Réseaux TCP/IP

Pr. Mahrach Safaa

mahrachsafaa@gmail.com

Plan

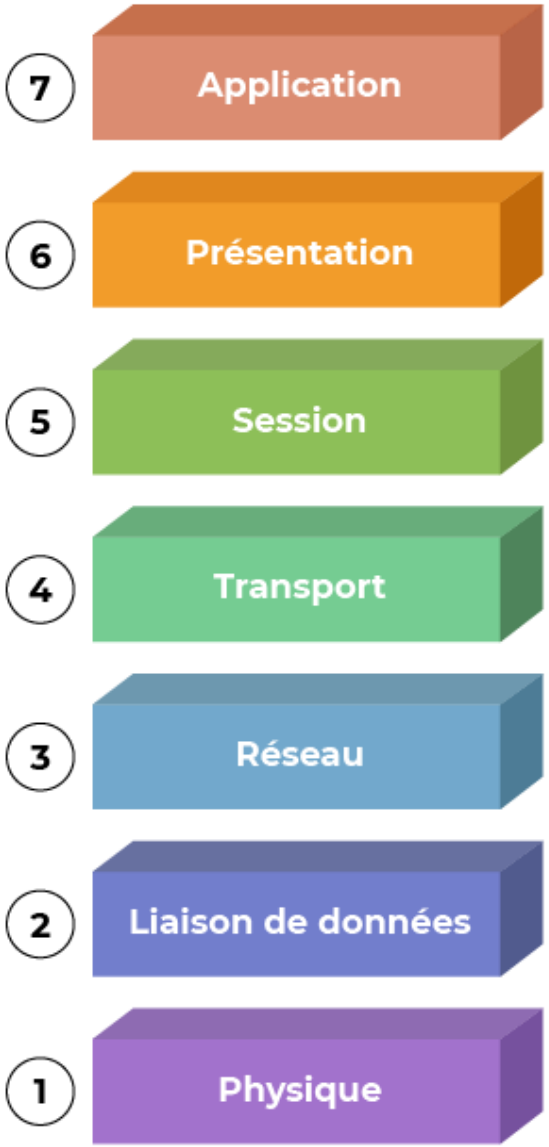
- Présentation du Modèle TCP/IP
  - Modèle TCP/IP vs OSI
  - Les couches TCP/IP
  - La couche Transport
    - Les Fonctionnalités
    - TCP & UDP (TP)
  - la couche Internet
    - Les protocoles Internet
    - IP, ICMP, ...(TP)
  - La couche Application
    - Les protocoles d'Applications
    - DNS, HTTP, ... (TP)

Plan

- Présentation du Modèle TCP/IP
  - Modèle TCP/IP vs OSI
  - Les couches TCP/IP

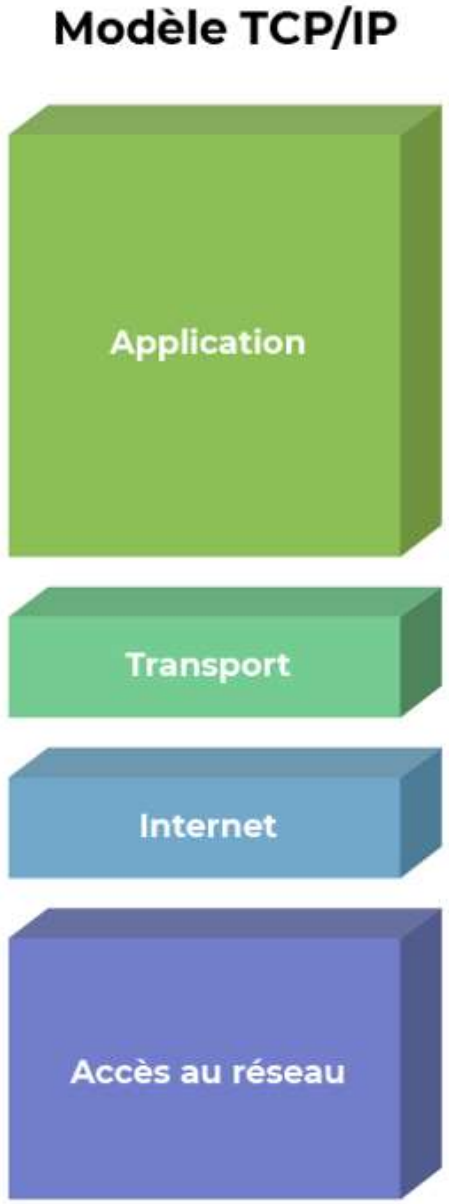
Modèle OSI

Modèle théorique utilisé principalement pour la compréhension des réseaux informatiques.



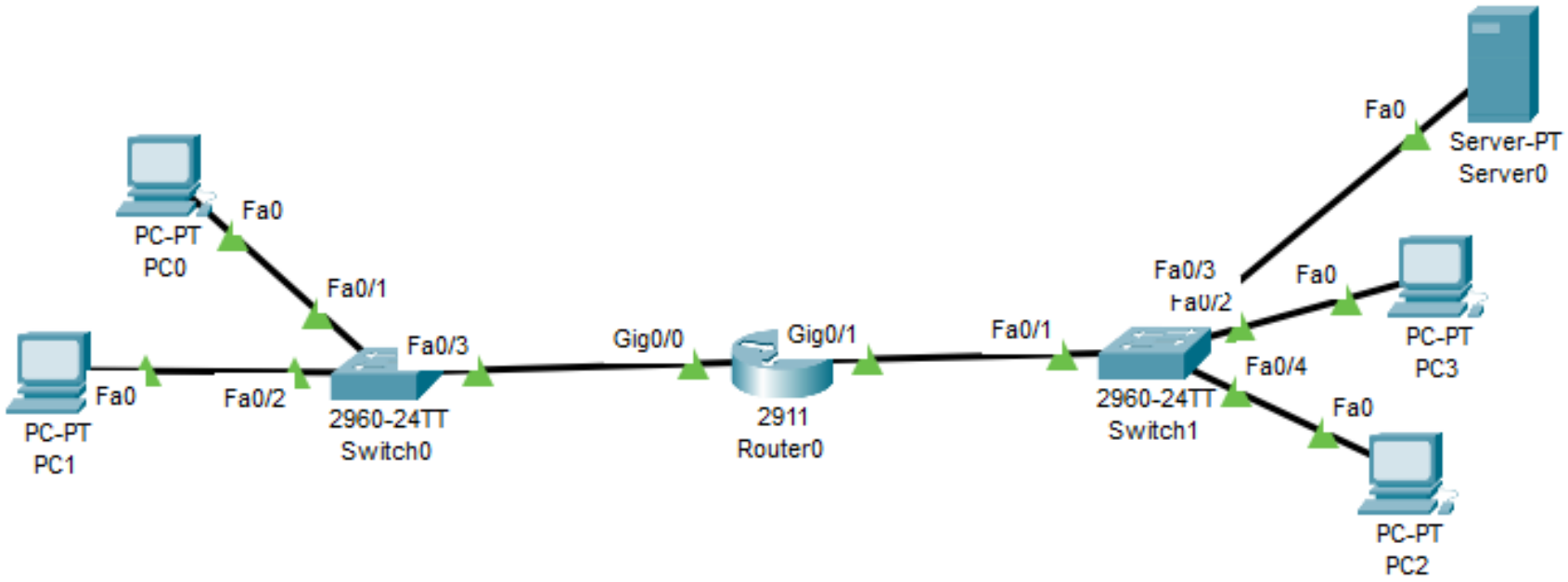
Modèle TCP/IP

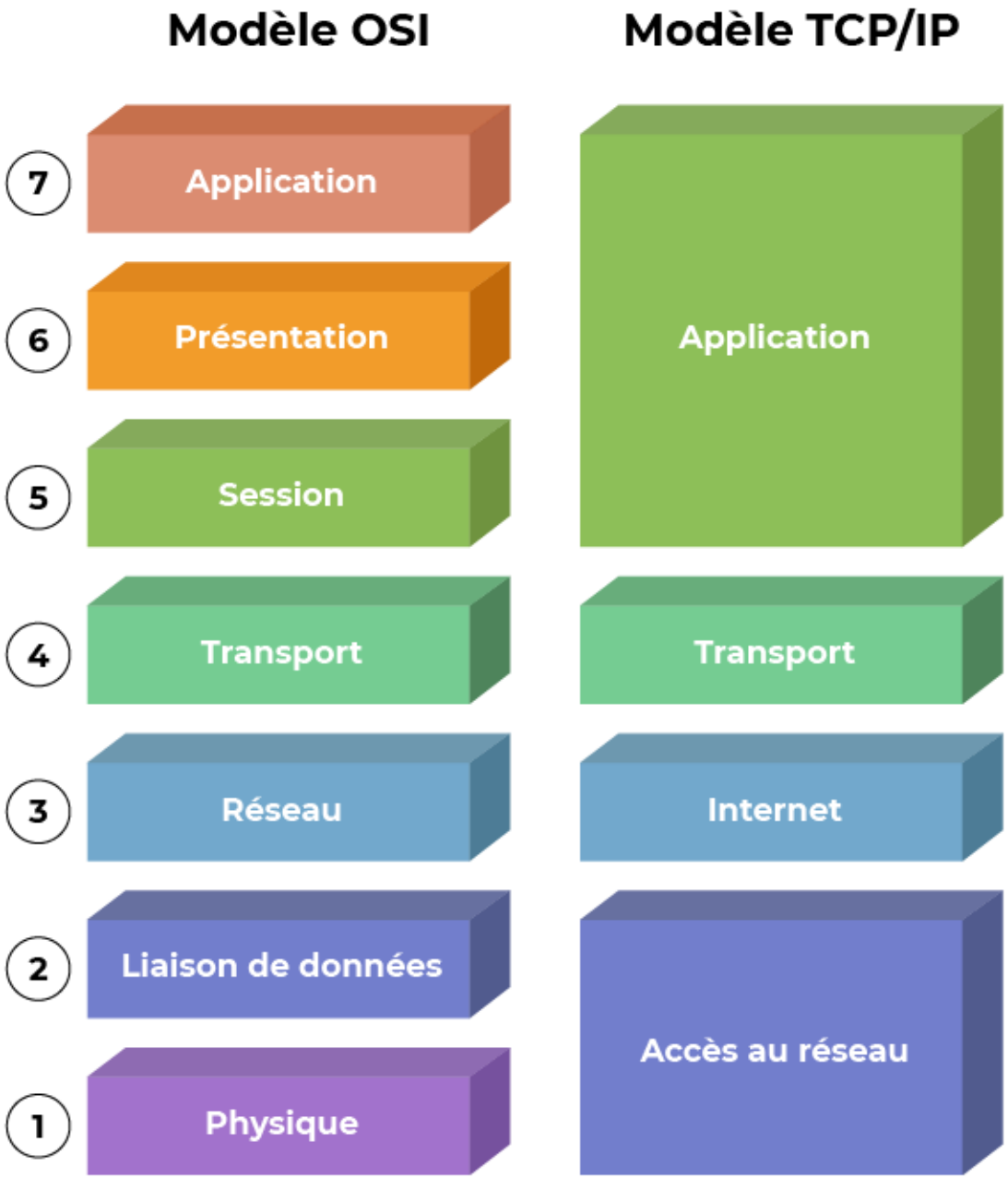
Pratiquement c’est le M**odèle TCP/IP** qui est utilisé pour assurer la communication entre les machines.

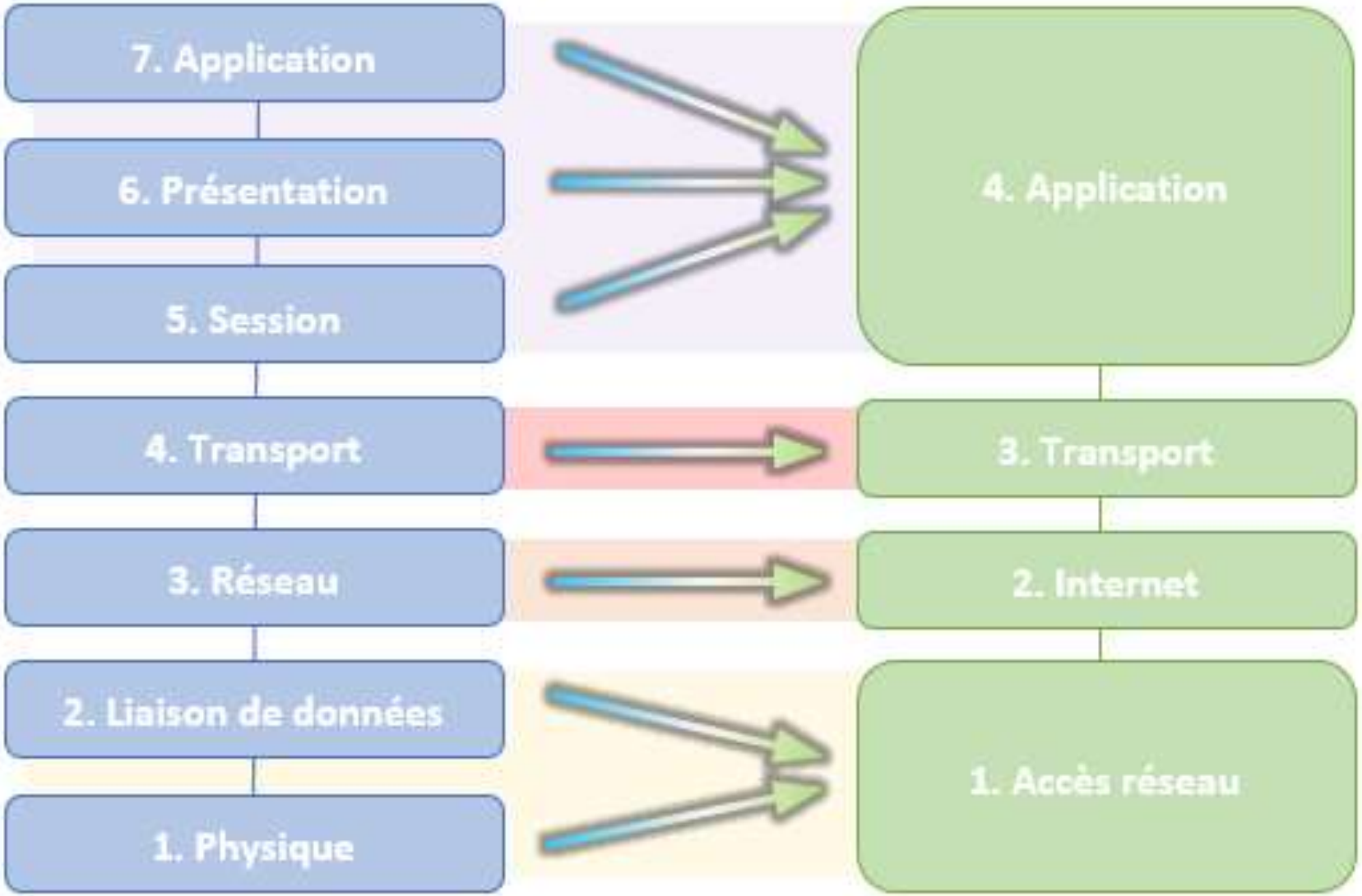


Modèle TCP/IP

TCP/IP est le modèle qui est implémenté au sein des machines



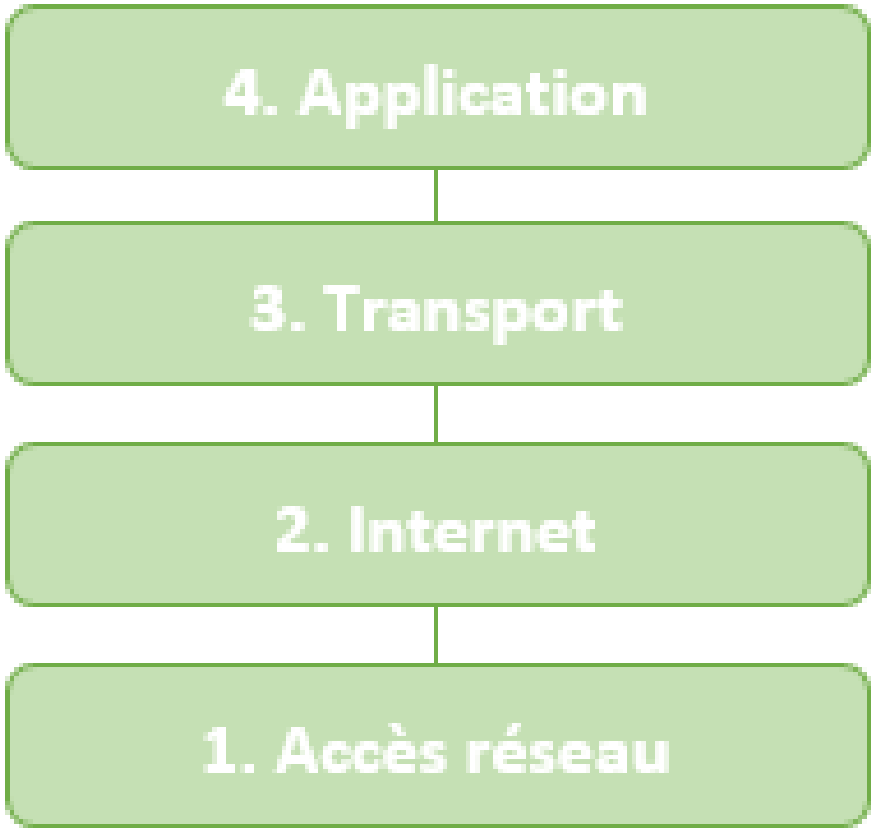

Modèle OSI
Modèle TCP/IP
7
Application
6
Présentation
5
Session
4
Transport
3
Réseau
2
Liaison de données
1
Physique
Application
Transport
Internet
Accès au réseau


7. Application
6. Présentation
5. Session
4. Transport
3. Réseau
2. Liaison de données
1. Physique
4. Application
3. Transport
2. Internet
1. Accès réseau

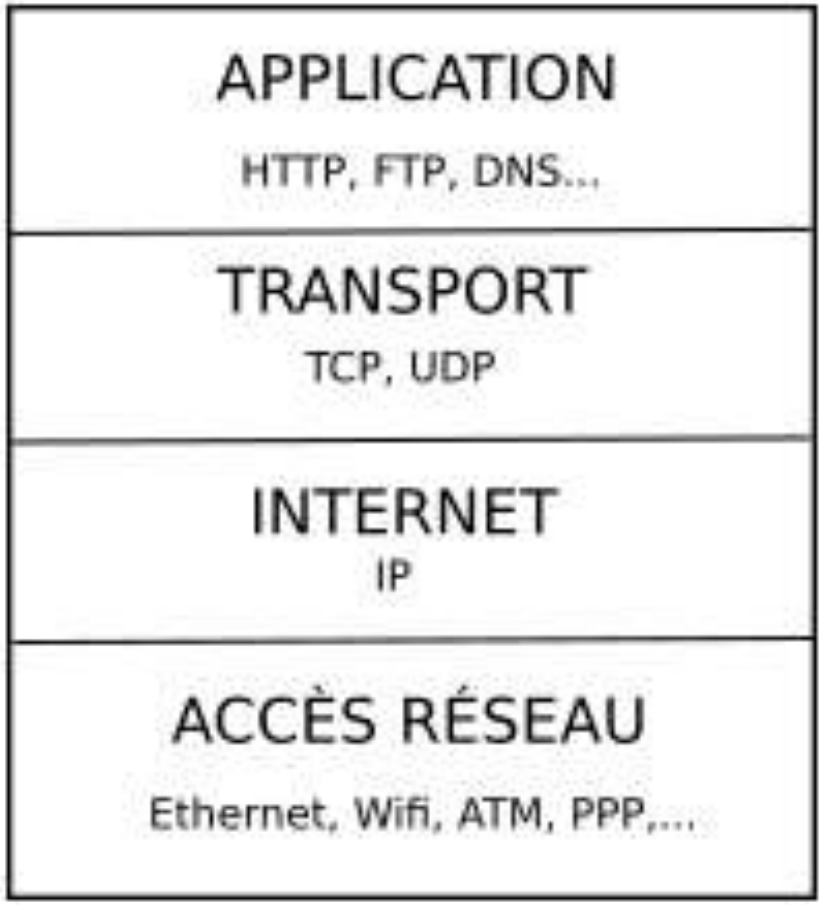
Modèle TCP/IP

- **Le modèle TCP/IP simplifie le modèle OSI en 4 couches:**
  - les 3 premières couches du modèle OSI ; application , présentation et session sont regroupées en une seule couche qui est **la couche Application,**
    - Cette dernière fournie des Applications a l'utilisateur final.
  - les 2 dernières couches du modèle OSI ;physique et liaison de données sont regroupées dans une couche unique sur le modèle TCP/IP, cette couche s'appeler Accès au Réseau,
    - cette couche s'occupe tout simplement de la connexion entre deux machines.

Modèle TCP/IP



Modèle TCP/IP



La couche 1 Accès réseau

- Elle contrôle **les périphériques matériels** ainsi que **les supports** constituant le réseau.

Au sens du *modèle TCP/IP la couche Accès Réseau est vide* (c.a.d; pas de protocole spécifique), car la pile des protocoles Internet (TCP/IP) est censée “inter-opérer” avec les différentes technologies (Ethernet, Wifi, ADSL, 4G, ...).

La couche 2 Internet

- Elle détermine **le meilleur chemin** à prendre par les paquets pour arriver à destination dans et vers n'importe quel réseau.
- Cette couche inclut :
  - **le protocole IP (Internet Protocol),**
  - le protocole ARP (Address Resolution Protocol, protocole de résolution d'adresse) et
  - le protocole ICMP (Internet Control Message Protocol, protocole de message de contrôle Internet).

La couche 3 Transport

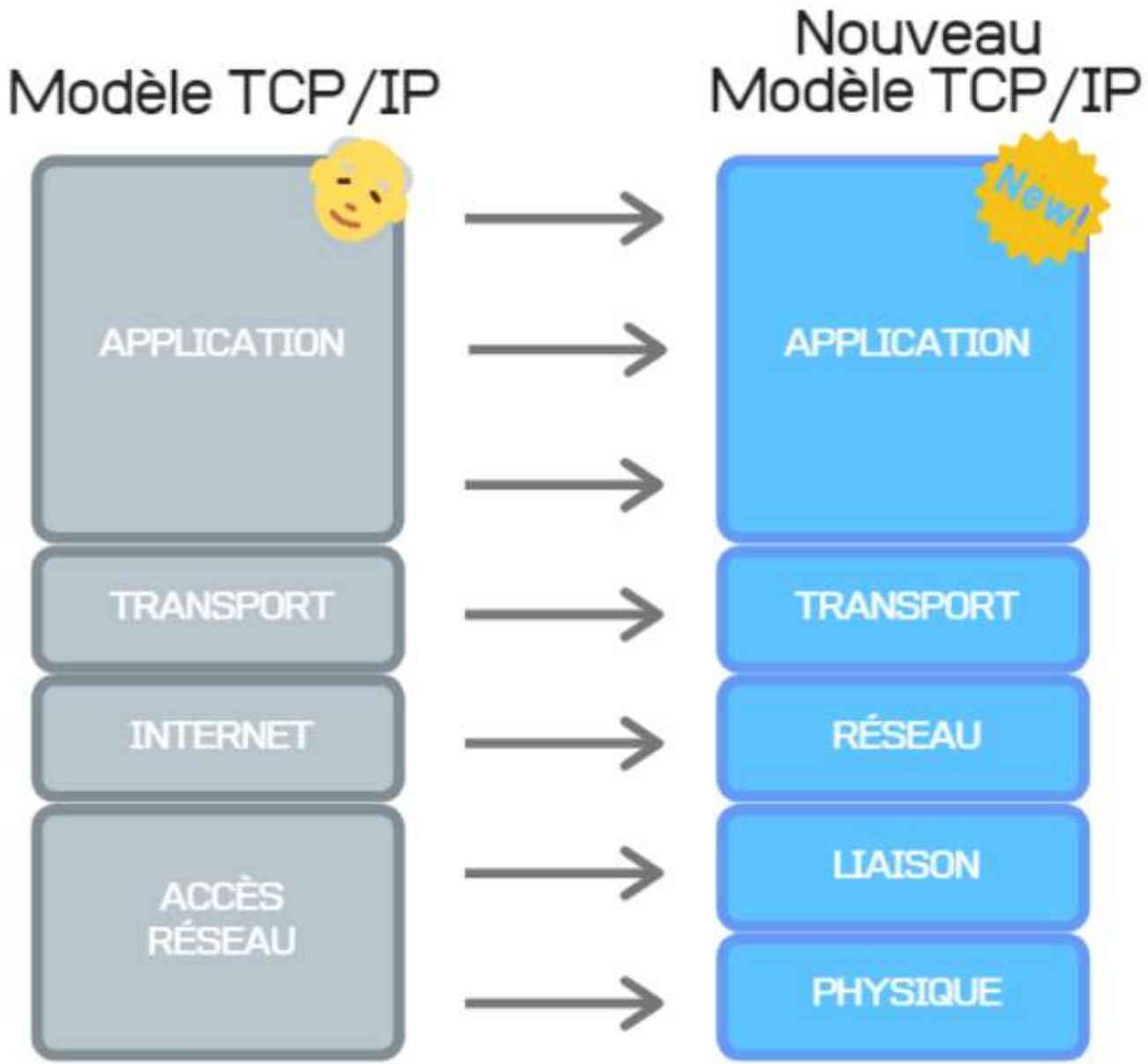
- Elle prend en charge **la communication entre les différents périphériques** à travers le réseau grâce aux protocoles de transport **TCP et UDP** notamment.

  - *Exemple d'usage de TCP*: Transferts de fichiers, Pages Web, mails, ...
  - *Exemple d'usage d'UDP:* Visioconférence, streaming en live, appel Skype, ...

La couche 4 Application

- Elle **représente les données** pour l'utilisateur (interface Homme/machine), gère le codage et le contrôle du dialogue

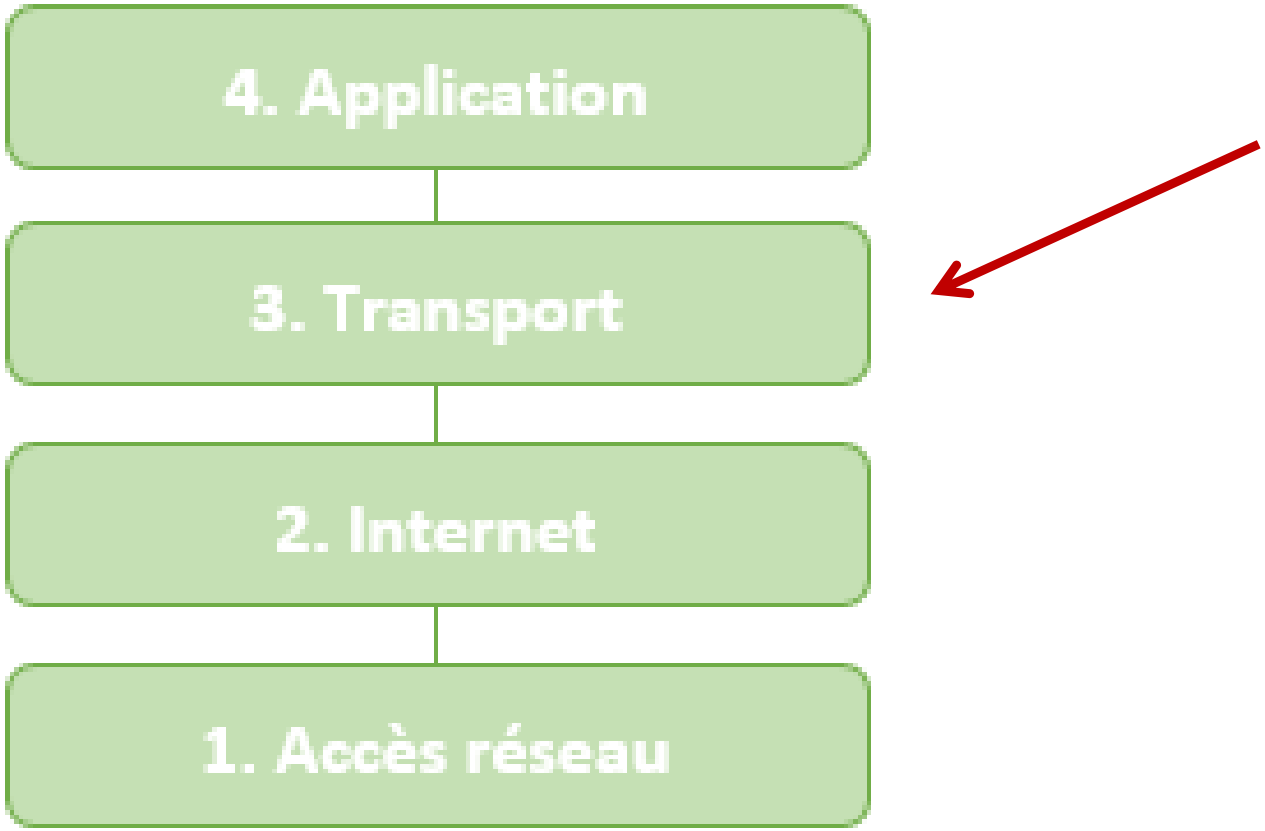
- *Des exemples de protocoles utilisés:*
  - HTTP,
  - FTP,
  - DNS,
  - DHCP, ...


Modèle TCP/IP
APPLICATION
TRANSPORT
INTERNET
ACCÈS RÉSEAU
Nouveau Modèle TCP/IP
New!
APPLICATION
TRANSPORT
RÉSEAU
LIAISON
PHYSIQUE

Plan

- Présentation du Modèle TCP/IP
  - Modèle TCP/IP vs OSI
  - Les couches TCP/IP
  - La couche Transport
    - Les Fonctionnalités
    - TCP & UDP (TP)

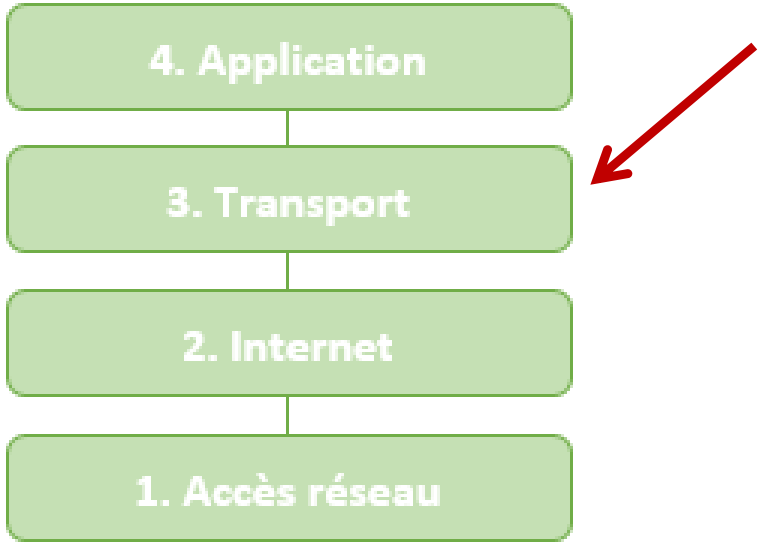
Modèle TCP/IP



La couche Transport

- **La couche transport** est fondamentale pour le bon fonctionnement de l'architecture réseau TCP/IP.
- *La couche Internet,* ne peut pas garantir, la livraison des informations vers sa destination.

➢C'est le rôle de la couche transport



La couche Transport

- Les deux protocoles de cette couche, les plus courants sont **TCP et UDP.**
- Ils gèrent la communication de plusieurs applications.
- Le service de base offert par **la couche de transport**
  - c'est le suivi de la communication individuelle entre les applications sur les hôtes sources et destination. Ce service s'appelle le multiplexage de session, et il est exécuté par ***UDP et TCP.***
  - Une différence majeure entre ***TCP et UDP*** est que TCP garantit la bonne livraison des données, chose que UDP ne fait pas.

La couche Transport

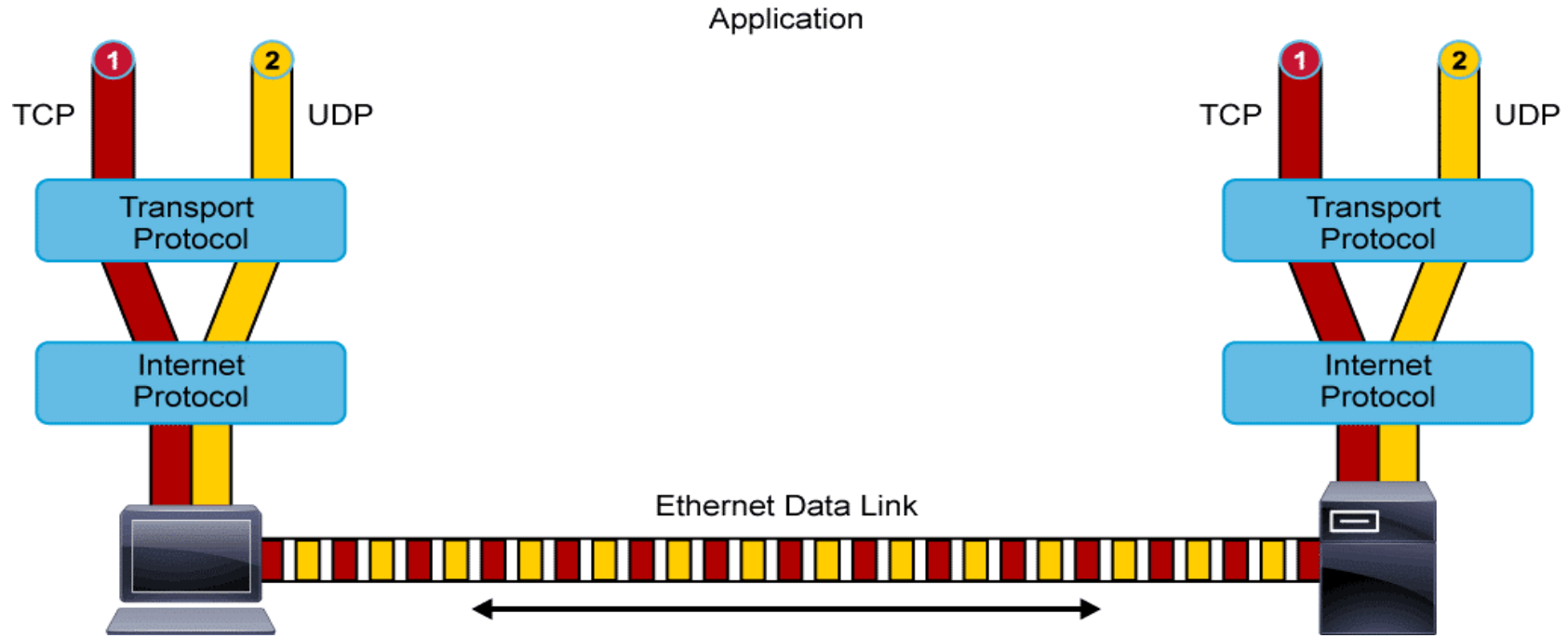
Multiplexage de session

- ***Des communications multiples*** arrivent très couramment, par exemple,
  - *faire des recherches sur le Web*
  - tout en utilisant *FTP pour transférer un fichier.*
- la communication entre deux hôtes, nécessite l'établissement et gestion d'une session, on peut dire qu'une communication est représentée par une session

La couche Transport

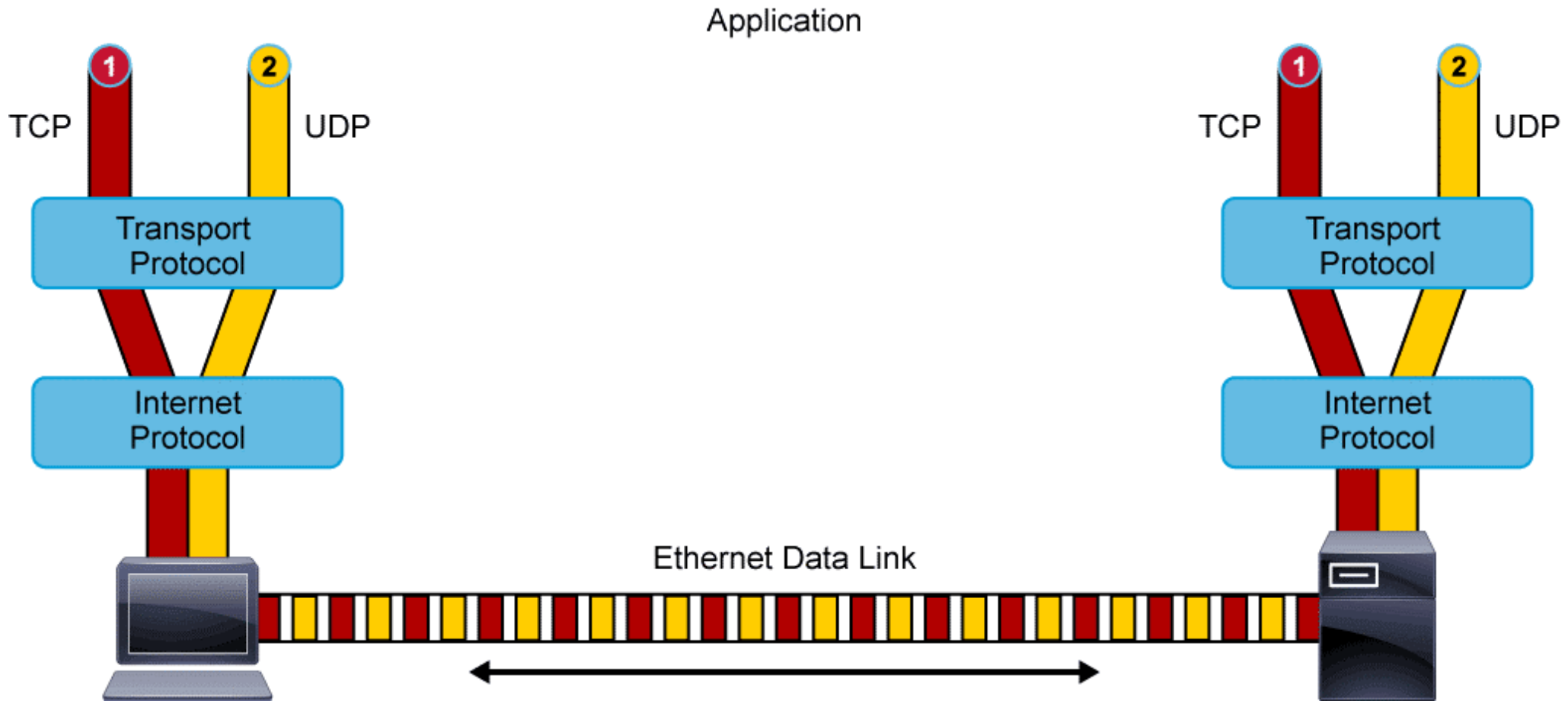
Multiplexage de session

- **La fonction de multiplexage** permet a un hôte IP de supporter simultanément plusieurs sessions et gérer les flux sur un seul lien de communication.



La couche Transport

Multiplexage de session



La couche Transport

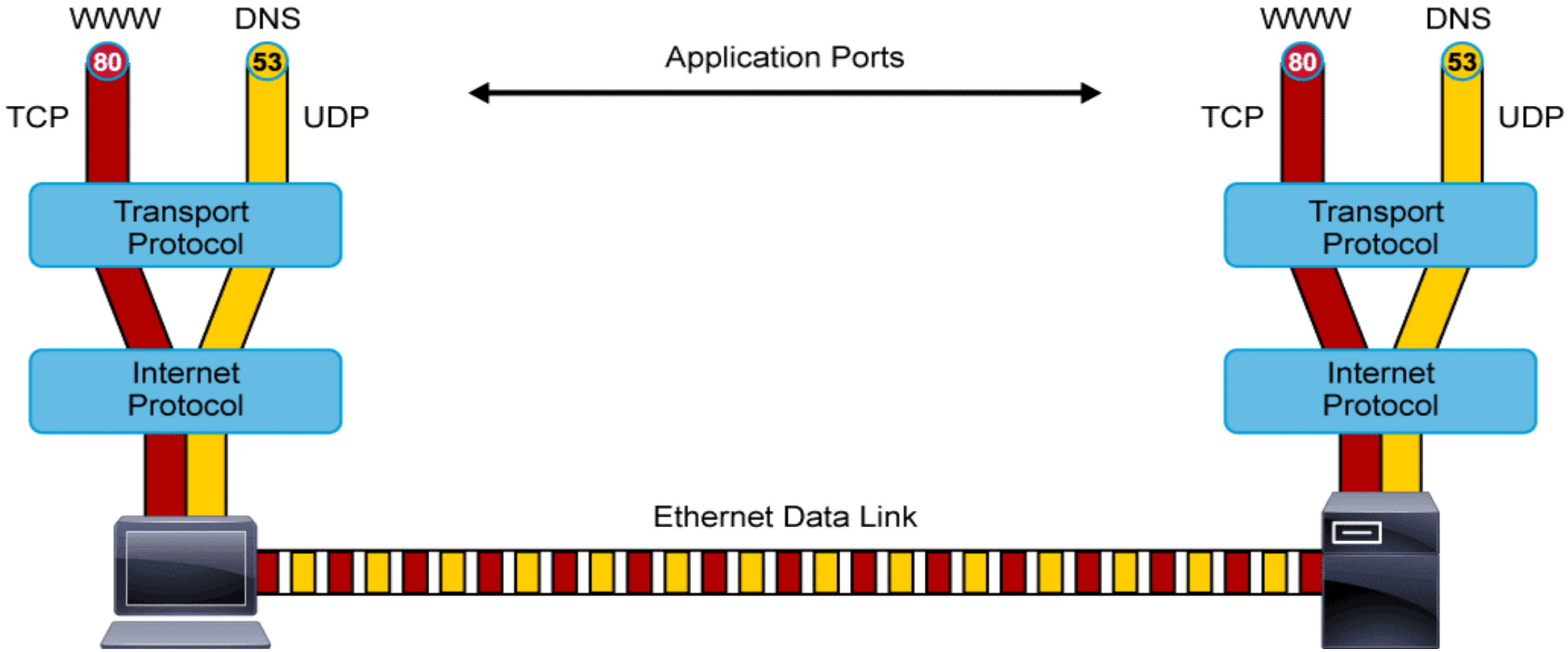
Identification des applications

- Pour transmettre les données aux applications, ***la couche de transport doit identifier l'application de destination.***

- Les protocoles de transport **TCP** et **UDP** utilisent des numéros de port pour accomplir cette tâche.
  - Chaque processus, qui doit accéder au réseau, reçoit un numéro de port unique pour cet hôte.
  - Le numéro de port est utilisé dans l'entête de la couche de transport pour indiquer de quelle application est associée cette donnée.

La couche Transport

Identification des applications



La couche Transport

Identification des applications

- L'organisme IANA (Internet Assigned Numbers Authority) classe les numéros de port en trois catégories principales, comme l'illustre le tableau ci-dessous.

Portée	Catégorie	Description
0 - 1023	Ports bien connus	Ports réservés pour des services bien connus (web, envoi de mail, etc.).
1024 – 49 151	Ports réservés	Ports réservés pour être utilisés par des applications propriétaires.
49 152 – 65 635	Ports dynamiques	Ports « libres » que vous pouvez utiliser pour vos applications. Ils ne sont ni pour des services bien connus, ni réservés par une entreprise quelconque.

La couche Transport

Segmentation

- **Le protocole TCP** prend des blocs de données de taille différente de la couche application et les prépare pour les transporter sur le réseau.

- L'application repose sur *TCP* pour s'assurer que chaque fragment est divisé **en segments plus petits correspondants au *MTU*** *(Maximum Transmission unit)*
  - La *MTU* permet de définir la taille maximale en octet, du paquet pouvant être transmis en une seule fois.

La couche Transport

Segmentation

- **Le protocole UDP** ne fournit pas de services de segmentation.
- Il s'attend à ce que le processus d'application effectue lui-même la segmentation et lui fournisse des blocs de données ne dépassant pas la MTU des couches inférieures.

- La MTU du protocole IP est de **1500 octets**. Des MTU plus grandes sont possibles, mais 1500 octets est la taille normale.

La couche Transport

Orienté connexion (Fiable)

- Dans la couche transport, **un protocole orienté connexion** établit une connexion de session entre deux hôtes IP puis maintient la connexion pendant toute la transmission.
- Une fois la transmission terminée, la session est terminée.

La couche Transport

Orienté connexion (Fiable)

- **TCP** est le protocole qui fournit un transport fiable orienté connexion pour les données d'application.

La couche Transport

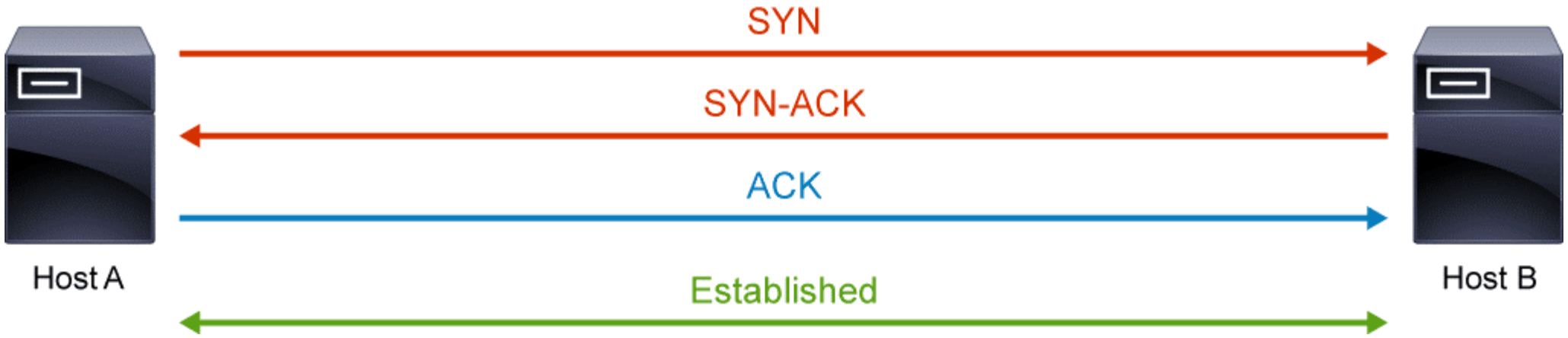
Orienté connexion (Fiable)

- *TCP utilise 3 étapes pour établir une connexion:*
  - En premier, la source de la connexion envoie un paquet SYN qui signifie : Synchronisation, à la destination demandant l'établissement d'une session. Le numéro de séquence commence par un 0.
  - En second, la destination répond au SYN avec un SYN-ACK: Synchronisation-ACKnowledgment, et augmente le numéro de séquence par 1.
  - Et pour finir, si la source accepte le SYN-ACK, elle envoie un paquet ACK:ACKnowledgment, pour établir la connexion.

La couche Transport

Orienté connexion (Fiable)

Connexion TCP (« 3-Way Handshake »)



La couche Transport

Contrôle de flux (Fiable)

- Le protocole TCP inclut également des mécanismes de contrôle de flux.
- Le contrôle de flux aide à maintenir la fiabilité des transmissions TCP en gérant le flux de données entre la source et la destination.

La couche Transport

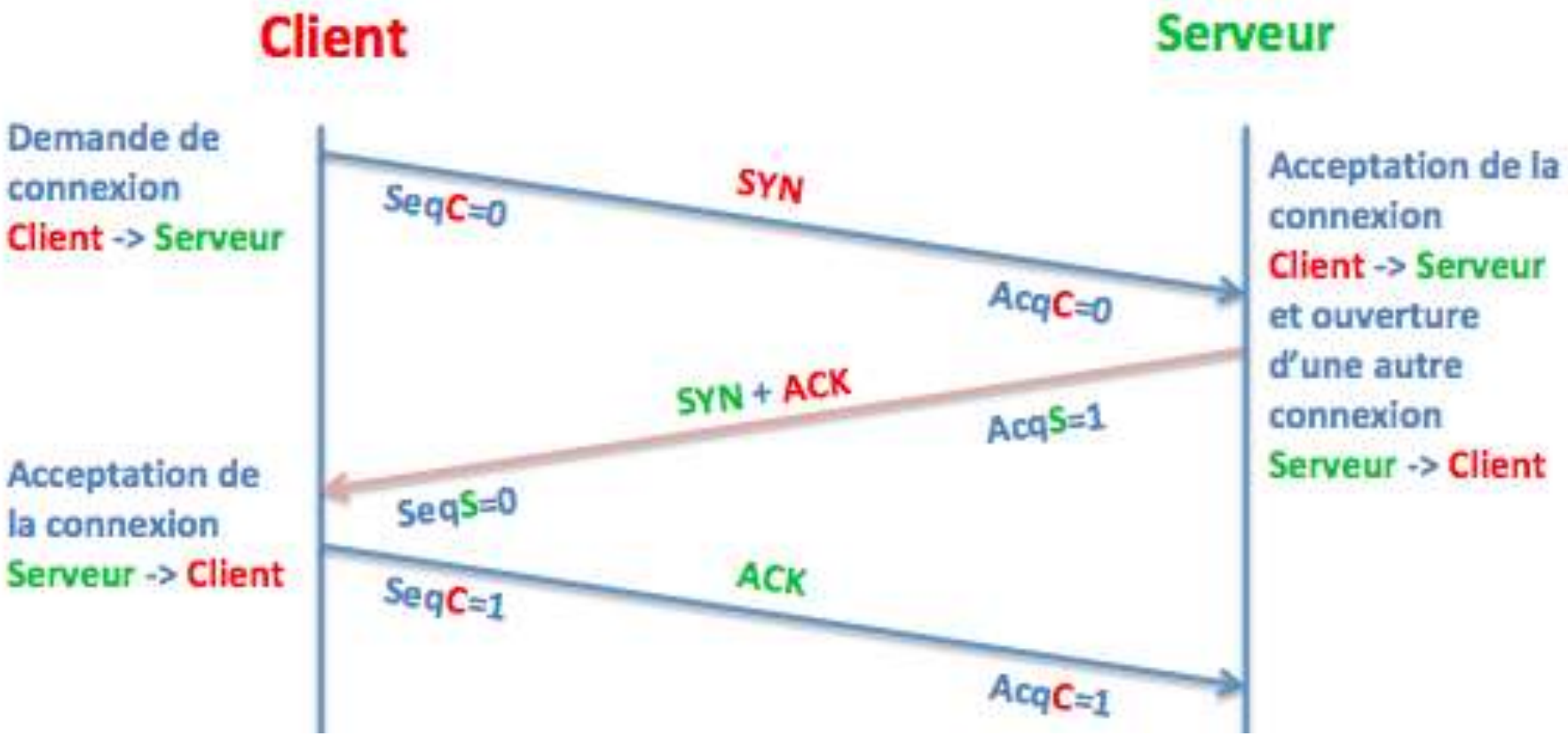
Contrôle de flux (Fiable)

Durant la transmission des données;

Lorsque l'ordinateur émetteur transmet des données, il attribue un **numéro de séquence** à chaque paquet. Le récepteur répond ensuite avec **un numéro d'accusé de réception** qui est égal au prochain numéro de séquence attendu.

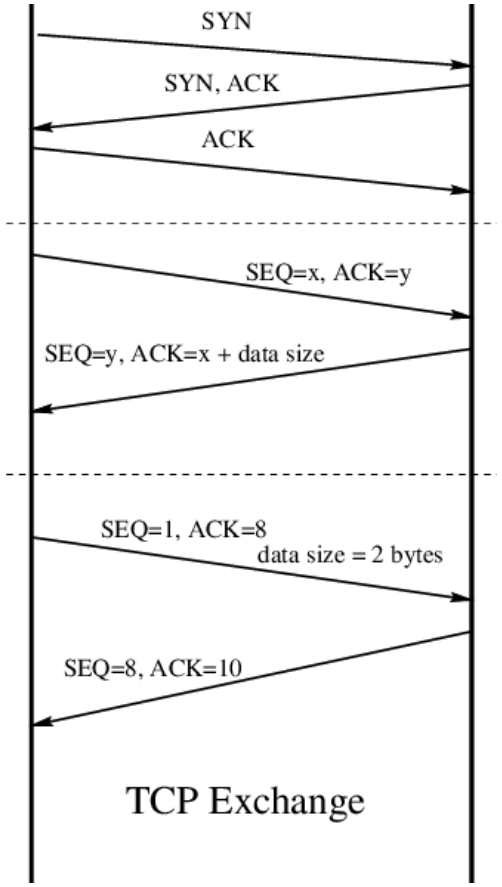
- Cet échange de séquence et les numéros d'accusé de réception permettent au protocole ***de reconnaître lorsque des données ont été perdues, ou en double.***

les numéros de séquence et d'accusé de réception



La couche Transport

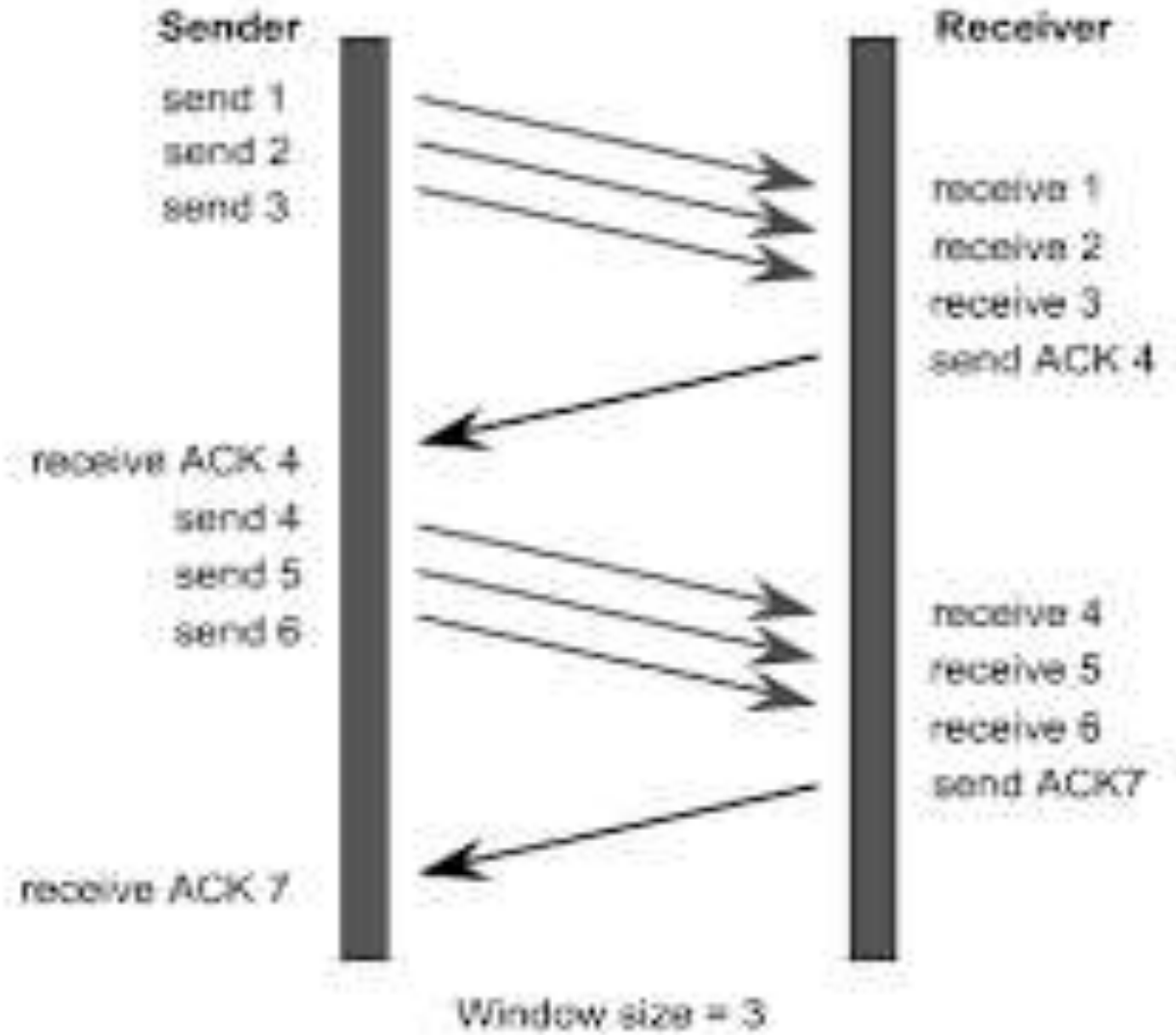
Contrôle de flux (Fiable)



La couche Transport

Contrôle de flux et fenêtrage

- Contrôle de flux avec fenêtrage.
- La combinaison de l'utilisation **des numéros de séquence et d'acquittement** avec **la notion de fenêtrage** permet de contrôler la quantité de données à transmettre avant de procéder à un acquittement.


Sender
Receiver
send 1
send 2
send 3
receive 1
receive 2
receive 3
send ACK 4
receive ACK 4
send 4
send 5
send 6
receive 4
receive 5
receive 6
send ACK7
receive ACK 7
Window size = 3

La couche Transport

Contrôle de flux (Fiable)

➢Fiabilité et Contrôle de flux

La fiabilité du TCP a trois objectifs principaux :

- Détecter et retransmettre les paquets abandonnés
- Détecter et réhabiliter les données en double ou hors service
- Et éviter la congestion dans le réseau

La couche Transport

Non Orienté Connexion

- La fiabilité n’est pas toujours nécessaire. Par exemple, si un ou deux segments d’un flux vidéo en streaming ne parviennent pas, cela créerait juste une perturbation momentanée dans le flux.
- Dans les applications en temps réel, telles que la diffusion audio et vidéo, les paquets abandonnés peuvent être tolérés tant que le pourcentage global de paquets n’est pas dépassé.

La couche Transport

Non Orienté Connexion

- **Le protocole UDP** fournit aux applications une livraison optimale et n'a pas besoin de conserver des informations d'état sur les données qu'elle à envoyer précédemment.
- Contrairement à TCP, UDP n'a pas besoin d'établir de connexion avec le récepteur, c'est un protocole sans connexion.
  - Cela est souhaitable pour les applications qui nécessitent une communication plus rapide, sans contrôle ni vérification.

La couche Transport

- Les applications les plus courantes qui utilisent TCP, on retrouve:
    - HTTP/HTTPS (World Wide Web),
    - SMTP/POP3/IMAP (messagerie)
    - et FTP (transfert de fichiers).

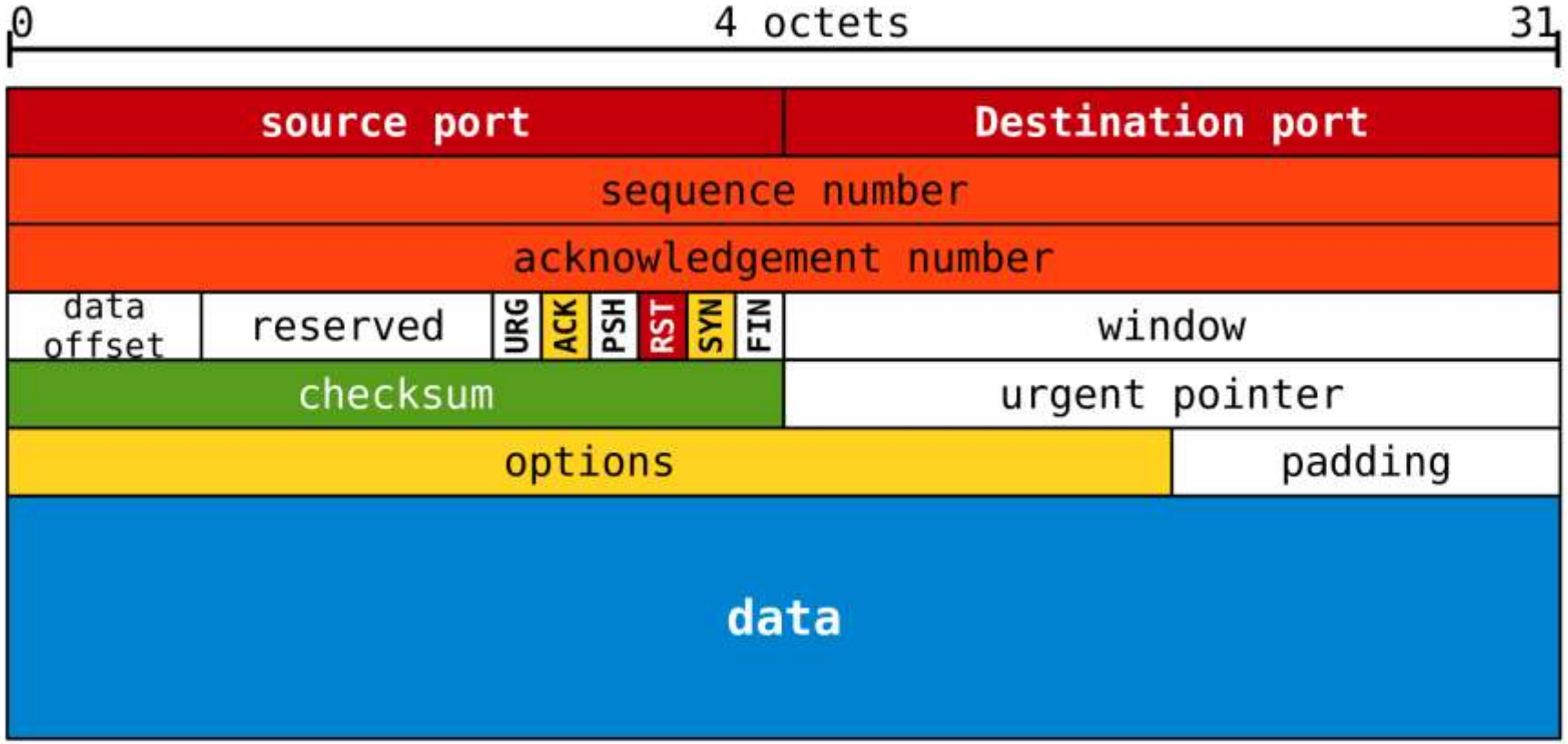
La couche Transport

- Parmi les applications les plus courantes qui utilisent UDP, on retrouve:
  - le DNS (Domain Name System), la vidéo en streaming,
  - la Voip (Voice Over IP),
  - ou bien le TFTP (Trivial File Transfert Protocol)

En-tête TCP

Les fonctions d'établissement, de maintien, de libération et de contrôle des échanges ont conduit au développement d'un en-tête comprenant un grand nombre de champs.

En-tête TCP


0
4 octets
31
source port
Destination port
sequence number
acknowledgement number
data offset
reserved
URG
ACK
PSH
RST
SYN
FIN
window
checksum
urgent pointer
options
padding
data

En-tête TCP

Source Port : 16 bits Numéro du port source.

Destination Port : 16 bits Numéro du port destination.

Sequence Number : 32 bits Numéro de séquence.

Acknowledgment Number : 32 bits Numéro d'acquittement.

Data Offset : 4 bits Indication du début des données.

Reserved : 6 bits Champ réservé pour une utilisation ultérieure.

Control bits : 6 bits Ces bits sont les indicateurs d'état qui servent à l'établissement, au maintien et à la libération des connexions TCP.

Psh demande a l'emetteur d'envoyer

En-tête TCP

Window : 16 bits Nombre d'octets de données à transmettre.

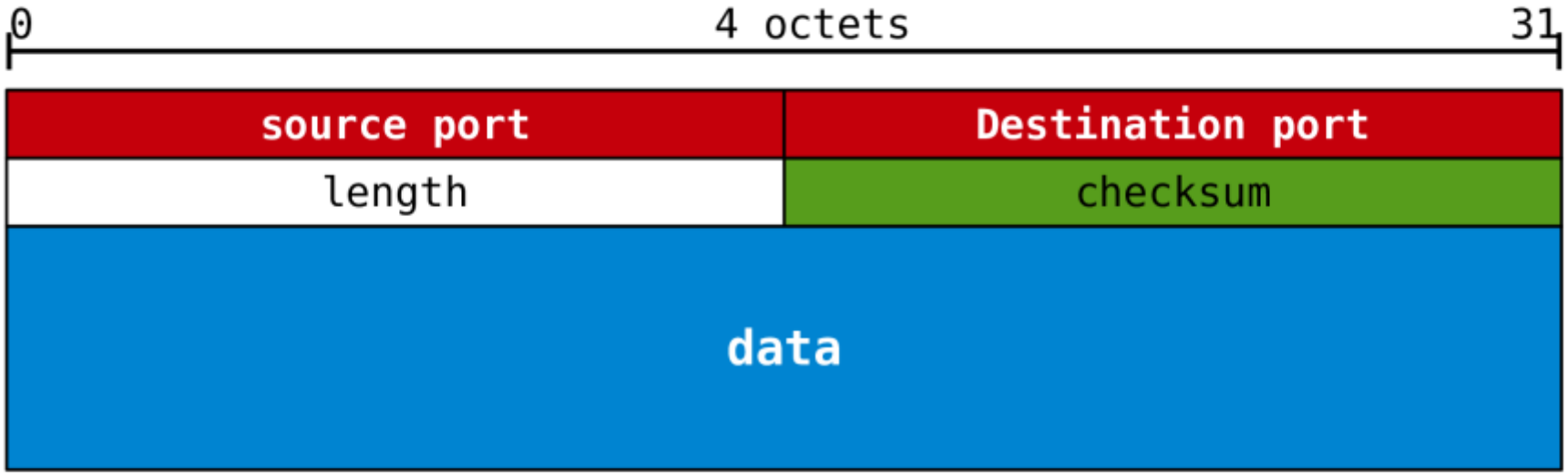
Checksum : 16 bits Somme de contrôle sur 16 bits de l'en-tête et des données.

Urgent Pointer : 16 bits Ce champ est interprété uniquement si le bit de contrôle URG est à 1.

Options : variable entre 0 et 44 octets

En-tête UDP

Les numéros de ports constituent le mécanisme d'adressage pour les communications de bout en bout comme dans le cas du protocole TCP.



Plan

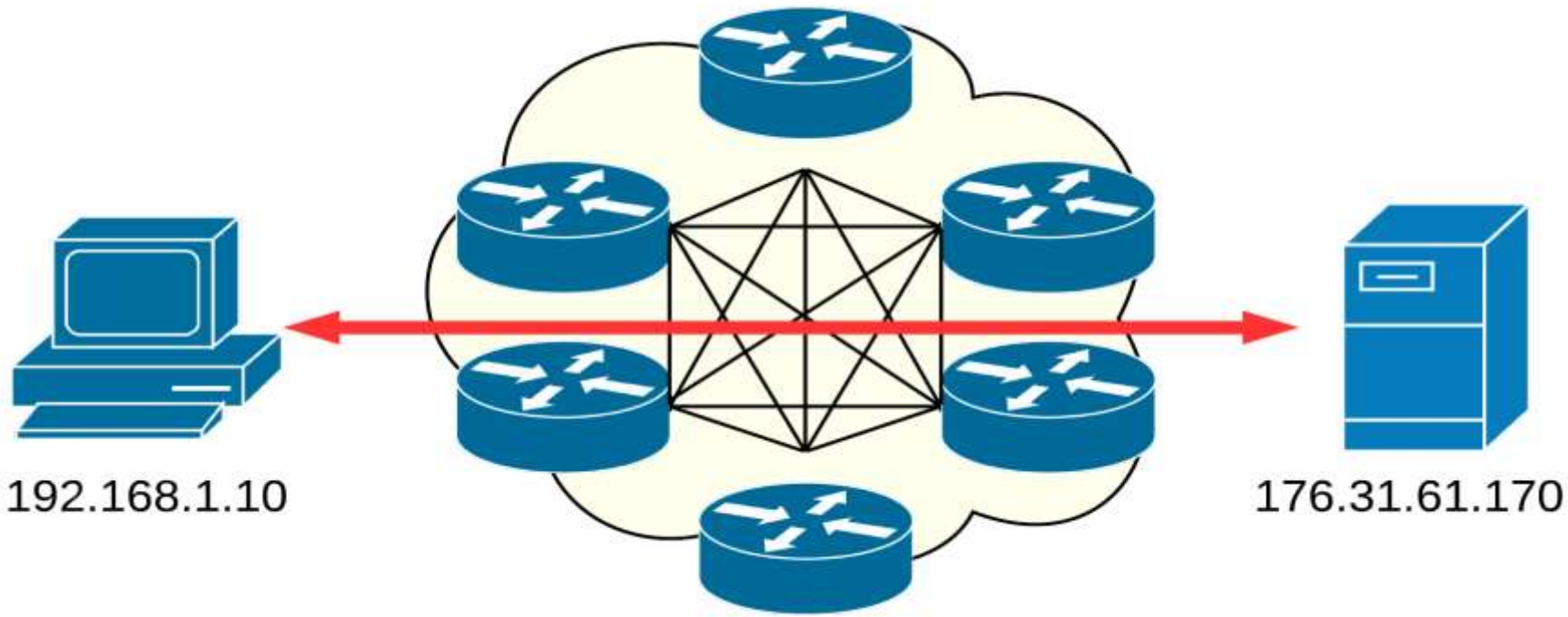
- Présentation du Modèle TCP/IP
  - Modèle TCP/IP vs OSI
  - Les couches TCP/IP
  - La couche Transport
    - TCP & UDP (TP)
  - la couche Internet
    - Les protocoles de la couche Internet
    - IP, ICMP, (TP)

Couche Internet

- **La couche Internet** est celle qui permet à deux ordinateurs situés à n'importe quel endroit du monde de communiquer directement entre eux.

✓Elle s'occupe d'adresser globalement les interfaces : elle remplit une fonction d'**adressage**.

✓Elle détermine les meilleurs chemins à travers les inter-réseaux : elle remplit une fonction de **routage**.


192.168.1.10
176.31.61.170

Couche Internet

Protocole IP

- Création des datagrammes (paquets) et routage
- Sans connexion, ni contrôle d'erreur, ni contrôle de flux, ni remise en ordre des datagrammes,
- IP fournit un service de qualité, mais ne garantit pas la livraison de paquets. Un paquet peut être mal orienté ou perdu sur le chemin de sa destination,
- Chaque datagramme est traité indépendamment des autres, ce qui signifie que chaque paquet peut parcourir un chemin différent pour se rendre vers la destination,
- Aucune sécurité
- Les services non assurés sont laissées à la charge de la couche supérieure.

Protocole IP

Aujourd'hui, il existe deux versions de protocole IP:

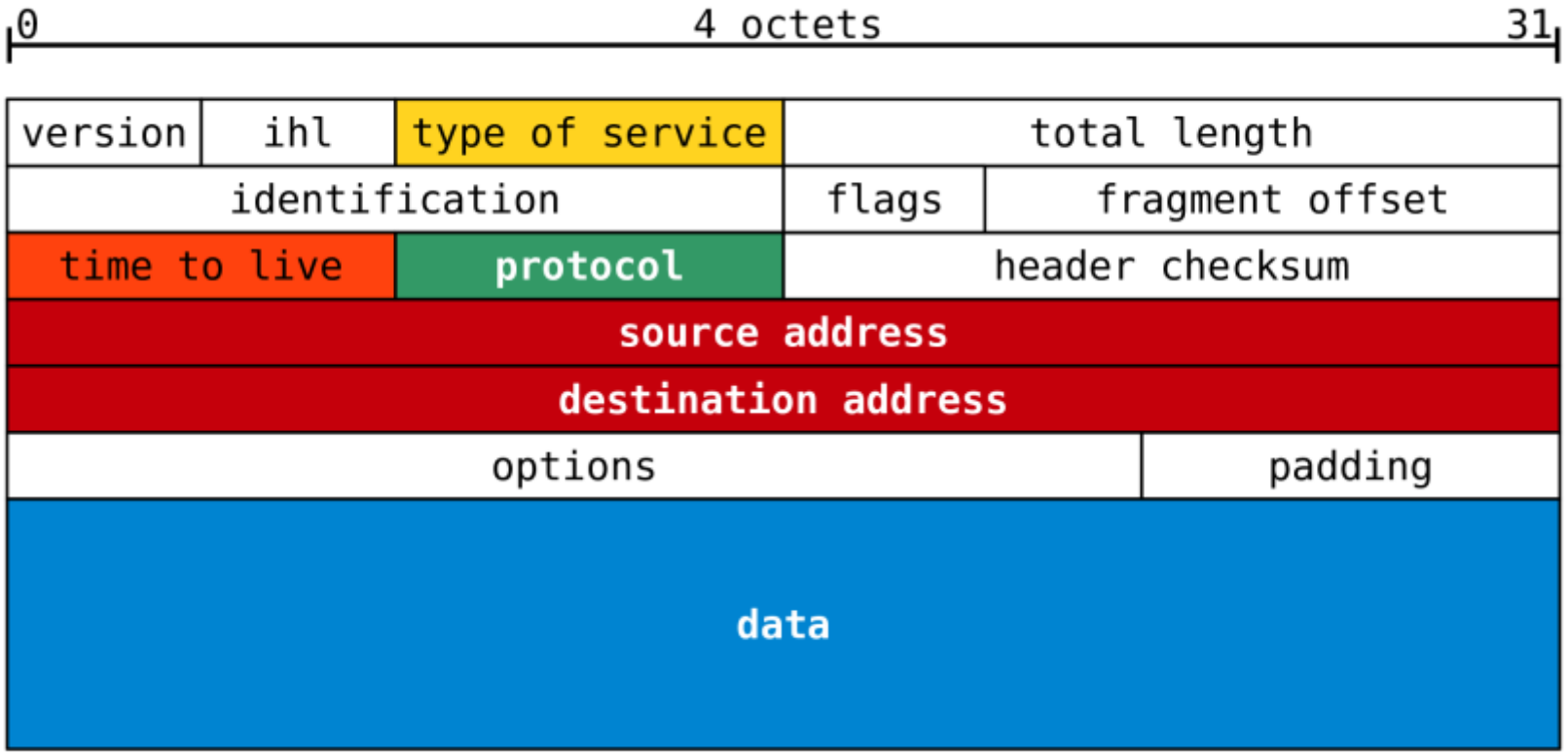
- Le protocole IPv4, dont les adresses sont représentées sur 32 bits, est le plus largement utilisé mais son espace d'adressage arrive à épuisement puisque toutes les adresses disponibles ont déjà été attribuées.

Protocole IP

- Le protocole IPv6, dont les adresses sont représentées sur 128 bits, est adopté progressivement mais à un rythme très lent.

Au delà du gain en espace d'adressage, l'en-tête IPv6 est plus simple que l'en-tête IPv4 et les traitements d'analyse doivent être plus légers et raccourcir les temps de commutation dans les équipements d'interconnexion.

En-tête paquet IPv4



En-tête paquet IPv4

Version: Version du protocole IP codée sur 4 bits

IHL: Longueur de l'en-tête codée 4 bits.

Type Of Service : 8 bits, TOS Type de Service, permet de donner une information sur la nature des données contenues dans le datagramme IP. Cette information peut être utilisée pour améliorer et diriger le routage.

Total Length : 16 bits Longueur du datagramme : en-tête & données.

Identification : 16 bits Chaque paquet IPv4 reçoit un numéro d'identification à sa création (fragments).Il est possible qu'un paquet soit découpé en fragments avant d'atteindre sa destination finale. Chaque fragment (appartenant au même paquet IPv4) possède le même numéro d'identification.

Flags : 3 bits Ce champ contient 3 indicateurs d'état (Reserved flag, Don't Fragment (DF), More Fragments (MF)) pour la gestion des fragments.

En-tête paquet IPv4

Fragment Offset : 13 bits Position du fragment dans le datagramme courant.

Time To Live : 8 bits, TTL Ce compteur est décrémenté à chaque traversée de routeur. Si la valeur 0 est atteinte, le paquet est rejeté.

Protocol : 8 bits Ce champ spécifie le protocole utilisé dans les données du paquet IP.

Header Checksum : 16 bits A chaque création ou modification d'un paquet, une somme de contrôle (cyclic redundancy check) est calculée sur son en-tête.

Source Address : 32 bits Adresse IPv4 de l'hôte qui a émis le paquet.

Destination Address : 32 bits Adresse IPv4 de l'hôte qui doit recevoir le paquet.

Options and Padding Cette partie de l'en-tête est optionnelle.

Protocoles de la couche Internet

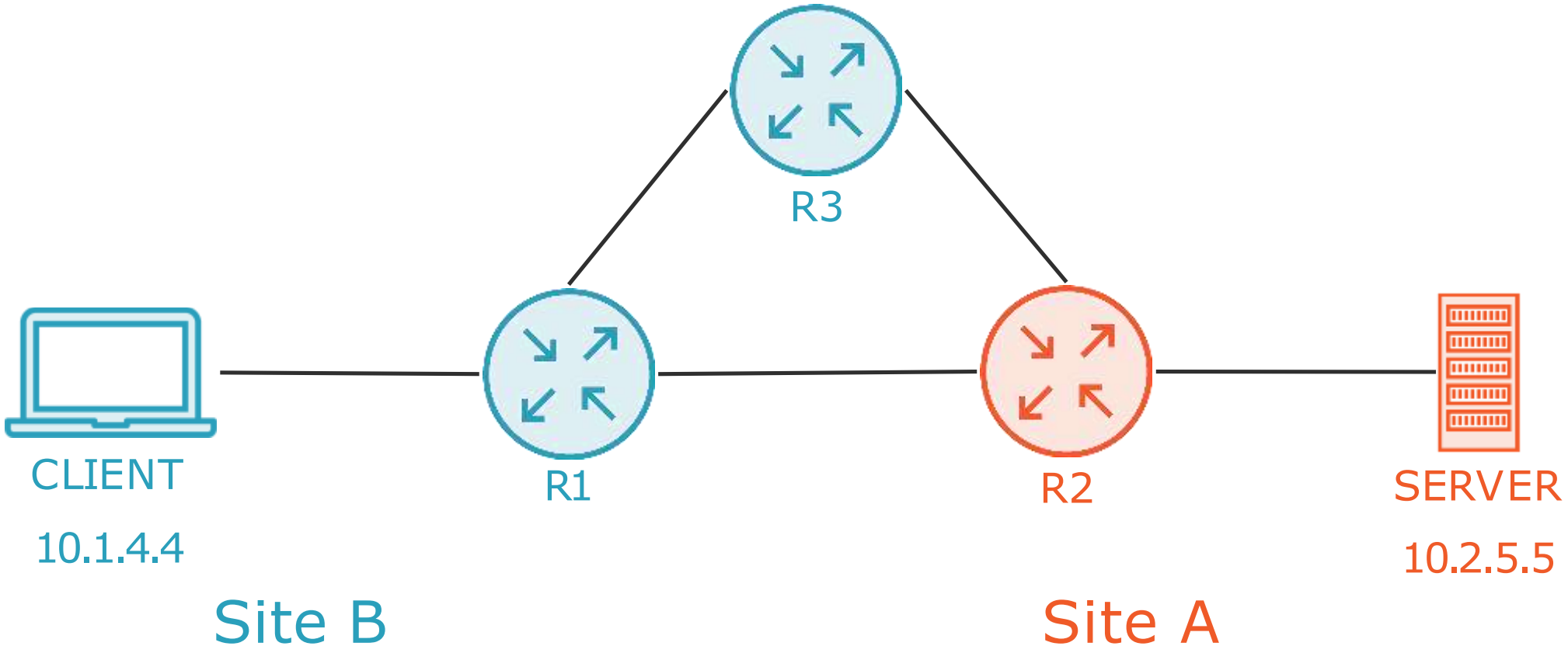
IPv4 et IPv6 sont accompagnés d'autres protocoles comme **ARP**, **ICMP** et **ICMPv6** :

- La couche Internet remplit aussi **le rôle de résolution d'adresses** a l'aide du protocole **ARP** en IPv4 et **Neighbor Discovery (ND)** en IPv6.
- IPv4 dispose d'**ICMP** et IPv6 d'**ICMPv6** pour le **diagnostic** et les **messages d'erreurs**.
- IPv4 et IPv6 sont aidés par des **protocoles de routage** pour maintenir le routage Internet (**RIP, OSPF, BGP, EIGRP**)

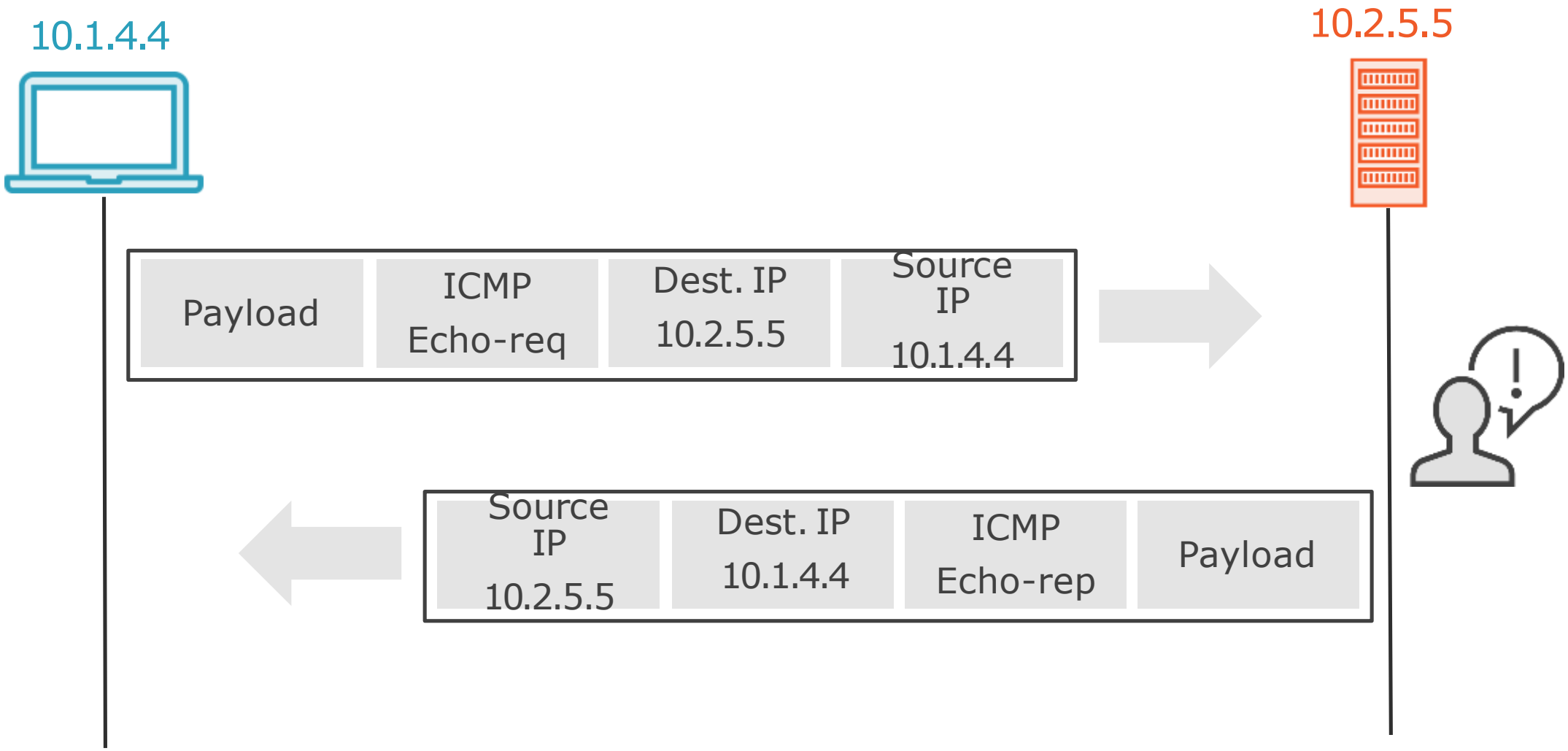
Protocoles de la couche Internet

ICMP - Internet Control Message Protocol

- **ICMP** est l'un des protocoles fondamentaux constituant la suite de protocoles Internet.
- Il est utilisé pour **véhiculer des messages de contrôle et d'erreur** pour la pile de protocoles TCP/IP.
- Comme le **protocole IP** ne gère que l'acheminement des paquets et ne permet pas l'envoi de messages d'erreur, c'est grâce au protocole **ICMP** qu'une machine émettrice peut savoir qu'il y a eu un incident de réseau.
- Divers services réseau, souvent utilisés comme **Traceroute** ou **Ping**, sont basés sur le protocole **ICMP**.


R3
CLIENT
10.1.4.4
R1
R2
SERVER
10.2.5.5
Site B
Site A

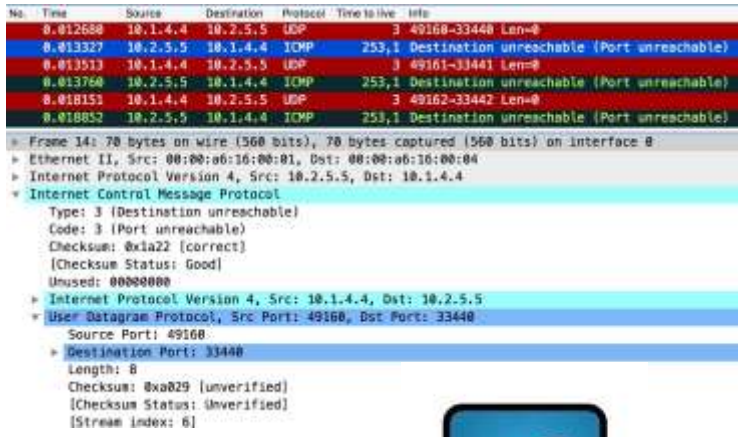
Flux de paquets Ping



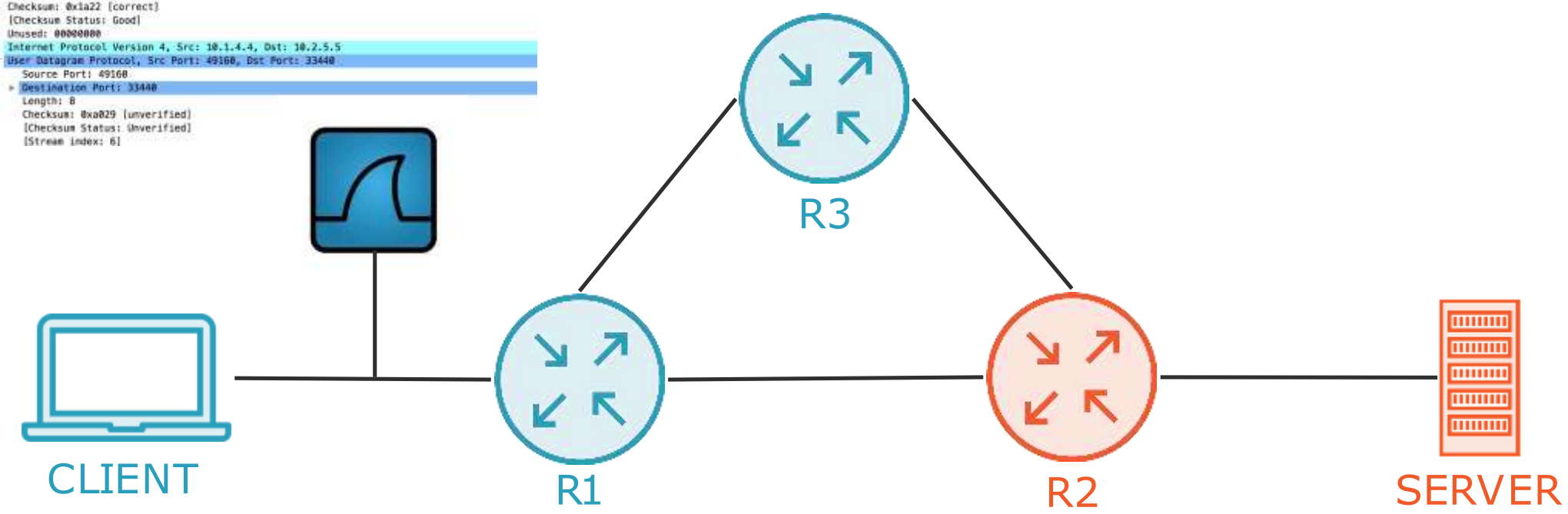
```
CLIENT#
CLIENT#
CLIENT#ping 10.2.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.5.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
CLIENT#
```

```
CLIENT#ping 10.2.5.5
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.5.5, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/3 ms
CLIENT#ping 10.2.5.6
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.2.5.6, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
CLIENT#
```

Wireshark



Free download:
wireshark.org



ICMP Echo-request

No.	Time	Source	Destination	Protocol	Info
→ 1	0.000000	10.1.4.4	10.2.5.5	ICMP	Echo (ping) request id=0x0005, seq=0/0, ttl=255 (reply in 2)
← 2	0.000627	10.2.5.5	10.1.4.4	ICMP	Echo (ping) reply id=0x0005, seq=0/0, ttl=253 (request in 1)
3	0.000741	10.1.4.4	10.2.5.5	ICMP	Echo (ping) request id=0x0005, seq=1/256, ttl=255 (reply in 4)
4	0.000996	10.2.5.5	10.1.4.4	ICMP	Echo (ping) reply id=0x0005, seq=1/256, ttl=253 (request in 3)
5	0.001078	10.1.4.4	10.2.5.5	ICMP	Echo (ping) request id=0x0005, seq=2/512, ttl=255 (reply in 6)
6	0.001283	10.2.5.5	10.1.4.4	ICMP	Echo (ping) reply id=0x0005, seq=2/512, ttl=253 (request in 5)
7	0.006586	10.1.4.4	10.2.5.5	ICMP	Echo (ping) request id=0x0005, seq=3/768, ttl=255 (reply in 8)
8	0.007057	10.2.5.5	10.1.4.4	ICMP	Echo (ping) reply id=0x0005, seq=3/768, ttl=253 (request in 7)
9	0.012692	10.1.4.4	10.2.5.5	ICMP	Echo (ping) request id=0x0005, seq=4/1024, ttl=255 (reply in 10)
	0.013141	10.2.5.5	10.1.4.4	ICMP	Echo (ping) reply id=0x0005, seq=4/1024, ttl=253 (request in 9)

```
▶ Frame 1: 114 bytes on wire (912 bits), 114 bytes captured (912 bits) on interface 0
▶ Ethernet II, Src: 00:00:a6:16:00:04, Dst: 00:00:a6:16:00:01
▶ Internet Protocol Version 4, Src: 10.1.4.4, Dst: 10.2.5.5
▼ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0xe180 [correct]
    [Checksum Status: Good]
    Identifier (BE): 5 (0x0005)
    Identifier (LE): 1280 (0x0500)
    Sequence number (BE): 0 (0x0000)
    Sequence number (LE): 0 (0x0000)
    [Response frame: 2]
  ▶ Data (72 bytes)
```

← Type 8 is echo-request (no codes)

ICMP Echo-reply

No.	Time	Source	Destination	Protocol	Info
→ 1	0.000000	10.1.4.4	10.2.5.5	ICMP	Echo (ping) request id=0x0005, seq=0/0, ttl=255 (reply in 2)
← 2	0.000627	10.2.5.5	10.1.4.4	ICMP	Echo (ping) reply id=0x0005, seq=0/0, ttl=253 (request in 1)
3	0.000741	10.1.4.4	10.2.5.5	ICMP	Echo (ping) request id=0x0005, seq=1/256, ttl=255 (reply in 4)
4	0.000996	10.2.5.5	10.1.4.4	ICMP	Echo (ping) reply id=0x0005, seq=1/256, ttl=253 (request in 3)
5	0.001078	10.1.4.4	10.2.5.5	ICMP	Echo (ping) request id=0x0005, seq=2/512, ttl=255 (reply in 6)
6	0.001283	10.2.5.5	10.1.4.4	ICMP	Echo (ping) reply id=0x0005, seq=2/512, ttl=253 (request in 5)
7	0.006586	10.1.4.4	10.2.5.5	ICMP	Echo (ping) request id=0x0005, seq=3/768, ttl=255 (reply in 8)
8	0.007057	10.2.5.5	10.1.4.4	ICMP	Echo (ping) reply id=0x0005, seq=3/768, ttl=253 (request in 7)
9	0.012692	10.1.4.4	10.2.5.5	ICMP	Echo (ping) request id=0x0005, seq=4/1024, ttl=255 (reply in 10)
	0.013141	10.2.5.5	10.1.4.4	ICMP	Echo (ping) reply id=0x0005, seq=4/1024, ttl=253 (request in 9)

```
▶ Frame 2: 114 bytes on wire (912 bits), 114 bytes captured (912 bits) on interface 0
▶ Ethernet II, Src: 00:00:a6:16:00:01, Dst: 00:00:a6:16:00:04
▶ Internet Protocol Version 4, Src: 10.2.5.5, Dst: 10.1.4.4
▼ Internet Control Message Protocol
    Type: 0 (Echo (ping) reply)
    Code: 0
    Checksum: 0xe980 [correct]
    [Checksum Status: Good]
    Identifier (BE): 5 (0x0005)
    Identifier (LE): 1280 (0x0500)
    Sequence number (BE): 0 (0x0000)
    Sequence number (LE): 0 (0x0000)
    [Request frame: 1]
    [Response time: 0.627 ms]
  ▶ Data (72 bytes)
```

← Type 0 is echo-reply (no codes)

Types/Codes - ICMP

Type 8: ICMP echo-request

Code 0: No code

Type 0: ICMP echo-reply

Code 0: No code

Type 3: Destination unreachable

Code 0: Network unreachable

Code 1: host unreachable

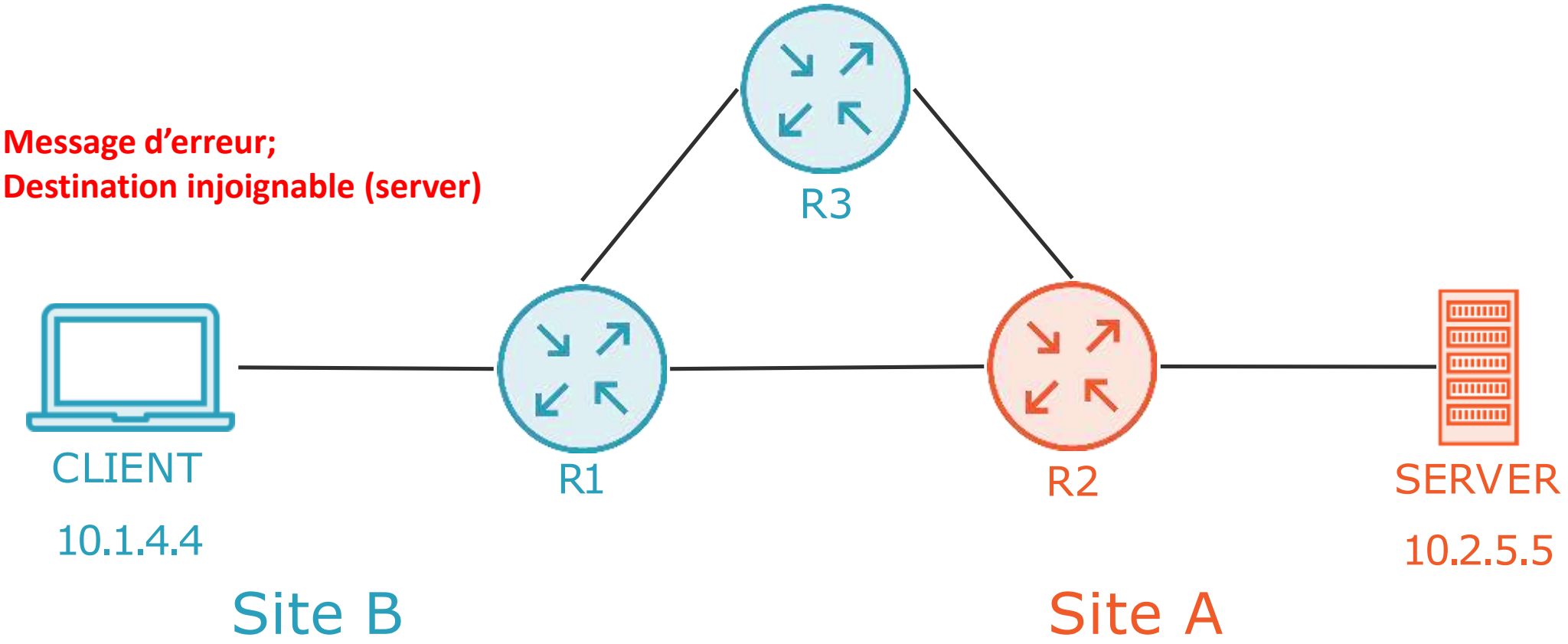
Code 4: Packet Too Big

Code 13: Admin prohibited

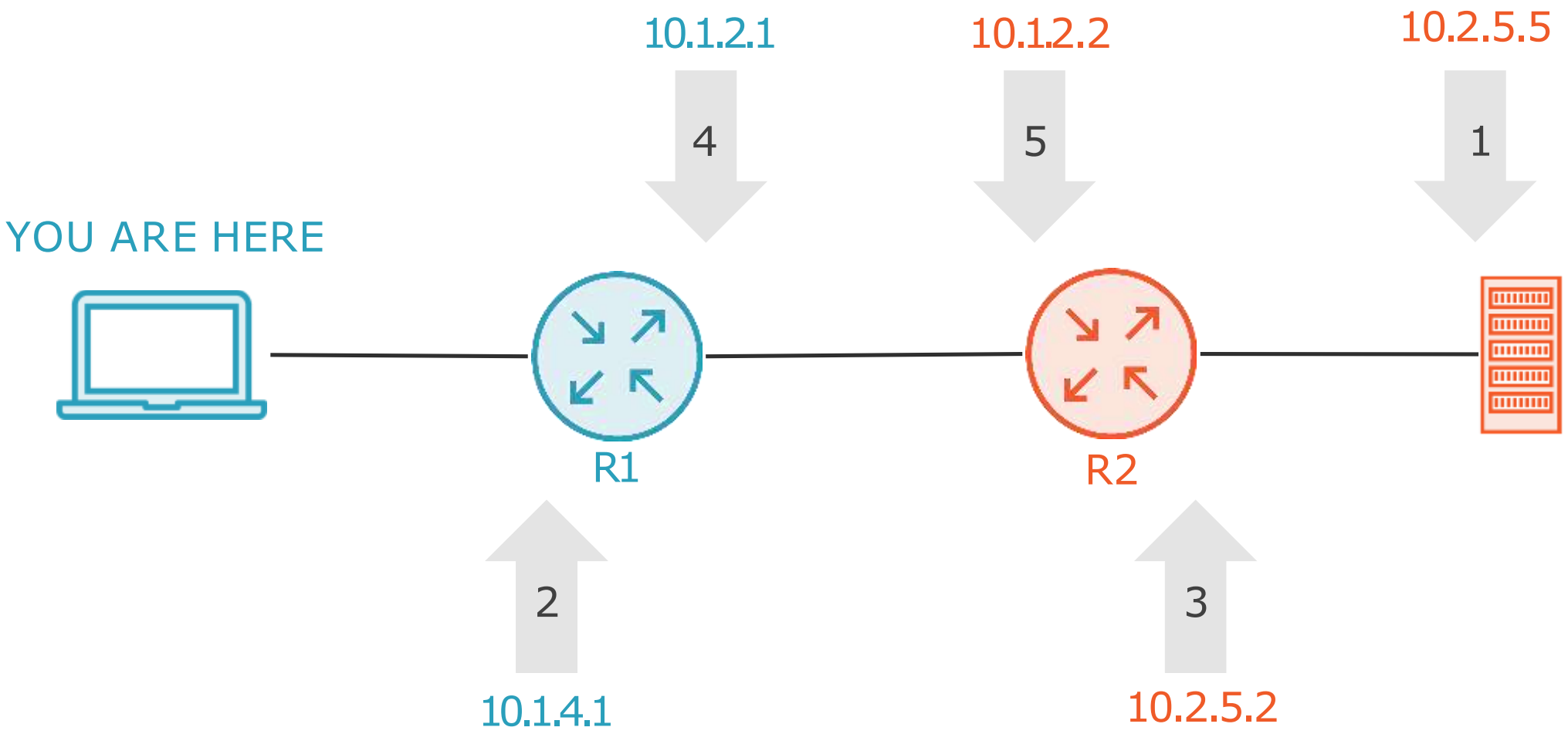
… Code 15!

https://www.frameip.com/entete-icmp/

Comment diagnostiquer avec l'utilitaire Ping?



Diagnostiqué avec le Ping



Protocoles de la couche Internet

ICMP - Internet Control Message Protocol

Un paquet ICMP est néanmoins encapsulé dans un **datagramme IP**.

- Dans le cadre de l'IPv4, la forme génerale d'un tel paquet est la suivante:

En-tête du datagramme ICMP(partie en rose)

Bit 0 - 7	Bit 8 - 15	Bit 16 - 23	Bit 24 - 31
Version/IHL	Type de service	Longueur totale	
Identification (fragmentation)		*flags* et *offset* (fragmentation)	
Durée de vie(TTL)	Protocole	Somme de contrôle de l'en-tête	
Adresse IP source			
Adresse IP destination			
Type de message	Code	Somme de contrôle	
Bourrage ou données			
Données (*optionnel et de longueur variable*)			

En-tête du datagramme ICMP

Un tel datagramme est composé :

- d'un en-tête IP (en bleu), avec Protocole valant 1 (ICMPv4) et Type de Service valant 0.
- du type de message ICMP (8 bits)
- du code de l'erreur (8 bits)
- d'une somme de contrôle (16 bits), calculée sur la partie spécifique à ICMP (sans l'en-tête IP)
- d'une partie aménagée pour des données relatives aux différents types de réponses (32 bits) , si elle n'est pas utilisée, on procède à un bourrage (cette partie peut correspondre aux Identifiant et Numéro de séquence pour un paquet de type Ping par exemple.
- du message

Table des types et codes ICMP

type 0	code=0	echo reply (requête)
type 3		destination unreachable (messages d'erreurs)
	code 0	network unreachable
	code 1	host unreachable
	code 2	protocol unreachable
	code 3	port unreachable
	code 4	fragmentation needed but don't-fragment bit set
	code 5	source route failed
	code 6	destination network unknown
	code 7	destination host unknown
	code 8	source host isolated (obsolete)
	code 9	destination network administratively prohibited
	code 10	destination host administratively prohibited
	code 11	network unreachable for TOS
	code 12	host unreachable for TOS
	code 13	communication administratively prohibited by filtering
	code 14	host precedence violation
	code 15	precedence cutoff in effect
type 4	code=0	source quench (contrôle de flux) (messages d'erreurs)

Table des types et codes ICMP

type 5		redirect (messages d'erreurs)
	code 0	redirect for network
	code 1	redirect for host
	code 2	redirect for type-of-service and network
	code 3	redirect for type-of-service and host
type 8	code=0	echo request (ping) (requête)
type 9	code=0	router advertisement (requête)
type 10	code=0	router solicitation (requête)
type 11		TTL (messages d'erreurs)
	code 0	TTL = 0 during transit
	code 1	TTL = 0 during reassembly
type 12		parameter problem (messages d'erreurs)
	code 0	IP header bad
	code 1	required option missing
type 13	code=0	timestamp request
type 14	code=0	timestamp reply
type 17	code=0	address mask request
type 18	code=0	address mask reply

Table des types et codes ICMP

Ci-dessous un lien qui décrive les différents types et codes ICMP.

https://www.frameip.com/entete-icmp/

Plan

- Présentation du Modèle TCP/IP
  - Modèle TCP/IP vs OSI
  - Les couches TCP/IP
  - La couche Transport
    - TCP & UDP (TP)
  - la couche Internet
    - Les protocoles de la couche Internet
    - IP, ICMP, …(TP)
  - La couche Application
    - Les protocoles d'Applications
    - DNS, HTTP, … (TP)

La couche Application

La couche application est l'interface entre l'utilisateur et le réseau.

Cette couche est directement en contact avec différentes applications et propose divers services et protocoles.

Les protocoles de la couche Application

Voici certains de protocoles applicatifs:

HTTP (Hypertext Transfer Protocol) : le protocole utilisé pour la transmission de pages HTML.

HTTPS (Hypertext Transfer Protocol Secure) : la version chiffrée du protocole HTTP.

FTP (File Transfer Protocol) : le protocole qui permet l'échange de données entre deux ordinateurs, même si leur conception et leur système d'exploitation sont différents.

TFTP (Trivial File Transfer Protocol) : est un protocole de transfert de fichiers *simplifié* qui ne nécessite aucun mécanisme d'authentification ni de cryptage. TFTP est basé sur UDP.

Les protocoles de la couche Application

SMTP (Simple Mail Transfer Protocol) : Le protocole qui permet l’échange d’emails entre deux ordinateurs.

DNS (Domain Name System) : le protocole qui transforme les domaines en adresses IP.

Telnet (Telecommunication Network) : le protocole qui permet l'accès à distance à un ordinateur via un terminal virtuel.

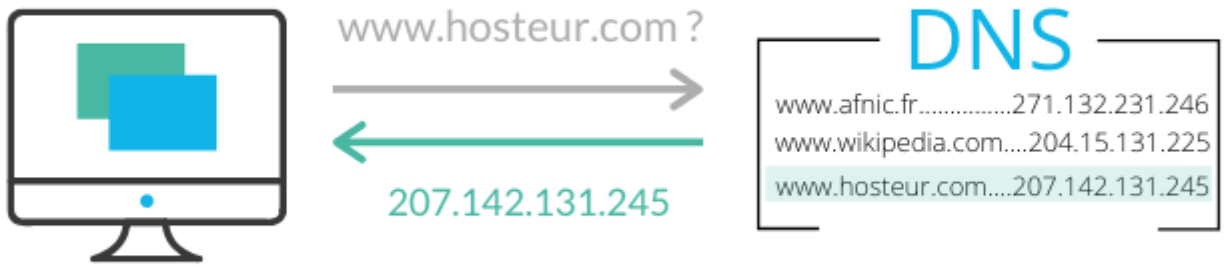
SSH (Secure Shell) : le protocole réseau qui permet aux administrateurs d'accéder à distance à un ordinateur, en toute sécurité.

Système de Nom DNS


Domain Name
IP Address
www.
52.201.108.115
.com
23.50.225.35
www.
.org
.co.id

Qu'est-ce que le DNS ?

- Le DNS (Domain Name System, système de nom de domaine) est en quelque sorte le répertoire téléphonique d'Internet;
- c’est a dire un répertoire (Annuaire) de noms de domaine traduits en adresses IP.



Qu'est-ce que le DNS ?

- Les internautes accèdent aux informations en ligne via des noms de domaine (par exemple, nytimes.com ou espn.com), tandis que les navigateurs interagissent par le biais d'adresses IP (Internet Protocol, protocole Internet).
- Le DNS traduit les noms de domaine en adresses IP afin que les navigateurs puissent charger les pages web.

Qu'est-ce que le DNS ?

- Chaque appareil connecté à Internet dispose d'une adresse IP unique que les autres appareils utilisent afin de le trouver.
- Grâce aux serveurs DNS, les internautes n'ont pas à mémoriser les adresses IP (par exemple, 192.168.1.1 en IPv4) ni les adresses IP alphanumériques plus récentes et plus complexes (par exemple, 2400:cb00:2048:1::c629:d7a2 en IPv6).

Comment fonctionne le DNS ?

- Le processus de résolution DNS implique la conversion d'un nom d'hôte (par exemple, www.amazon.com) en adresse IP « au format informatique » (par exemple, 192.0.2.44).

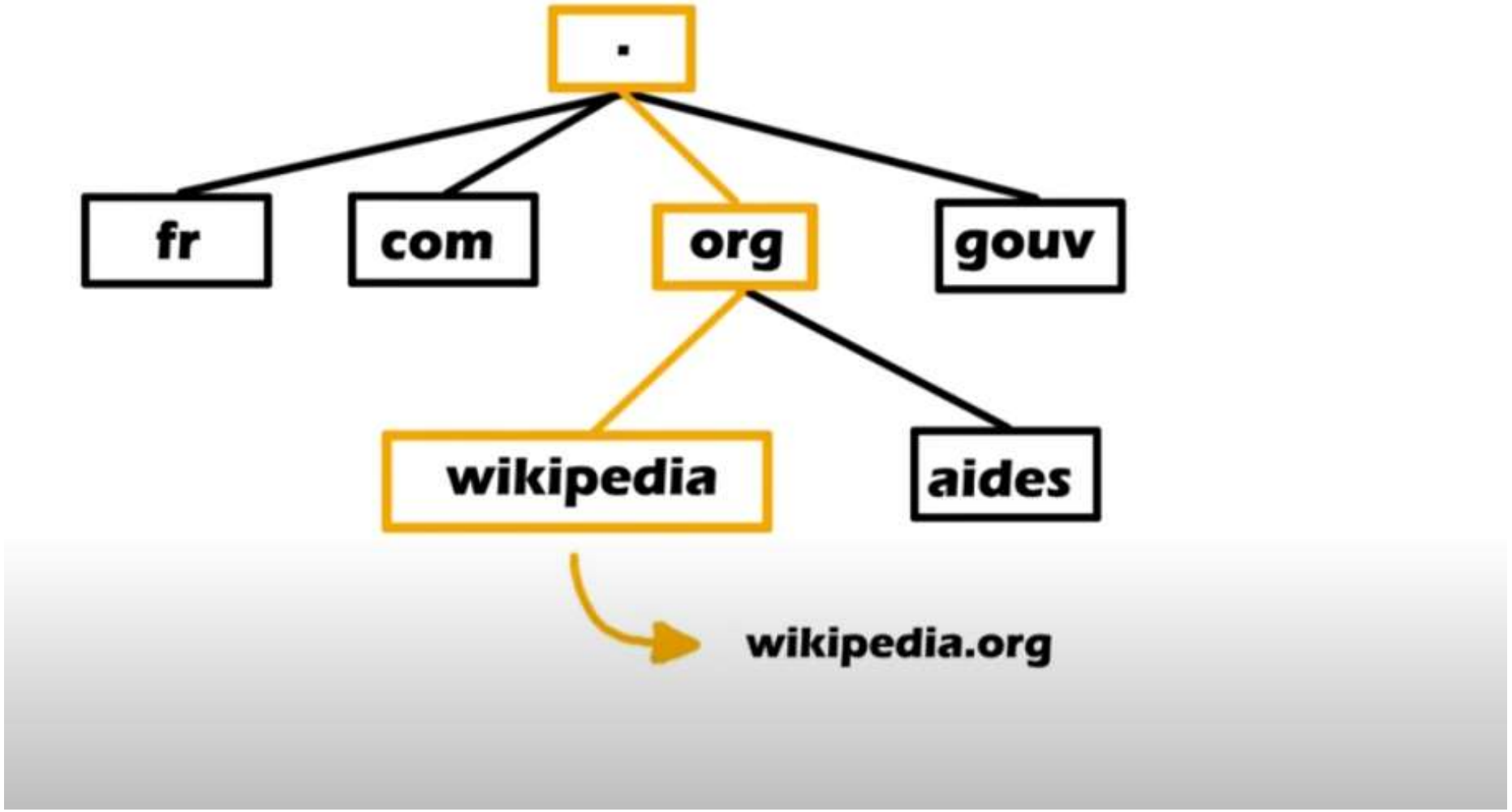
Système DNS

- FQDN (fully qualified domain name) est l'adresse complète d'un hôte Internet ou d'un ordinateur.

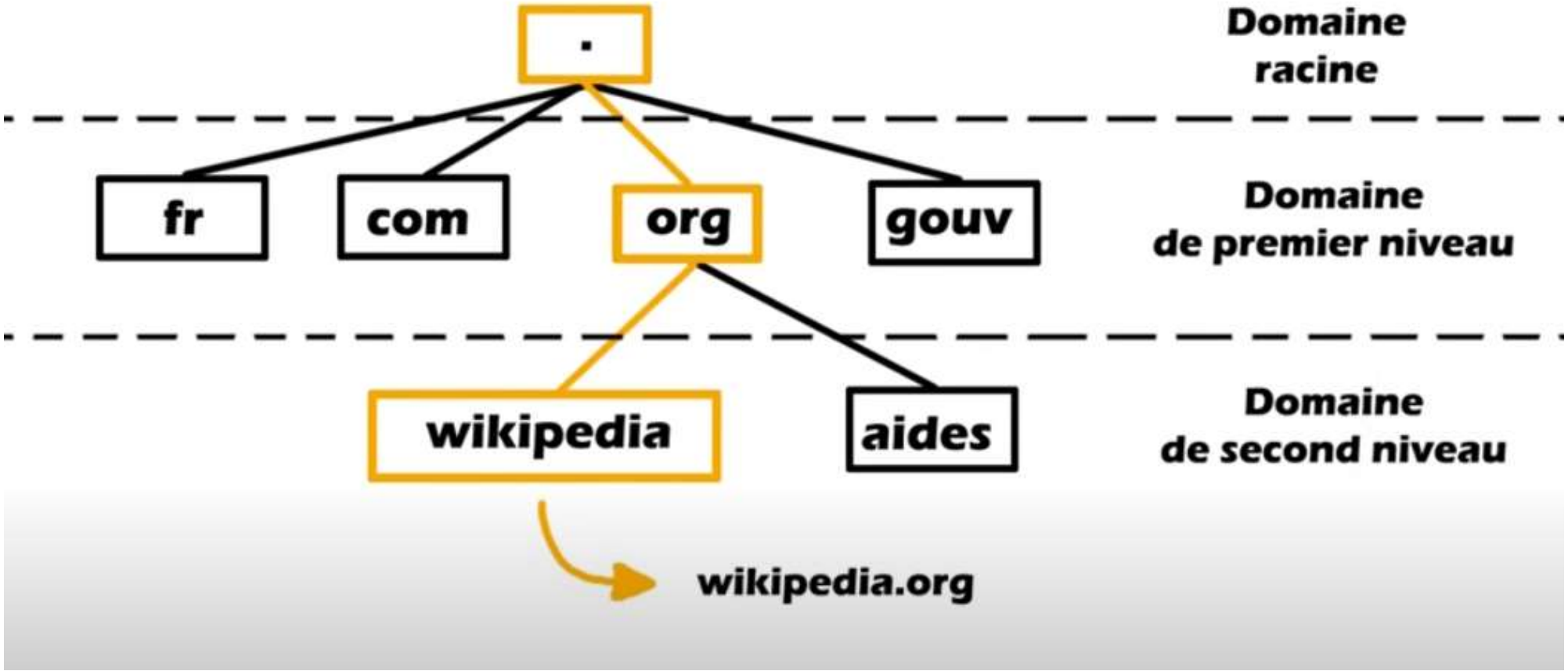
- Exemple ??

Exemples de Noms de Domaine	Adresses IP
dns.google.com	8.8.8.8
app.pluralsight.com	104.18.114.44
www.globomantics.com	52.88.138.56
linkedin.com	2620:109:c002::6cae:a0a

Comment ça marche



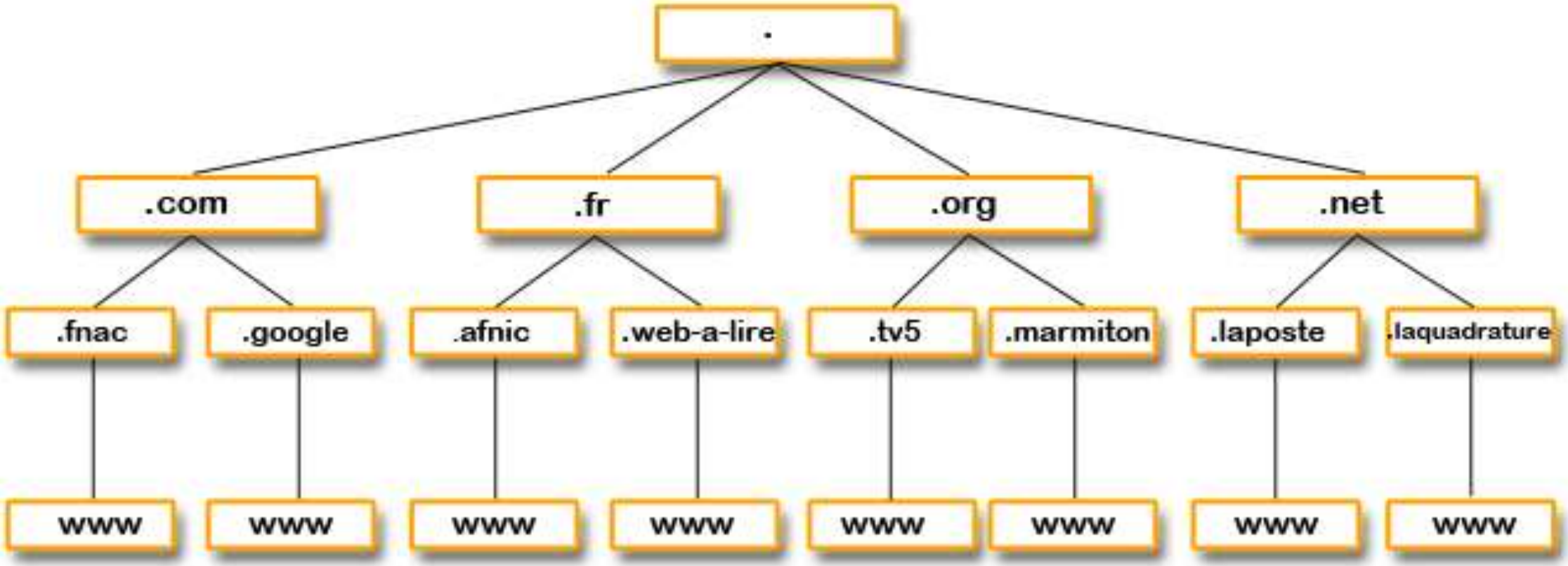
Système de Nom de Domaine



Comment ça marche

- Le système de nom de domaine représente une hiérarchie de domaines ;
  - domaine racine représenté par un point < . > , n'est pas visible au niveau du site web
  - Domaine de 1ier niveau (Ex; .com, .org, .gouv, ...)
  - Domaine de second niveau (Ex; wikipedia, google, ...)
  - possibilité d'avoir des Sous-Domaines (Ex des langues: en, fr, ...)

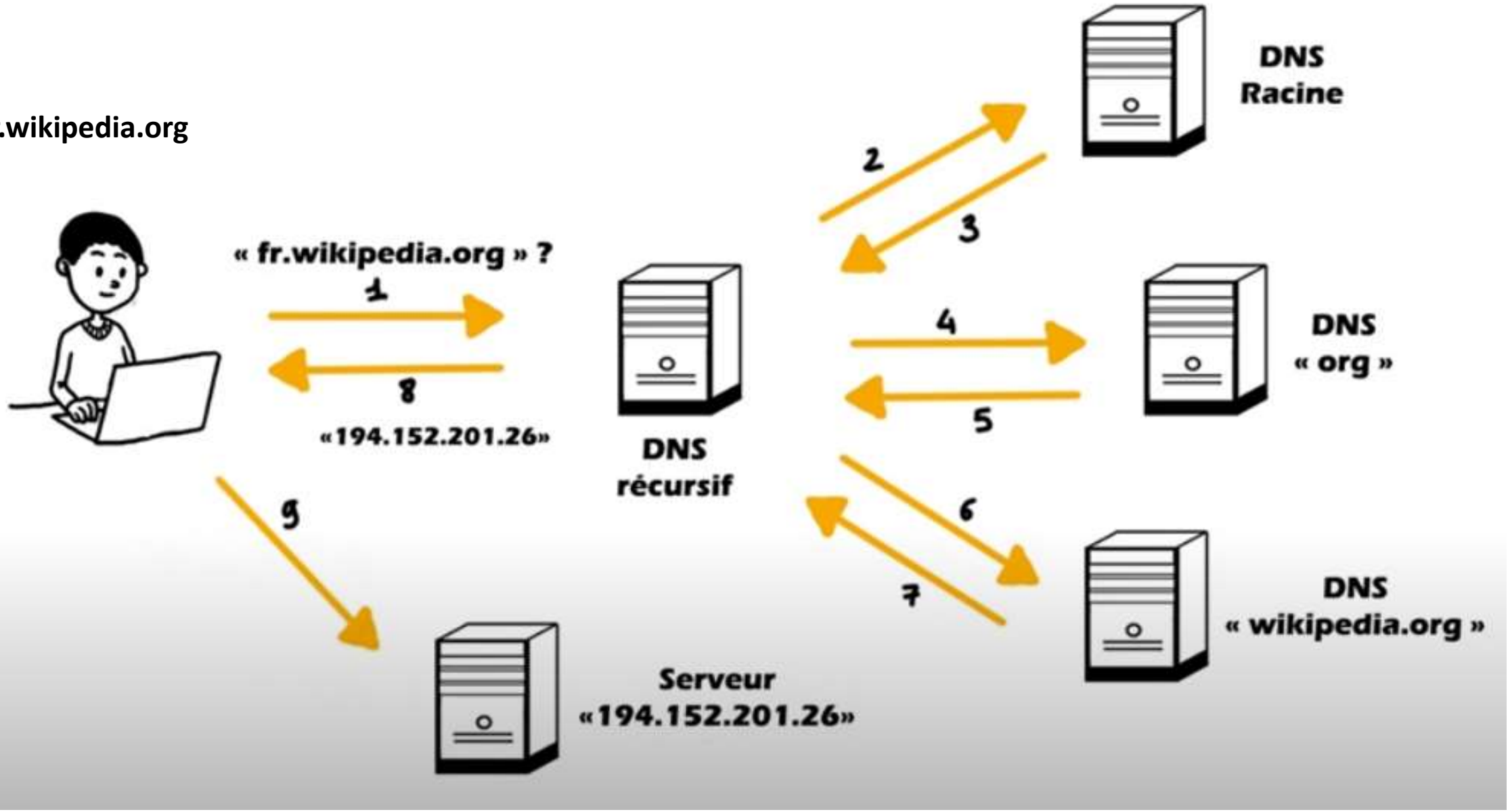
✓la résolution des noms de domaine (FQDN) se réalise du haut vers le bas; c'est a dire du domaine racine vers le domaine de bas niveau.


.
.com
.fr
.org
.net
.fnac
.google
.afnic
.web-a-lire
.tv5
.marmiton
.laposte
.laquadrature
www
www
www
www
www
www
www
www

La résolution des noms de domaine

La résolution des noms de domaine (FQDN) se réalise du haut vers le bas; c’est a dire du domaine racine vers le domaine de bas niveau.

fr.wikipedia.org



La résolution des noms de domaine

- Serveur DNS récursif est le premier serveur interrogé,
  - Il est le serveur DNS fournit par le Fournisseur d'Internet - FAI
  - Il existe des serveurs DNS récursif ouvert
    - Ex: Google DNS , OPenDNS, ...

En premier temps le Serveur DNS récursif cherche dans son annuaire;

- si l'adresse IP du serveur demandé existe, il la renvoie a l'utilisateur,
- si l'adresse n'existe pas il interrogera le serveur DNS racine, ...

Les enregistrements des DNS

- Un serveur DNS possède une base de données dans laquelle se trouve toute une série d'enregistrements.
- Types d'enregistrements :
  - SOA le serveur qui a l'autorité administrative
  - NS les serveurs de nom primaire et secondaire
  - MX le serveur de messagerie
  - A pour la correspondance nom → adresse
  - PTR pour la correspondance adresse → nom
  - CNAME pour les alias (www, ftp, mail, news, etc.)
  - D'autres enregistrements exotiques

La commande Nslookup

Nslookup est un outil qui permet *d'interroger le serveur DNS* afin d'obtenir des informations concernant *les nom de domaine, adresses IP et d'autre enregistrement DNS.*

Protocole HTTP

Principe du protocole HTTP

- Le protocole HTTP, pour HyperText Transfer Protocol, est un protocole de communication client-serveur qui permet d'accéder à des ressources (pages Web) situées sur un serveur Web.
- Aujourd'hui, on lui préfère le HTTPS, dont le *S* signifie *Secure*, il s'agit d'une variante sécurisée du protocole HTTP et qui s'appuie sur *les protocoles SSL/TLS* pour chiffrer les échanges entre le client et le serveur.

Principe du protocole HTTP

- Pour communiquer avec un serveur Web au travers du protocole HTTP, on s'appuiera sur un client HTTP.
- Au quotidien, ce client HTTP prend la forme d'un navigateur Internet (Firefox, Google Chrome, Safari, etc...).

Les différentes versions de HTTP

Les requêtes et les réponses HTTP

- Lorsqu'un client communique avec un serveur au travers du protocole HTTP, il émet une requête HTTP à destination du serveur.

Exemple d’une requête HTTP

Exemple d’une requête HTTP

Une requête HTTP contient :

- **une méthode** (ex: GET),
- **une cible** ex:index.php
- et la version du protocole utilisé.

En complément, la requête HTTP contient un en-tête HTTP avec différents champs qui indiquent l'hôte cible, le langage, etc...

➢ D’autres méthodes existent, comme exemple la *méthode POST*.

REFERENCES

- https://docs.oracle.com/cd/E38898_01/pdf/E38852.pdf
- https://wiki.wireshark.org/CaptureFilters

- https://www.youtube.com/watch?v=bRsET43i1o4
- https://www.youtube.com/@Cookieconnecte
- https://www.it-connect.fr/le-protocole-http-pour-les-debutants/
- https://www.ionos.fr/digitalguide/hebergement/aspects-techniques/definition-protocole-http/

Bonne Chance