

Comprendre comment les données circulent sur un réseau

Une exploration structurée des modèles fondamentaux qui régissent les communications numériques modernes.

Objectifs Pédagogiques



Comprendre les grands concepts réseau



Savoir expliquer le modèle OSI



Comprendre le mécanisme d'encapsulation



Identifier les en-têtes réseau (IP / TCP / HTTP)

LE MODÈLE OSI : VUE GLOBALE

Open Systems Interconnection

Définition

Le modèle OSI est un modèle théorique standardisé qui décrit comment les données circulent sur un réseau en 7 couches successives.

OBJECTIFS DU MODÈLE

 Comprendre

 Standardiser

 Diagnostiquer

7 Application



6 Présentation



5 Session



4 Transport



3 Réseau



2 Liaison de données



1 Physique



Couches H

Proches de l'utilisateur

Cœur

Couches B

Proches du (Réseau)

LES 7 COUCHES OSI

Détail des rôles et responsabilités

N°	COUCHE	RÔLE PRINCIPAL
	 Application	Interaction directe avec l'utilisateur (HTTP, FTP, DNS)
	 Présentation	Formatage des données, chiffrement, compression
	 Session	Ouverture, gestion et fermeture des sessions
	 Transport	Fiabilité, segmentation, gestion des ports (TCP/UDP)
	 Réseau	Adressage logique (IP) et routage des paquets
	 Liaison	Adressage physique (MAC), trames, détection d'erreurs
	 Physique	Transmission binaire, câbles, signaux électriques/optiques



OSI : APPROCHE SIMPLIFIÉE

Vision fonctionnelle à retenir



LES APPLICATIONS



Ce que l'utilisateur voit et utilise.

Couche 7 Couche 6 Couche 5



LE TRANSPORT



Comment les données voyagent (fiabilité).

Couche 4



LE ROUTAGE



Le chemin à travers le réseau (adresse IP).

Couche 3



LE MATÉRIEL



Câbles, signaux, cartes réseau, switchs.

Couche 2 Couche 1

Version Pédagogique

Couches Hautes (7-5) : Les logiciels & données

Couche 4 : La gestion du transport

Couche 3 : L'adressage

Couches Basses (2-1) : Le "tuyau" physique



QU'EST-CE QUE L'ENCAPSULATION ?

Le principe d'empilement des protocoles

Définition

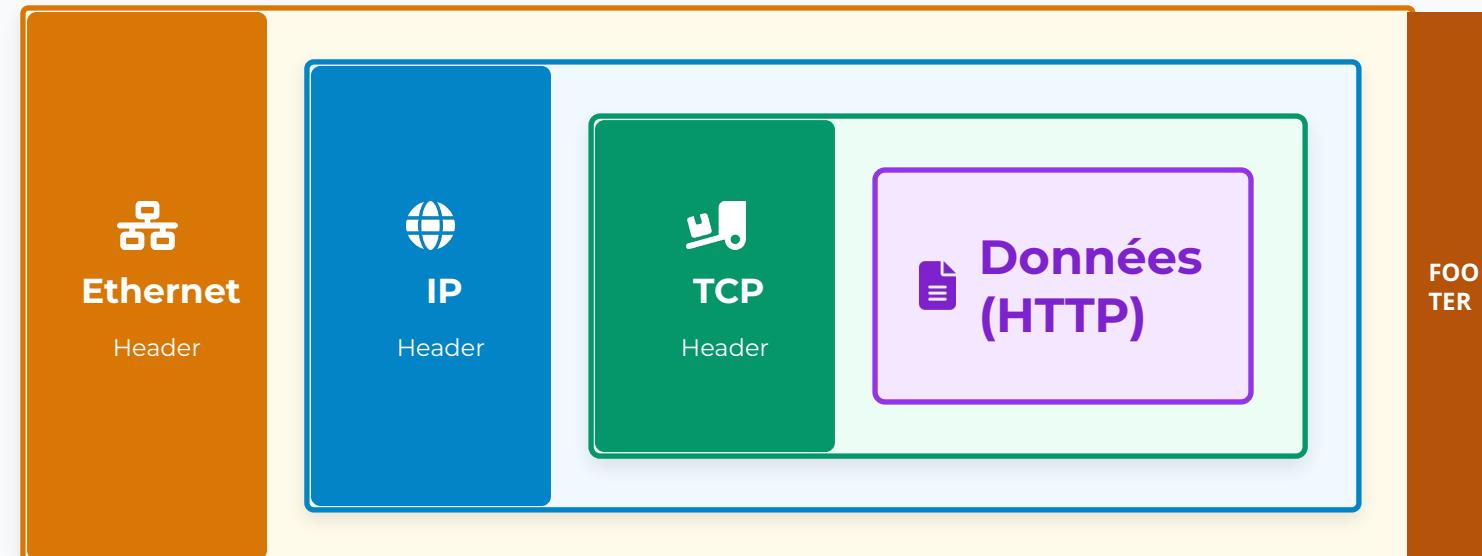
“L'encapsulation est le fait de prendre une donnée et de l'envelopper successivement à chaque couche réseau.”

CE QUI SE PASSE RÉELLEMENT :

+ Chaque couche ajoute un Header

↓ La donnée grossit en descendant

→ Parfois un Footer est ajouté



ANALOGIE : Lettre → Enveloppe → Colis → Camion

ENCAPSULATION : VISION PAS À PAS

Processus d'ajout des en-têtes lors de l'envoi

Exemple : "Je charge une page web"

1 La donnée HTTP est créée

APP

2 Encapsulée dans un segment TCP

TRANS

3 Encapsulée dans un paquet IP

RÉSEAU

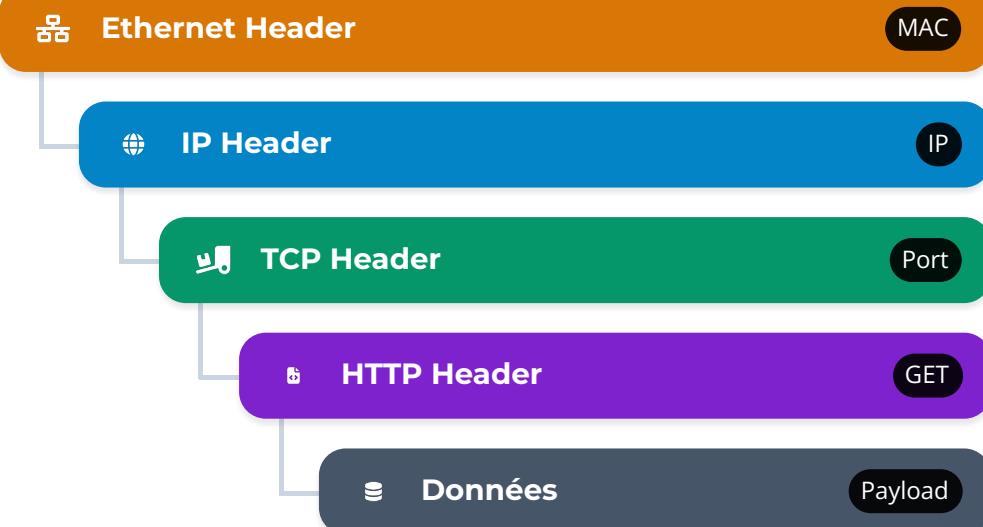
4 Encapsulée dans une trame Ethernet

LIAIS.

5 Envoyée sur le câble (Bits)

PHYS.

Structure Logique (Pile)



Chaque couche considère tout ce qui vient du dessus comme une simple "donnée" à transporter (Payload).

QU'EST-CE QU'UN HEADER ?

Comprendre la structure d'un paquet

❶ Définition

Un header (en-tête) est une zone d'information de contrôle placée avant la donnée utile (payload).

IL CONTIENT 3 TYPES D'INFOS :



La destination

Où envoyer le message (Adresse IP, MAC...)



Le traitement

Comment lire la donnée (Encodage, Protocole)

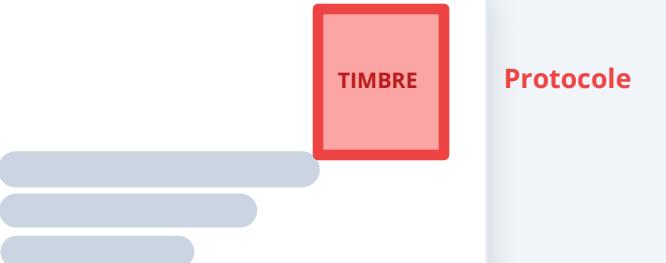


Le destinataire

À qui remettre le message (Port, Application)

L'ANALOGIE DE L'ENVELOPPE

Expéditeur :



HEADER

Instructions

DONNÉES

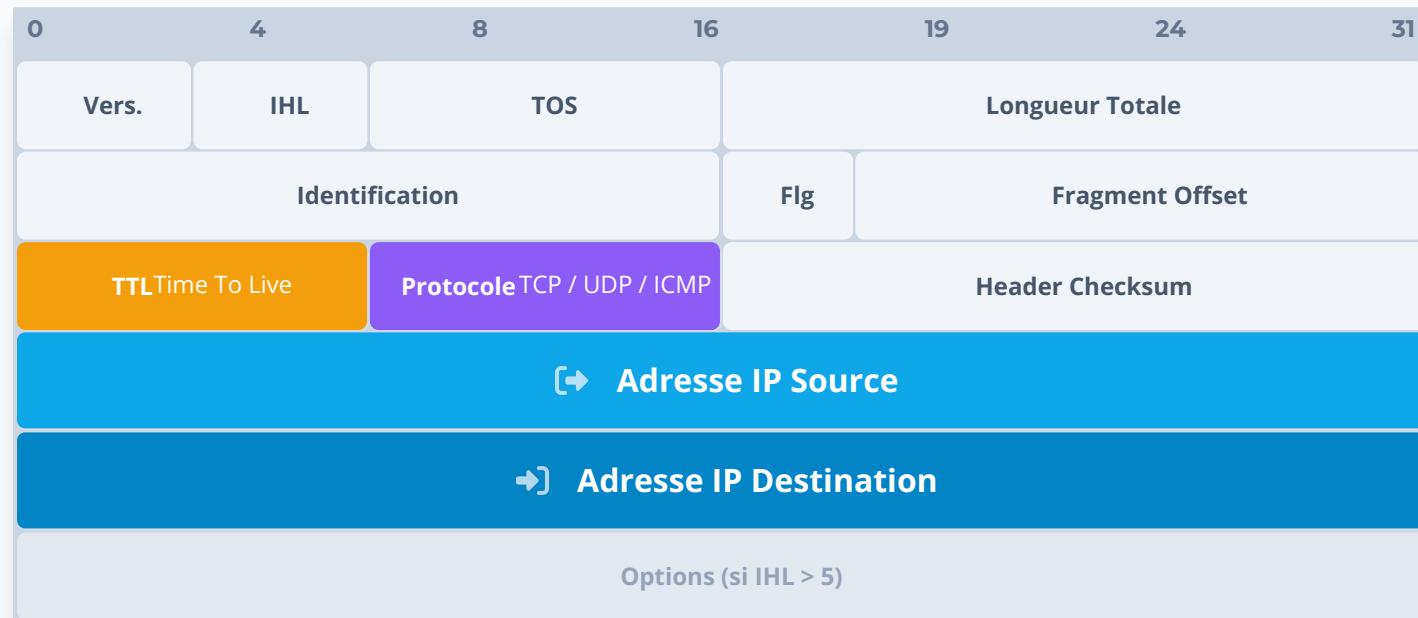
Contenu de la lettre

Structure technique d'un paquet

LE HEADER IP (COUCHE RÉSEAU)

Le protocole qui sait où aller

Structure du paquet IPv4 (32 bits)



RÔLE PRINCIPAL

Acheminer les paquets d'un réseau à un autre (Routage).

IP Source & Destination



L'adresse de l'expéditeur et celle du destinataire final. Indispensable pour la réponse.

TTL (Time To Live)

Durée de vie du paquet. Évite qu'un paquet ne tourne en boucle indéfiniment sur le réseau.

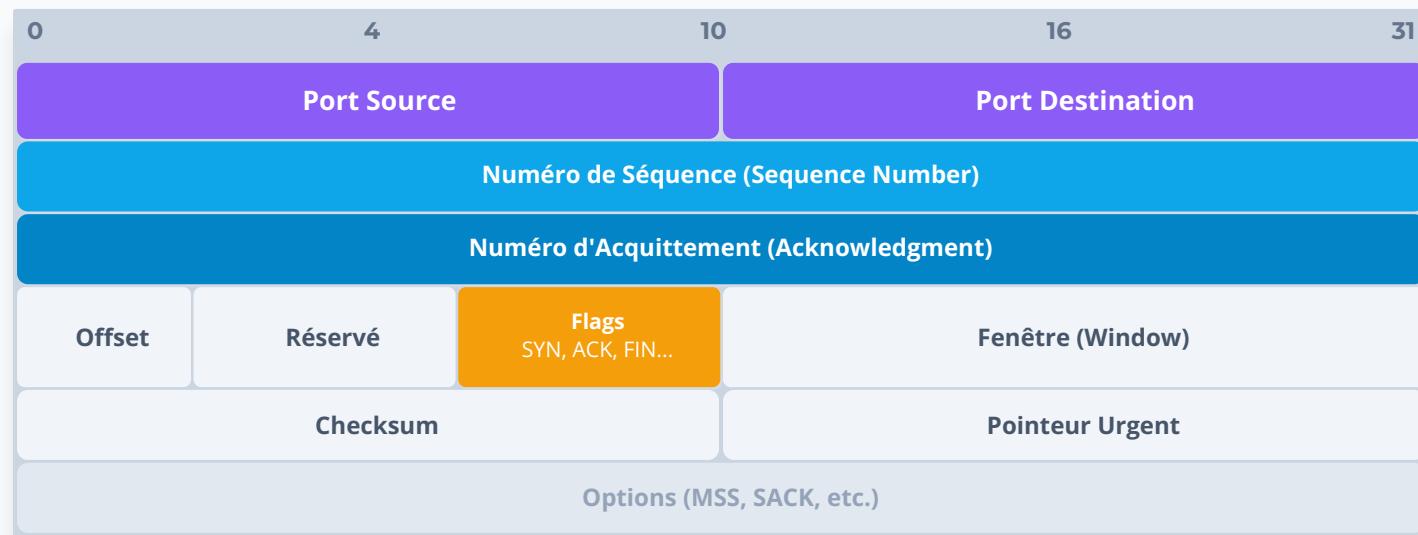
Protocole

Indique à qui remettre le contenu à l'arrivée (TCP = 6, UDP = 17).

LE HEADER TCP (COUCHE TRANSPORT)

Fiabilité et connexion orientée

Structure du Segment TCP (32 bits)



RÔLE PRINCIPAL

Garantir que les données arrivent complètes, sans erreurs et dans le bon ordre.

Ports Source / Destination



Identifie l'application précise (ex: Web=80, Mail=25) sur les machines.

Sequence & ACK



Permettent de remettre les paquets dans l'ordre et de confirmer la bonne réception.

Flags (Drapeaux)



Contrôlent l'état de la connexion : SYN (Ouvrir), FIN (Fermer), ACK (Valider).

LE HEADER HTTP (COUCHE APPLICATION)

Le protocole du Web (HyperText Transfer Protocol)

Raw Request

http_request.txt

```
GET/index.html HTTP/1.1
Host: www.example.com
User-Agent: Mozilla/5.0 (Windows NT 10) ...
Accept: text/html,application/xml
Accept-Language: fr-FR,en;q=0.5
Connection: keep-alive
(ligne vide)
```

RÔLE PRINCIPAL

Permettre la communication application ↔ application
(Client Web vers Serveur Web).

Méthode & URL



L'action demandée (GET = lire, POST = envoyer) et la ressource ciblée.

Host



Le nom de domaine du serveur destinataire. Obligatoire en HTTP/1.1.

User-Agent



La "carte d'identité" du client (navigateur, version, OS) qui effectue la requête.

RÉCAPITULATIF : QUI FAIT QUOI ?

Synthèse des rôles par protocole

Chaque en-tête (header) apporte une réponse précise à une question nécessaire à l'acheminement.

PROTOCOLE / HEADER	COUCHE OSI	QUESTION À LAQUELLE IL RÉPOND
 HTTP	APPLICATION (7)	? "Que veux-tu faire ?" Ex: Je veux voir la page index.html
 TCP	TRANSPORT (4)	? "À quel programme je parle ?" Ex: Au serveur Web (Port 80) ou Mail (25)
 IP	RÉSEAU (3)	? "Vers quelle machine je vais ?" Ex: 192.168.1.10 (Adresse logique)
 Ethernet	LIAISON (2)	? "Vers quel matériel voisin ?" Ex: Le routeur ou le PC d'à côté (Adresse MAC)

MESSAGE FINAL À RETENIR

Synthèse du cours

LES 3 IDÉES CLÉS

1

Le modèle OSI n'est pas qu'une théorie, c'est une structure essentielle pour comprendre le réseau.

2

L'encapsulation est un empilement progressif : on ajoute une enveloppe à chaque étape.

3

Chaque header a un rôle précis : IP pour la route, TCP pour la fiabilité, HTTP pour le contenu.