Les modèles de référence OSI et TCP/IP

|  |  |
| --- | --- |
| **Objectifs** | * Comprendre les modèles en couches, * Comprendre le rôle de chaque couche, * Connaître les unités de données gérées par chaque couche, * Connaître les protocoles fonctionnant au niveau de chaque couche, * Connaître les matériels fonctionnant au niveau de chaque couche. |
| **Mots-clés** | OSI, TCP, IP, encapsulation, décapsulation, applications, client, serveur, service, port, http, ftp, smtp, pop, dhcp, dns, … |

#### La modélisation en couche

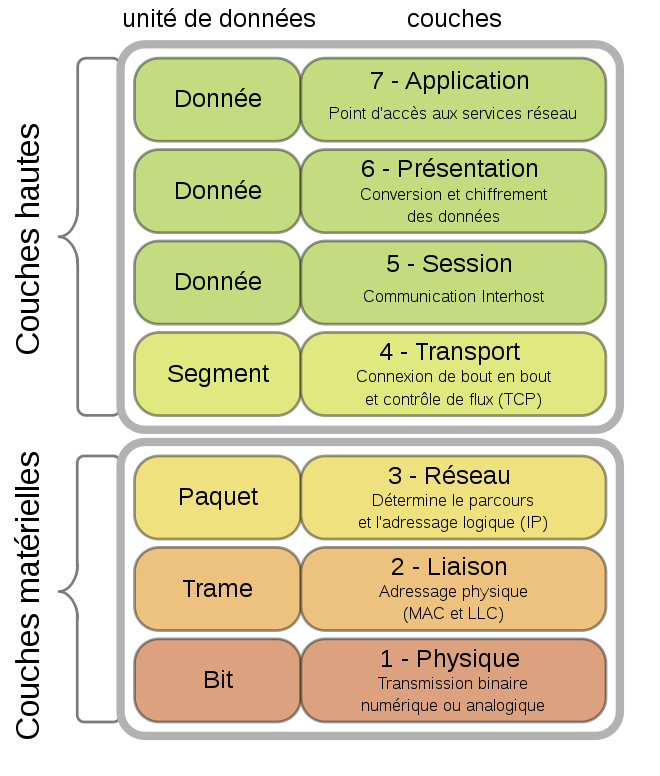
##### La nécessité et l’utilité d’un modèle ouvert

Les éditeurs informatiques avaient depuis longtemps leurs propres réseaux pour interconnecter leurs équipements, comme l’architecture SNA (System Network Architecture) d’IBM, DSA de CII-Honeywell-Bull, Decnet, etc. Mais il s'est vite avéré **qu'il serait impossible d'interconnecter ces différents réseaux « propriétaires »** si une norme internationale n'était pas établie.

Ainsi, afin d’éviter la multiplication des solutions d'interconnexion, l'ISO (International Standards Organisation) a développé dans les années 1970 **un modèle de référence à 7 couches appelé modèle OSI** (Open Systems Interconnection). L’objectif est de fournir un cadre commun pour **développer des systèmes ouverts qui ne dépendraient pas de systèmes propriétaires.**

En parallèle du modèle OSI, un autre modèle appelé **TCP/IP** (appelé aussi modèle Internet) a été développé le département de la défense américain. Ce modèle est basé uniquement sur **4 couches**.

##### Le modèle OSI (Open System Interconnection)



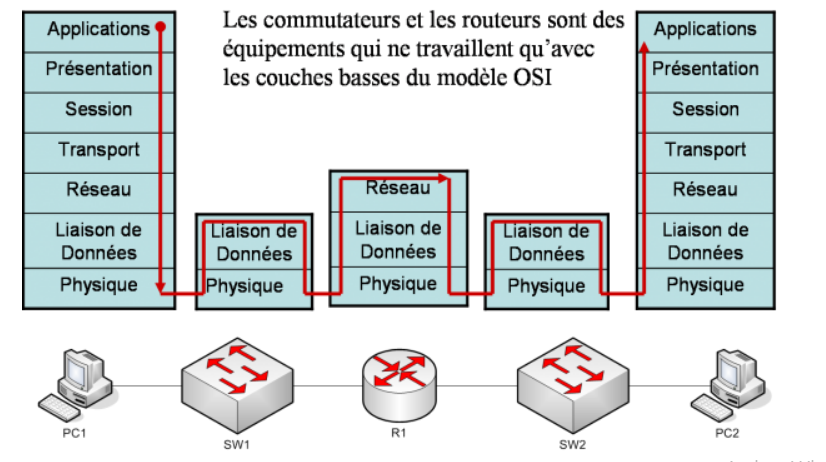
Ce modèle établi par l'ISO, dans les années 1970, définit un modèle de référence permettant l'interconnexion de systèmes ouverts.

* Il se décline en **7 parties appelées couches**.
* Chaque couche est responsable de l’un des aspects de la communication (une fonction).
* Chaque couche gère un PDU (unité de données) spécifique : donnée, segment, paquet, trame ou bit.

**Aperçu des rôles de chaque couche du modèle OSI :**

|  |  |
| --- | --- |
| **Couche** | **Rôles** |
| 7- Application | Fourniture de services réseaux aux applications.  C’est le lieu d’entrée et de sortie des données échangées. |
| 6- Présentation | Formatage (mise en forme), conversion, compression et cryptage des données. |
| 5- Session | Établissement, contrôle, terminaison d'une connexion entre 2 systèmes. |
| 4- Transport | * Contrôle des flux : réception ou non des segments, * Segmentation (découpage) et réassemblage des segments selon la taille permise sur le canal de transmission, * Numérotation des segments, * Identification des applications grâce aux numéros de ports. |
| 3- Réseau | - Adressage logique (IP),  - Routage des paquets. |
| 2- Liaison de données | - Détection des erreurs,  - Contrôler comment les données accèdent au média physique,  - Structuration des données en **trames** Ethernet,  - Adressage physique (MAC) |
| 1- Physique | Définit la façon dont les données sont converties en signaux numériques selon le support de transmission :  - électrique : pour les câbles Ethernet en cuivre,  - lumière : pour la fibre optique,  - onde radio : pour le WiFi. |

**Matériels réseaux associés aux couches du modèle OSI :**

****

Récepteur

Commutateur

Routeur

Commutateur

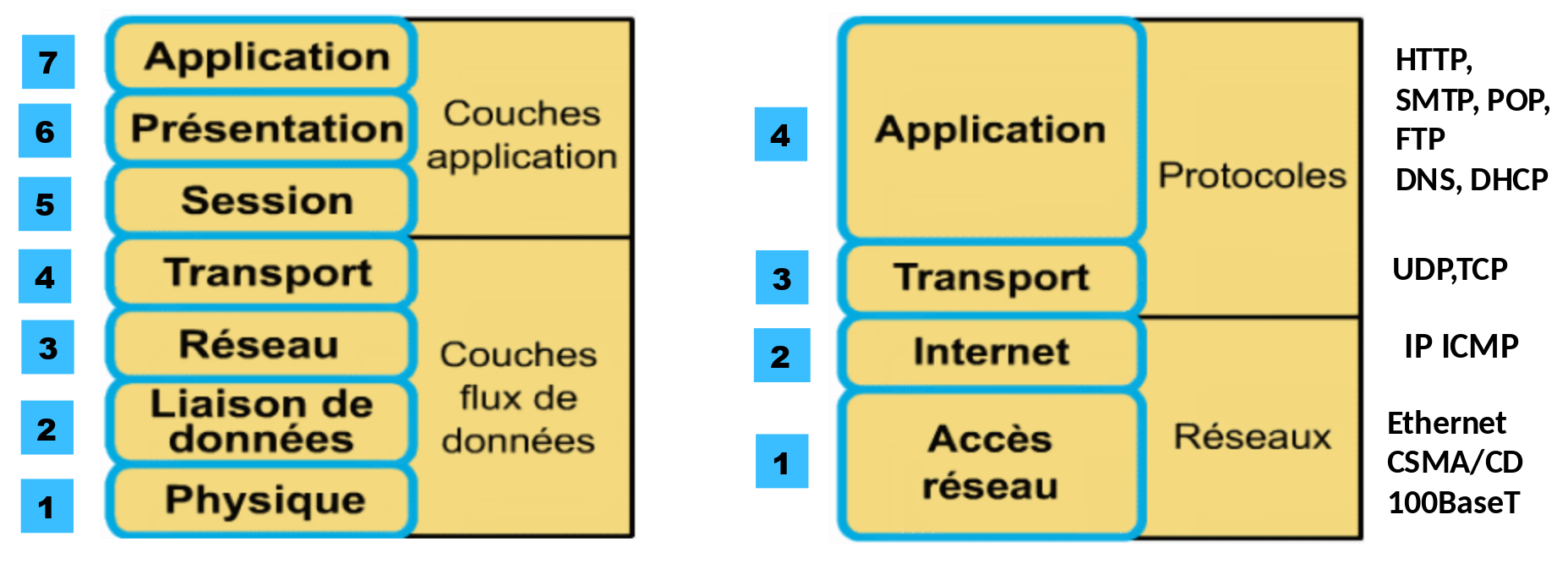
Émetteur

* Les routeurs gèrent les données (paquets) de la couche 3 du modèle OSI,
* Les commutateurs et les cartes réseau gèrent les données (trames) de la couche 2 du modèle OSI,
* Le médias de transmission (câblage) gèrent les données (bits) de la couche 1 du modèle OSI.

##### Le modèle TCP/IP

* **Regardez la vidéo** [**https://www.youtube.com/watch?v=\_0thnFumSdA**](https://www.youtube.com/watch?v=_0thnFumSdA) **.**

L'architecture OSI est une architecture ouverte qui permet de s'adapter à différents contextes, mais elle est complexe à implémenter. Elle n'a donc pas pu s'imposer vraiment en rivale des autres architectures. Ainsi il est plus courant de trouver les architectures basées sur le modèle TCP/IP.



**Le modèle TCP/IP est basé sur 4 couches :**

**Modèle OSI Modèle TCP/IP**

**Exemples de protocoles**

**utilisés dans le modèle TCP/IP : Rôles de chaque couche :**

**HTTP – FTP**

**DNS – DHCP**

**Couche 4**appelée **Application :**

\* Construit les **données.**

\* Définit le format des requêtes et des réponses échangées selon l'application.

**Couche 3**appelée **Transport** :

\* Découpe les données à envoyer en **segments** de petite taille.

\* Permet de repérer précisément le service utilisé par le biais du numéro de port.

\* Permet de transporter les données de la couche application de façon fiable ou non (selon le protocole utilisé).

**TCP/UDP**

**IP**

**Couche 2** appelée **Inter-réseau ou Internet**:

\* Permet de repérer l’hôte source et l’hôte destination par le biais d’adresses IP et de les ajouter au segment afin de construire le **paquet**.

\* Permet de router les paquets entre les réseaux.

**ETHERNET**

**Couche 1** appelée **Accès réseau :**

\* Permet de connaître les adresses physiques et de fabriquer la **trame**.

\* Assure la bonne gestion du média.

 \* Permet l'acheminement des informations sous forme binaire entre émetteur et destinataire.

**Question : Quels sont les éléments matériels que l’on peut associer respectivement aux couches 1 et 2 du modèle TCP/IP ?**

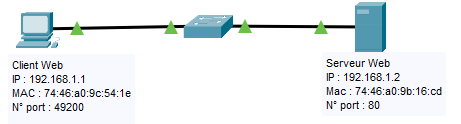
**- Couche 1 : RJ45, câbles Ethernet, ondes radio, cartes réseau, commutateurs (Switch), fibre optique.**

**- Couche 2 : routeurs, Switch niveau L3**

#### Modèle et protocoles en couches : le processus d'encapsulation des données

Lorsque les **données de la couche application** descendent la pile de protocoles en vue de leur transmission sur le support réseau, les différents protocoles ajoutent des informations à chaque niveau. Le résultat est appelé un **PDU** (Protocol Data Unit ou unité de donnée de protocole).

Exemple : interaction entre un client Web (navigateur) et un serveur Web



a) Sur le poste émetteur : encapsulation des données

**1.** L’utilisateur lance son navigateur et saisit une URL : *http://192.168.1.2/coucou.html*. **Le navigateur passe l’ordre d’envoi d’une requête sur l'adresse IP à TCP**.

Données

Données

Données

TCP

TCP

**Segment**

IP

**Paquet**

**Trame**

**Protocole HTTP**

**Protocole TCP**

**Protocole IP**

Données

TCP

IP

Ethernet

**Protocole Ethernet**

**2.** Le protocole TCP divise les **données** en parties de plus petite taille, appelées **segments**, pour les envoyer au serveur. Il ajoute également aux données transmises des informations essentielles comme les **numéros de ports**.

**3.** Le **protocole IP** récupère les segments formatés à partir du protocole TCP, s'occupe de leur encapsulation en **paquets**, de l’affectation des adresses IP appropriées et de la sélection du meilleur chemin vers l’hôte de destination.

**4.** Les protocoles en jeu ici prennent les paquets depuis le protocole IP et les formatent pour les transmettre à travers les supports sous forme de **trames** qui contiennent les adresses MAC source et destination.

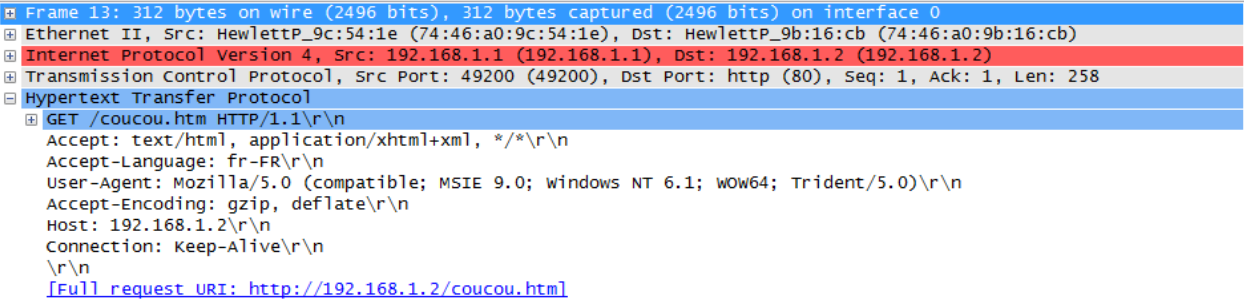
Transmission des données sous forme de bits sur le support de transmission.

La **trame** arrive donc en bout de processus et intègre l'ensemble des données y compris les numéros de port ajoutés par la couche transport, les adresses IP ajoutées par la couche IP et les adresses MAC ajoutées par la couche 2 :

Entête de la trame Ethernet du client vers le serveur :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Adresse MAC destination** | **Adresse MAC source** | **Adresse IP source** | **Adresse IP destination** | **Port source** | **Port**  **Dest** |
| 74:46:a0:9b:16:cd | 74:46:a0:9c:54:1e | 192.168.1.1 | 192.168.1.2 | 49200 | 80 |

On visualise, dans l’analyse de trame (Frame 13), les 4 couches (Ethernet, IP, TCP et HTTP) :



**Données**

**b) Sur le poste destinataire : dés-encapsulation des données**

Lorsque la trame finale est récupérée par l'hôte de destination (après qu’éventuellement le paquet IP soit passé par quelques routeurs) un **processus inverse** est mis en œuvre.

Données

Données

Données

TCP

TCP

**Segment**

IP

**Paquet**

**Trame**

**Protocole HTTP**

**Protocole TCP**

**Protocole IP**

Données

TCP

IP

Ethernet

**Protocole Ethernet**

**4.** Le protocole HTTP récupère les **données** (en l'espèce, *l'URL http://192.168.1.2/coucou.html* avec les informations prévues par le protocole)

**3.** Le protocole TCP récupère le **segment** et lit le numéro de port (port 80). Il supprime l'en-tête TCP et fait appel à l’application serveur Web.

**2.** Les protocoles IP récupère le **paquet** et supprime l'en-tête IP. Le segment obtenu est remonté vers le protocole TCP

**1.** Les protocoles en jeu récupèrent la **trame** depuis le support et supprime l'en-tête de trame. Le paquet obtenu est remonté vers le protocole IP

Sur les 2 machines nous avons la même pile de protocoles, chacun jouant un rôle précis (simplifié dans les explications précédentes). Chaque protocole de niveau supérieur demande un service au protocole de niveau inférieur et fournit des informations au protocole de même niveau sur la machine distante.

L'ensemble de ce dialogue à distance entre protocoles est véhiculé par les trames Ethernet dans son champ données, plus précisément ce champ contient les données IP, les données IP contiennent les données TCP, les données TCP contiennent les données HTTP, et les données HTTP contiennent les données (en l'espèce, l'URL). **Ce mécanisme de poupée russe s'appelle l'encapsulation.**

**Une fois la requête reçue et interprétée, le serveur répond en envoyant la page demandée (ou un message d’erreur si, par exemple, la page n’existe pas).**

**Reconstitution de la trame simplifiée de la réponse du serveur Web au client.**

Compléter les informations de l’entête de la trame :

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Adresse MAC destination** | **Adresse MAC source** | **Adresse IP source** | **Adresse IP destination** | **Port source** | **Port**  **Dest** |
| 74:46:a0:9c:54:1e | 74:46:a0:9b:16:cd | 192.168.1.1 | 192.168.1.2 | 80 | 49200 |

Récapitulatif : encapsulation/dés-encapsulation (modèle TCP/IP)



Données

Données

TCP

IP

Ethernet

**Trame**

**Application**

**Transport**

**Internet**

**Accès réseau**

Données

TCP

**Paquet**

**Application**

**Transport**

**Internet**

**Accès réseau**

**Transfert sur le réseau**

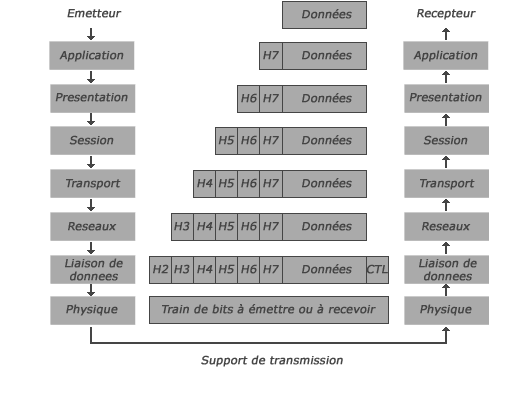
**Segments**

IP

TCP

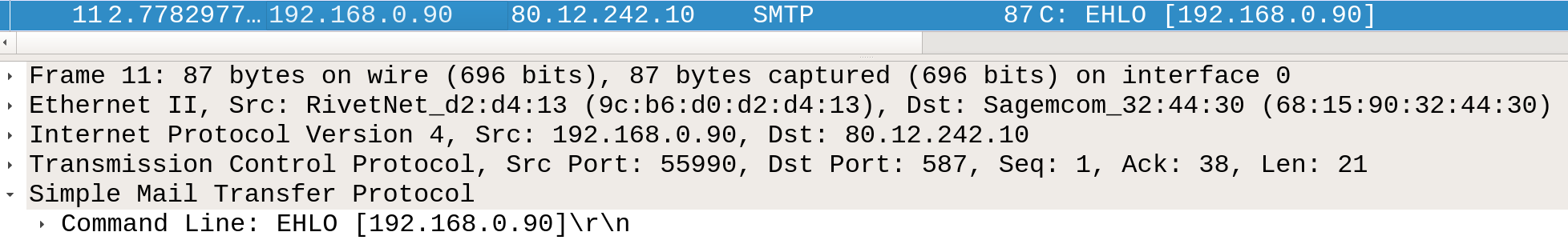
Données

Récapitulatif : encapsulation/dés-encapsulation (modèle OSI)



Exercice

Observez la capture de trame ci-dessous.



* 1. Quel est l’objet de cette communication ?

Transfert de mail entre une client et un serveur

* 1. Complétez le tableau ci-dessous qui associe à chaque couche du modèle TCP/IP le protocole correspondant à la trame émise.

|  |  |
| --- | --- |
| **Couches du modèle TCP/IP** | **Protocole associé** |
| 4 - Application | Application SMTP (Simple Mail Transfer Protocol |
| 3 - Transport | TCP (Transmission Control Protocol) : Port source 49200 / Port de destination : 80 |
| 2 - Internet | IPv4 : IP source 192.168.0.90 (privé) IP destination 80.12.242.10 |
| 1 – Accès réseau | Ethernet II : Adresse MAC source : 9c:b6:d0:d2:d4:d13  Adresse MAC destination : 68:15:90:32:44:30 |

* 1. Compléter la trame simplifiée correspondante à cette analyse de trame.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Adresse MAC destination** | **Adresse MAC source** | **Adresse IP source** | **Adresse IP destination** | **Port source** | **Port**  **Dest** |
|  |  |  |  |  |  |

* 1. En déduire la trame simplifiée de la réponse à la trame précédente.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Adresse MAC destination** | **Adresse MAC source** | **Adresse IP source** | **Adresse IP destination** | **Port source** | **Port**  **Dest** |
|  |  |  |  |  |  |