couche transport nécessitera les primitives suivantes:

T-DATA-REQUEST, T-DAT-INDICATION, T-DAT-RESPONSE, T-DAT-CONFIRM.

Service/protocole

La séparation "service/protocole" permet de cacher le fonctionnement interne d'une couche aux couches adjacentes.

Les services représentent les communications / échanges entre couches (une couche peut fournir ou utiliser un service) d'une même entité (instance), alors que le protocole permet d'échanger des informations pour la communication entre couches homologues dans des entités (instances) différentes.

Mécanismes de Fractionnement

Opérations entre couches/niveaux

Segmentation/réassemblage:

Il s'agit du découpage d'un PDU(N+1) en plusieurs SDUs(N), ou encore de rendre le service (à la couche supérieure) peuvent utiliser plusieurs PDUs(N) pour traiter une SDU(N).

Groupage/dégroupage:

Il est également possible de faire correspondre plusieurs SDUs de couche N avec une seule PDU de couche N.