IAE LIMOGES
École Universitaire
de Management

Master CCA 2nde année

"On the Internet, nobody knows you're a dog."

Réseau et Système d'Information — P-F. Bonnefoi

Version du 26 janvier 2023

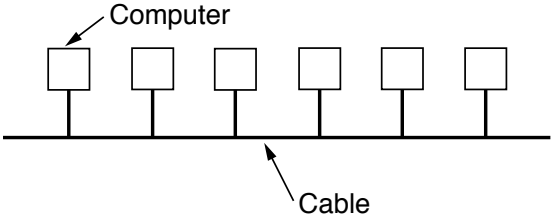
Plan de la partie Fondamentaux

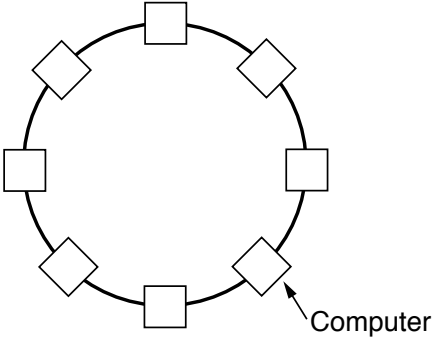
Éléments importants

- ▷ Deux topologies théoriques : diffusion et «point-à-point» ;
- ▷ Les réseaux utilisés et le matériel d'interconnexion ;
- ▷ La gouvernance d'Internet : les organisations et les RFCs ;
- ▷ Le réseau TCP/IP : adressage, encapsulation, routage direct & indirect ;
- ▷ Le DNS : global et local ;
- ▷ La configuration du poste de travail.

Le réseau en mode «diffusion»

Les réseaux à diffusion, *broadcast network*, n'ont qu'un **seul canal de communication** que toutes les machines partagent.





Une machine envoie de **petits messages** qui sont reçus par toutes les autres machines :

- ⋆ dans le message un **champ d'adresse** permet d'identifier le destinataire
- ⋆ à la réception du message, une machine teste ce champ :
  - ◇ si le message est pour elle, elle le traite ;
  - ◇ sinon, elle l'ignore.

Exemple :

un couloir sur lequel débouche un certain nombre de portes de bureau.
quelqu'un sort dans le couloir et appelle une personne,
tout le monde entend l'appel mais une seule personne répond à l'appel
cas des annonces dans les gares ou les aéroports

Dans le cas d'Ethernet, mais aussi de WiFi, de Bluetooth…

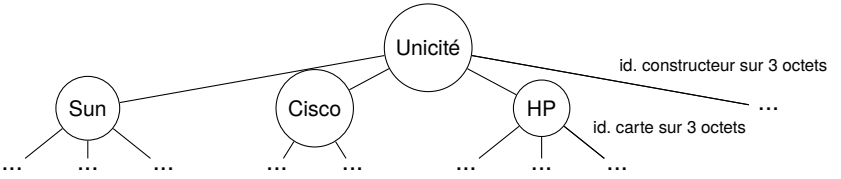
Chaque carte réseau possède une adresse matérielle appelée adresse MAC (Medium Access Control) :

- ▷ **unique** par rapport à toutes les cartes réseaux existantes !
- ▷ **exprimée sur 48 bits** ou 6 octets.
  - ◇ **Syntaxe** : `08:22:EF:E3:D0:FF`
  - ◇ **Adresse de Broadcast** : FF:FF:FF:FF:FF:FF (en IPv4).

Pour garantir l'**unicité** :

a. des «tranches d'adresses» sont affectées aux différents constructeurs :

00:00:0C:XX:XX:XX	**Cisco**
08:00:20:XX:XX:XX	**Sun**
08:00:09:XX:XX:XX	**HP**
00:09:BF:XX:XX:XX	**Nintendo**
00:D0:F1:XX:XX:XX	**Sega**

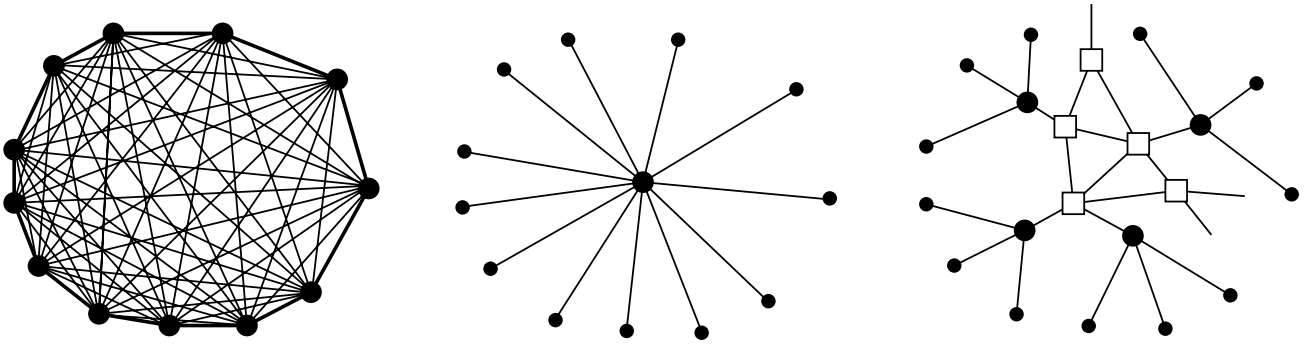


Ce préfixe est appelé OUI, «Organization Unique Identifier».
La liste est consultable à `http://standards.ieee.org/regauth/oui/index.shtml`.

b. **chaque constructeur** numérote différemment chaque carte réseau qu'il construit.

Avantage	**impossible de trouver deux fois** la même adresse dans un même réseau
Inconvénient	elle ne donne **aucune information sur la localisation** d'une machine «*dans quel réseau est la machine avec qui je veux parler ?*»
Solution	**utilisation de l'adresse IP !**

Réseaux en mode «point à point»



(a) (b) (c)

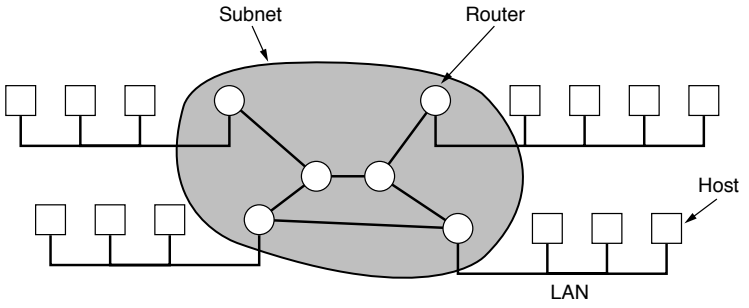
Ces réseaux sont formés d'un **grand nombre de connexions** entres les machines prises **deux à deux**.

Le trajet des communications est rendu plus complexe :

- ▷ pour aller de la source au destinataire, un message doit alors **passer par un plusieurs intermédiaires**.
- ▷ il existe plusieurs routes de **longueurs différentes** pour joindre ces deux machines, il est nécessaire d'utiliser de **bons algorithmes d'acheminement des messages** ;

Inconvénient

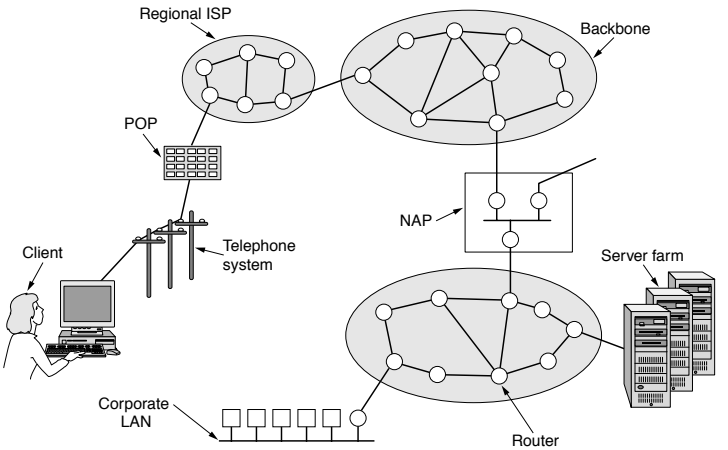
- ▷ le **temps de transfert** d'un message devient presque impossible à prévoir.
- ▷ il est nécessaire de faire du **routage** des messages dans le réseau.



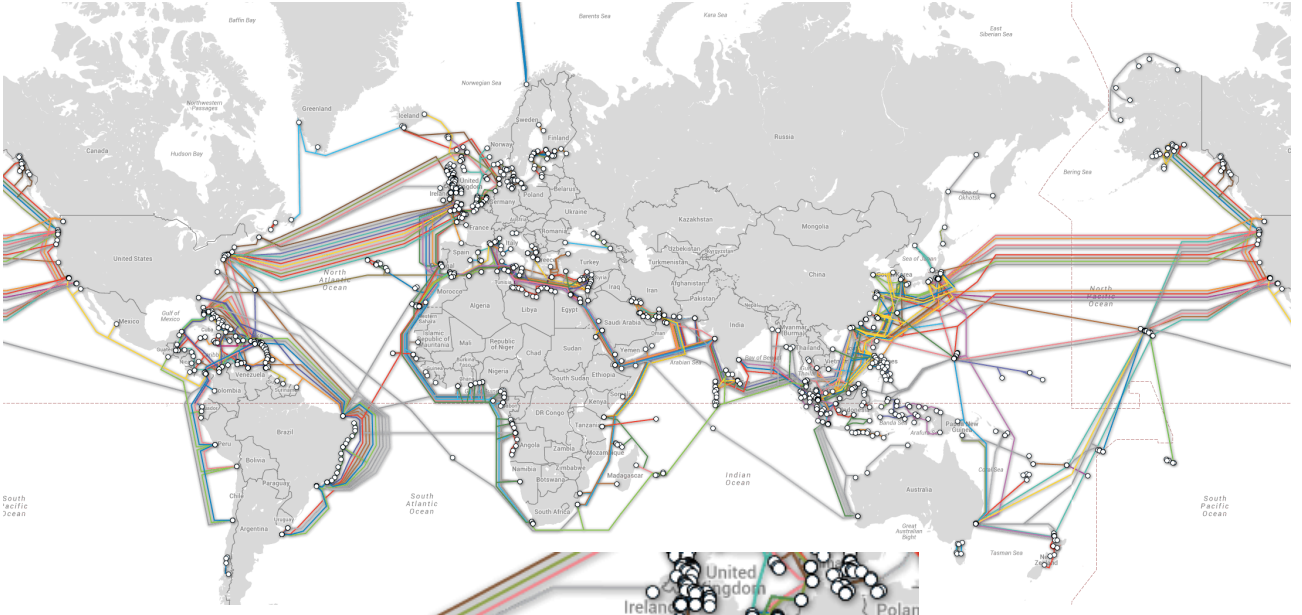
- ▷ **diffusion** : réseau de petite taille, LAN, *Local Area Network* ; *Exemple :Ethernet*
- ▷ **point-à-point** : **réseau d'interconnexion**, constitué uniquement de routeur et de ligne de transmission *Exemple : liaison satellite.*
- ▷ la **combinaison des deux** : WAN, *Wide Area Network*.

Inter(connexion)Net(work) :

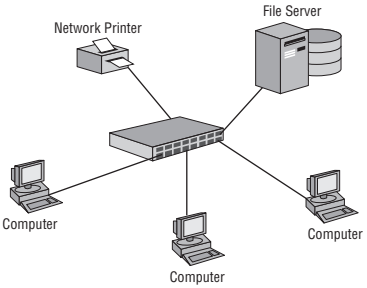
- ⋆ du client à la maison ;
- ⋆ de la ligne téléphonique au POP, «*Point of Presence*» vers ATM ;
- ⋆ ou en passant par de la fibre optique pour une liaison IP directe vers l'ISP ;
- ⋆ en passant par l'ISP, *Internet Service Provider* ;
- ⋆ au réseau national : *backbone* ;
- ⋆ par une connexion à un réseau, *Network Access Point* ;
- ⋆ vers le LAN de l'entreprise…

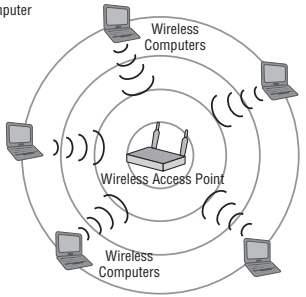


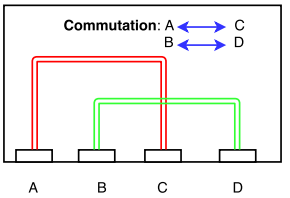
Et au niveau mondial ?

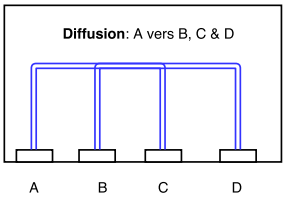


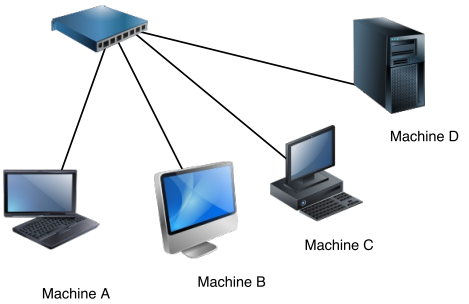
Étoile ou «star»











- la topologie la plus courante pour définir un LAN : plusieurs matériels connectés à un nœud de connexion central :
  - un «hub» : **tout le monde entend** les communications de tout le monde ;
  - un «switch» ou un point d'accès sans fil : capacité de mettre en relation deux matériels voulant communiquer entre eux (commutation ou «switching») : **seuls les deux matériels en communication** entendent ce qu'ils échangent.

  *Sur le schéma à droite, les machines A et C, & B et D peuvent communiquer **simultanément** en mode commutation. La machine A peut aussi **diffuser** un message vers B, C et D.*
  Le switch est un matériel qui peut passer automatiquement du fonctionnement «diffusion» au fonctionnement «commutation» suivant la nature des échanges des machines connectées.

- **InterNIC**, «*Internet Network Information Center*» entre 1992-1998 :
  - organisme public américain chargé de la gestion centrale des adresses et des noms de domaines Internet et de l'accréditation d'un organisme homologue dans chaque pays, les organismes délégués :
    - AfriNIC (Afrique),
    - APNIC (Asie, Pacifique),
    - ARIN (Amérique du Nord),
    - LACNIC (Amérique du Sud, îles Caraïbes),
    - RIPE NCC (Europe, Moyen-Orient)
    - NIC France ou afnic, NIC Angleterre, etc.

- **ICANN**, «*Internet Corporation for Assigned Names and Numbers*» :
  - Organisation créée en octobre 1998, pour s'ouvrir à la concurrence
  - traite les noms de domaine et leur délégation (par exemple VERISIGN Inc. : zone « .com ») ;
  - exploitation des serveurs de la racine du DNS (ceux qui font autorité) ;
  - allocation de blocs de numéro IP ;
  - en France, les prestataires (fournisseurs d'accès) font l'intermédiaire avec l'afNIC

- **IANA**, «*Internet Assigned Numbers Authority*» :
  - tient l'annuaire : adresses IP & numéros de protocoles ;
  - adresses IP et numéros d'AS : déléguées aux RIR régionaux, «Regional Internet Registries» ;
  - numéros de protocoles et de ports (entre 1 et 1023) ;
  - déléguées aux LIRs, «Local Internet Registry» (eg. FAI).

- Les «**Registrar**» :
  - Un registrar (bureau d'enregistrement) est une société ou une association permettant le dépôt de noms de domaine internet, dans les TLD, «Top Level Domain», où il n'y a pas de vente directe.
  - Il faut payer un certain montant pour acquérir et protéger un nom de domaine.

- **GIP Renater** (Groupement d'Intérêt Public) :
  - Réseau de la recherche en France, «Réseau National de télécommunications pour la Technologie l'Enseignement et la Recherche»

La conception

Contraintes du protocole IP «Internet Protocol» RFC 791 :

- ⋆ utiliser la topologie réseau **point à point** (pour permettre de franchir des grandes distances c'est obligatoire) ;
- ⋆ la panne d'un équipement du **sous-réseau d'interconnexion** ne doit pas entraîner une rupture du réseau ;
- ⋆ privilégier la **disponibilité** du réseau : il doit servir au **maximum**.

Le but est que les **échanges persistent** :

- ▷ du moment que l'ordinateur **source** et l'ordinateur **destination fonctionnent** ;
- ▷ même si certains **routeurs** ou certaines **lignes de transmission** tombent en **panne** (origine militaire de la création d'Internet par le DoD, «*Department of Defense*», états-uniens).

Les moyens

- ◦ privilégier la **décentralisation** : pas de nœud central, redondance et distribution des informations nécessaires au fonctionnement du réseau (routeurs, DNS par exemple) ;
- ◦ privilégier le côté **dynamique** : chaque appareil connecté au réseau recherche/découvre tout le temps les matériels nécessaires à sa communication (routeurs, lignes de transmission, chemins, *etc.*) ;
- ◦ définir une **architecture très souple** pour pouvoir mettre en œuvre des applications très diverses comme le transfert de fichiers ou la transmission de la parole en temps réel (TCP et UDP) ;
- ◦ faciliter le **routage** : construire une méthode simple et rapide (opérations binaires par exemple) ;
- ◦ permettre le **regroupement de machines** pour les gérer ensemble (regroupement en réseau) ;
- ◦ faciliter le travail de l'administrateur (*sisi…*).

Adressage pour le protocole IP (IPv4)

Chaque ordinateur et chaque routeur du réseau Internet possède une adresse IP.

L'adresse IP est une **adresse binaire** composées de deux parties `<id. réseau><id. machine>`:

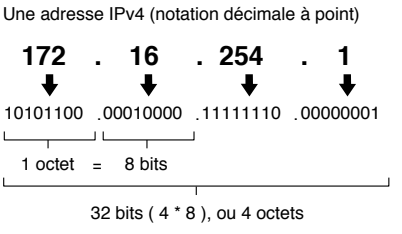
- ⋆ un **identifiant de réseau** ;
- ⋆ un **identifiant machine** pour la distinguer dans le réseau.
  Chaque adresse IP doit être unique pour permettre de la localiser sur la planète.

- ⋆ Il existent **différentes répartitions** des 32 bits entre identifiant réseau et identifiant machine :
  - ◇ ces **différentes répartitions** définissent un ensemble de **classes de réseaux** ;
  - ◇ ces classes **ne sont plus utilisées** en CIDR, où on indique uniquement le nombre de bits de la partie réseau ;

Propriétés

- ▷ Codée sur **32 bits.**

- ▷ Représentée par **commodité** en «*décimale pointée*» : 4 entiers variant entre 0 et 255 séparés par des points
  exemple : 164.81.1.4



- ▷ un **organisme officiel**, le NIC, «*Network Information Center*», est seul habilité à délivrer des numéros d'identification des réseaux.

- ▷ il y a, **en général**, une **seule adresse IP** par interface réseau.
  Dans le cas d'un routeur interconnectant 2 réseaux différents, il possède une adresse IP pour chacune de ses interfaces connectées à un réseau.

Ces **adresses** permettent :

- ⋆ des envois de messages **multi-destinataires** ;
- ⋆ désigner la **machine courante** ;
- ⋆ désigner le **réseau courant**.

Tout à zéro		L'ordinateur lui-même
Tout à zéro	id. de machine	Un ordinateur sur le réseau lui-même
Tout à 1		Diffusion limitée au réseau lui-même
Id. de réseau	Tout à 1	Diffusion dirigée vers ce réseau
127	Nombre quelconque	Boucle

L'**adresse de «boucle»** (127.X.Y.Z) permet d'effectuer :

- ◇ des communications inter-programme sur la même machine
- ◇ des tests de logiciels réseaux. *Dans ces cas là, les paquets ne sont pas réellement émis sur le réseau.*

D'autres **adresses particulières** :

- ◇ 0.0.0.0 est utilisé par une machine pour connaître sa propre adresse IP lors d'un processus d'amorçage (BOOTP). *Elle devra se procurer une adresse IP par l'intermédiaire d'une autre machine.*
- ◇ 255.255.255.255 est une adresse de diffusion locale car elle désigne toutes les machines du réseau auquel appartient l'ordinateur qui utilise cette adresse⟹ ***pas besoin de connaissance du réseau.***

Réseaux privés, RFC1918

Les adresses pour **réseau privé** ou **intranet** (sans **accès direct** à l'extérieur) :

- ⋆ `10.0.0.0/8` : de `10.0.0.0` à `10.255.255.255` ⟹ *classe A*
- ⋆ `172.16.0.0/12:172.16.0.0` à `172.31.255.255` ⟹ *pas de classe*
- ⋆ `192.168.0.0/16:192.168.0.0` à `192.168.255.255` ⟹ *classe B*

Ces réseaux ne sont pas «routables» !

Faire le point…

- sur un **réseau à datagramme**, il circule…**des datagrammes !** ;
- le **réseau à datagramme** est appelé **réseau IP** : il utilise des algorithmes, des formats de messages définis dans la norme IP, «*Internet Protocol*» ;
- un **réseau à diffusion** fait circuler des messages de format différent : les **trames** (on parle de **trame Ethernet** ou IEEE 802.3) ;
- un datagramme doit emprunter un réseau à diffusion pour atteindre un ordinateur :
  - principe **d'encapsulation** : le datagramme est «inclus» dans une trame Ethernet :
  - à l'**adresse IP** d'une machine doit correspondre l'identifiant de cette machine dans le réseau à diffusion : une **adresse MAC** :
    - l'adresse MAC est attachée à la carte réseau et est choisie par le constructeur de cette carte ;
    - l'adresse IP est choisi par l'administrateur réseau suivant la configuration qu'il veut donner à son réseau ;

Comment faire la correspondance entre @MAC et @IP ?

a. c'est **l'ordinateur** qui connaît l'adresse MAC de sa carte réseau ;

b. c'est **l'ordinateur** qui connaît son adresse IP ;

c. **Qui** peut dire à quelle adresse IP correspond tel adresse MAC ? **L'ordinateur lui même** !

d. **Définition d'un protocole** pour « questionner » les ordinateurs : ARP, «*Address Resolution Protocol*»

Transmission physique des datagrammes IP

La **couche liaison de données** est chargée de :

- ▷ la mise en correspondance des **@IP** avec les **@MAC** des interfaces physiques.
- ▷ l'**encapsulation** des datagrammes IP afin qu'ils puissent être transmis sur un support physique particulier.
  Lorsque le protocole IP doit envoyer un datagramme à un équipement relié à un réseau à diffusion, la couche liaison de donnée doit construire une trame ethernet avec l'@MAC du destinataire.

Correspondance entre adresses physiques, @MAC, et adresses IP, @IP

Le **protocole ARP**, «*Address Resolution Protocol*» fournit une **correspondance dynamique** entre une adresse IP connue et l'adresse matérielle correspondante.

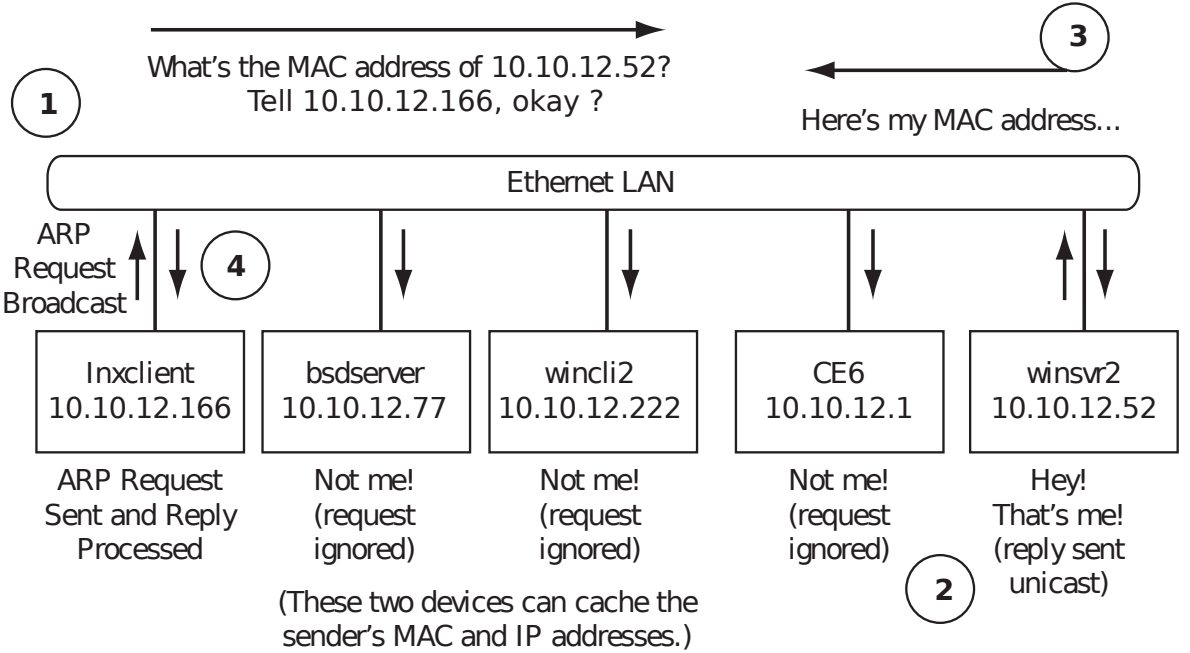
Fonctionnement :

- ▷ ARP dispose d'une **mémoire cache** : lors de la demande de l'@MAC associée à une @IP, ilconsulte sa **mémoire cache ARP** pour voir si l'@IP distante y est mise en correspondance avec @MAC.
  - ◇ Si c'est le cas le datagramme IP est émis immédiatement, enveloppé dans une **trame Ethernet** envoyée à l'adresse physique destination (@MAC).
  - ◇ Sinon la couche liaison de données construit une **requête ARP**.
- ▷ ARP utilise le principe de «*diffusion*» du réseau local : la requête ARP est transmise en «*broadcast*».
- ▷ Lorsqu'un **message ARP** est reçu, la couche liaison de donnée fait :
  - ◇ une **première vérification** pour voir si c'est une **requête ARP** et que **l'@IP demandée correspond à l'@IP locale** alors une **réponse ARP** est renvoyée à destination de l'@MAC de l'expéditeur.
    La machine répond parce qu'elle est concernée : c'est son adresse qui est demandée.
  - ◇ une **seconde vérification** pour vérifier si l'adresse IP de l'émetteur se trouve déjà dans la **mémoire cache ARP locale** sinon il y a **mise à jour** de la mémoire cache avec cette nouvelle association.
    Elle apprend l'association, comme dans le cas d'un «gratuitous ARP»*, c-à-d une réponse ARP non sollicitée envoyée en* ***broadcast***.

Illustration d'ARP

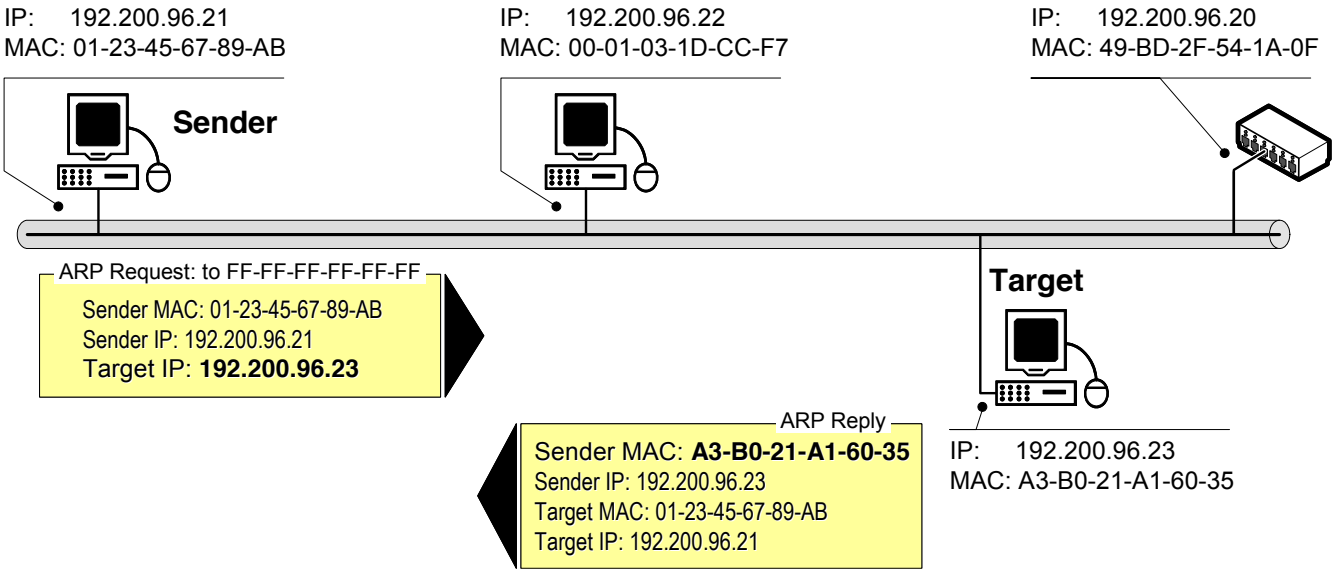
Comment échanger réellement sur un réseau local à diffusion ?

- Les machines ont chacune une carte réseau ;
- Chaque carte a une **adresse MAC unique** donnée par le constructeur ;
- Chaque machine dispose d'une **adresse IP** donnée par l'administrateur du réseau.



La machine `10.10.10.166` *demande l'@MAC de la machine* `10.10.12.52`.

Illustration d'ARP : les échanges



- La requête de la machine «`192.200.96.21`» demande l'@MAC de la machine `192.200.96.23`;
- La réponse de la machine `192.200.96.23` donne la réponse @MAC `A3:B0:21:A1:60:35`.

Pour envoyer un datagramme d'une source vers une destination, il faut savoir **localiser** la machine destination.

Deux possibilités :

- ▷ les deux machines **font partie** du même réseau local : on parle de **routage direct** (sur Ethernet, on utilisera le protocole ARP et l'envoi direct sur le réseau à diffusion) ;
- ▷ les deux machines **ne font pas partie** du même réseau local : on parle de **routage indirect**.
  On doit passer par un **intermédiaire** qui permet de sortir du réseau local pour aller vers l'extérieur : le **routeur** (ou appelé «passerelle» ou *gateway*).

Pour faire du routage direct ou indirect pour un datagramme

- ▷ connaître l'@IP d'un **routeur de sortie** ;
- ▷ savoir si les deux machines font **partie du même réseau** local :
  - ◇ si elles **sont** dans le même réseau local : remettre **directement** le datagramme à la machine destination ;
  - ◇ si elles **ne sont pas** dans le même réseau local : remettre le datagramme **au routeur** pour l'envoyer **indirectement** à la machine destination.

Comment savoir si Source et Destination sont dans le même réseau local ?

Il faut **comparer** l' `<id. réseau>` des deux adresses : si c'est **la même** $\Longrightarrow$ les deux sont dans le **même réseau local**.

Comment remettre le datagramme au routeur

Il faut utiliser le mécanisme **d'encapsulation** d'un datagramme dans une trame :

- ▷ la **trame** sert à remettre des données d'une machine connectée à un réseau local à une autre machine connectée au même réseau local ;
- ▷ la **trame** possède une @MAC de destination **indépendante** de l'@IP : il est possible d'envoyer la trame à une machine dont l'@IP **ne correspond pas** à son @IP !

Par exemple : on peut envoyer une datagramme à destination de l'extérieur du réseau local à l'@MAC du routeur.

Attention : *les attaques MiTM, «Man-in-the-Middle», opèrent sur l'association @MAC* $\Leftrightarrow$ @IP du routeur !

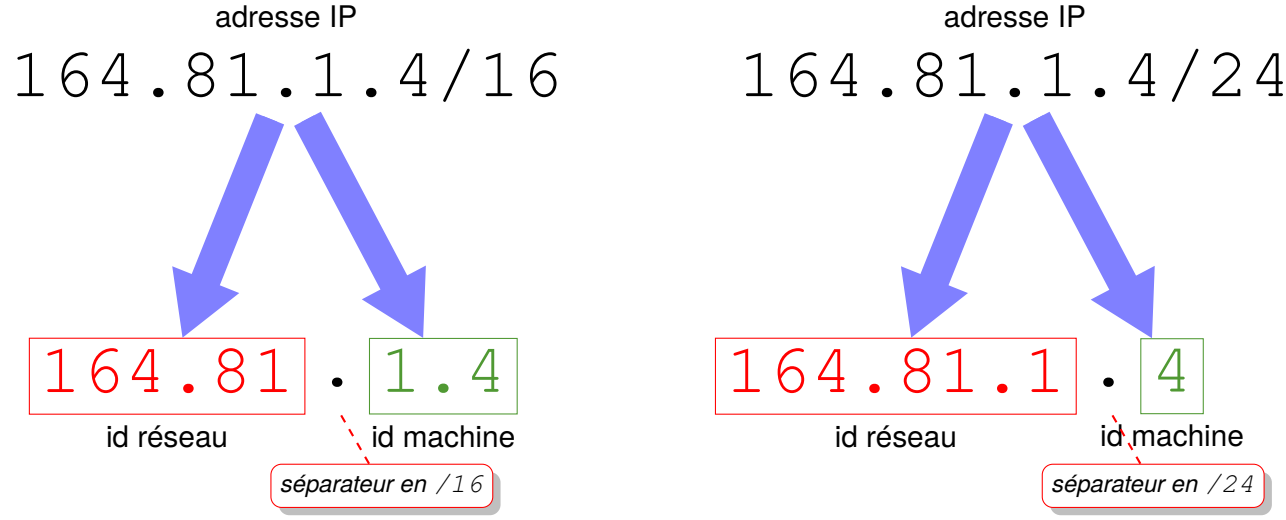
Comment connaître son réseau ?

Décomposition de l'adresse IP suivant la répartition `<id réseau><id machine>`

Configuration d'une machine pour l'accès au réseau :

- adresse IP sur 32bits en notation «*décimale pointée*» ;
- taille de l'identifiant réseau en bit⟹ indique l'emplacement du séparateur ;

Exemples :



Chaque valeur «décimale» *correspond à un octet ou 8bits :*

- ▷ */16 correspond à 16bits ou deux octets ou deux valeurs décimales ;*
- ▷ */24 correspond à 24bits ou trois octets ou trois valeurs décimales ;*

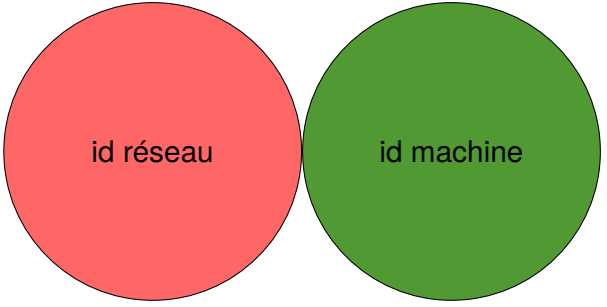
Comment connaître son réseau ?

Pourquoi faire varier la taille de l'identifiant réseau ?

Si la taille de l'identifiant réseau augmente ↗ alors :

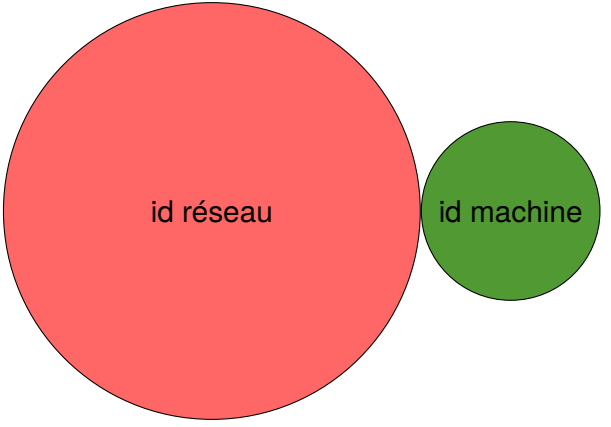
- ▷ le nombre de réseau possible augmente ↗ ;
- ▷ le nombre de machines pour chacun des réseaux diminue ↘.

164.81.1.4/16



2^{16} *réseaux, chacun contenant* 2^{16} *machines.*

164.81.1.4/24



2^{24} *réseaux, chacun contenant* 2^{8} *machines.*

Si la taille de l'identifiant réseau diminue ↘ alors :

- ▷ le nombre de réseau possible diminue ↘ ;
- ▷ le nombre de machines pour chacun des réseaux augmente ↗.

Communiquer vers l'extérieur de son réseau

A veut communiquer avec B

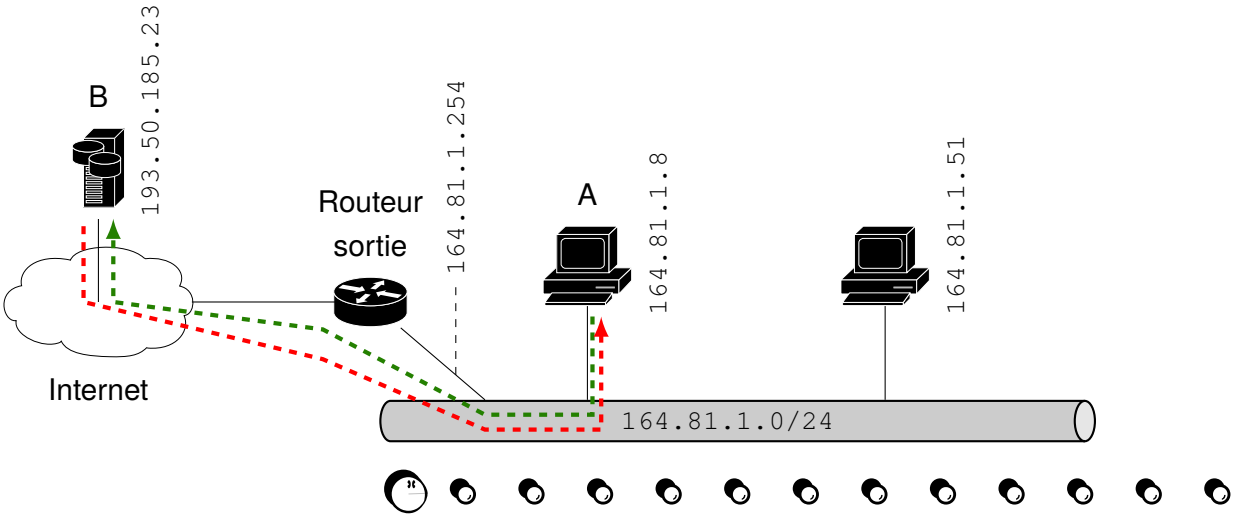
- A⟹164.81.1.8/24;
- B⟹193.50.185.23;

164.81.1.8 → 164.81.1.**8**

193.50.185.23 → 193.50.185.**23**

164.81.1 ≠ 193.50.185

Les identifiants réseaux sont différents⟹Routage indirect⟹passage par le Routeur.

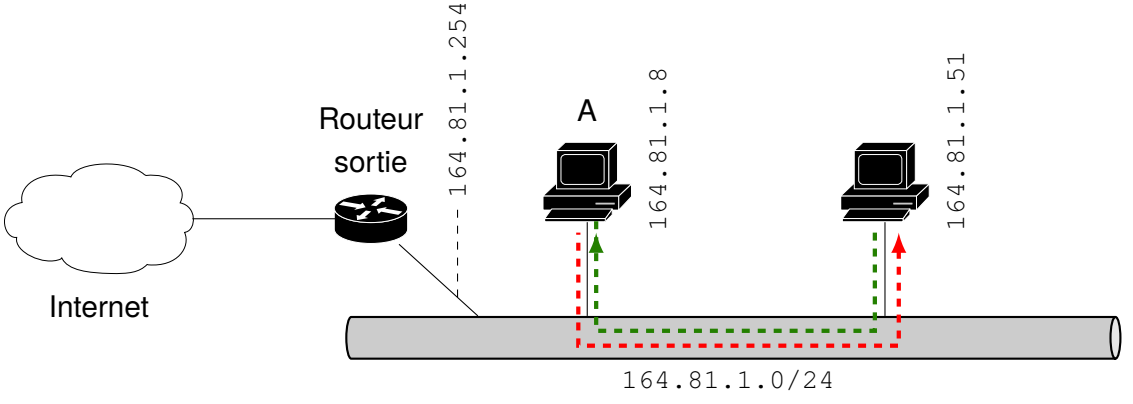


Communiquer vers l'intérieur de son réseau

A veut communiquer avec B

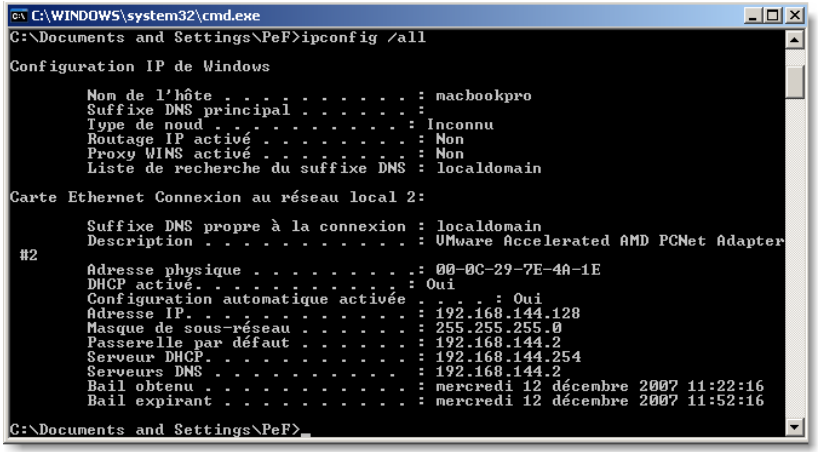
- A⟹164.81.1.8/24;
- B⟹164.81.1.51;

Les identifiants réseaux sont identiques⟹Routage direct.

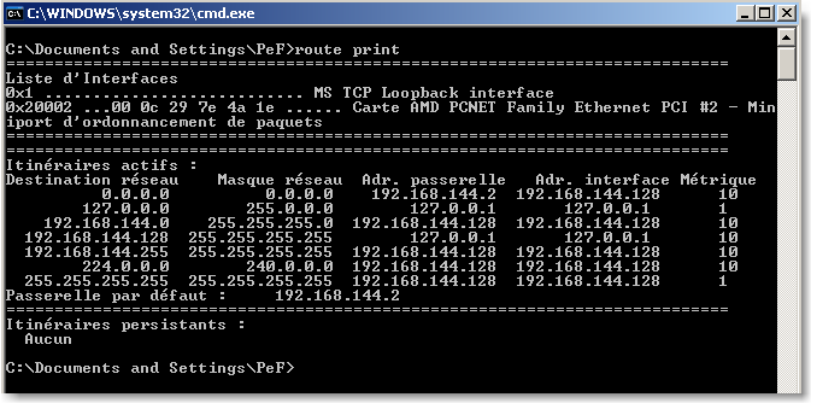


Sous Windows

Commande `ipconfig`

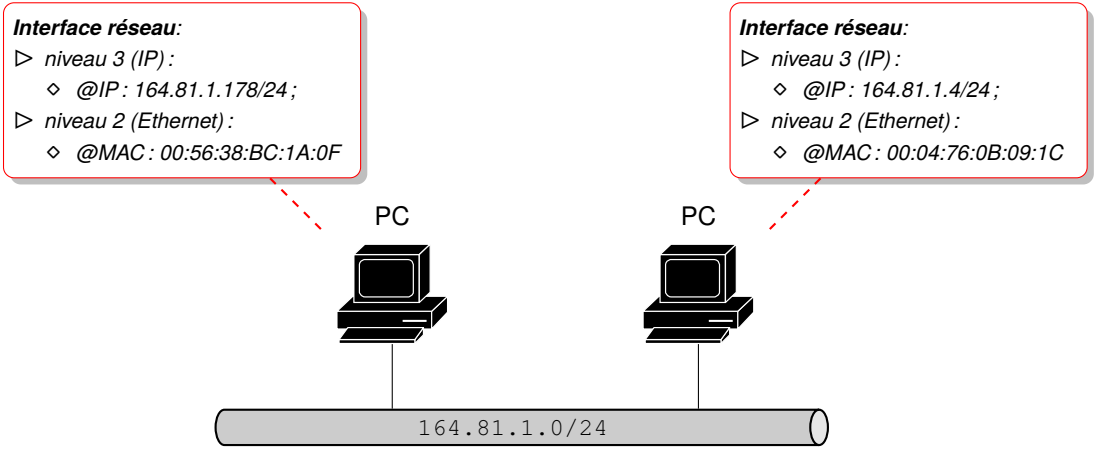


Commande `route print`



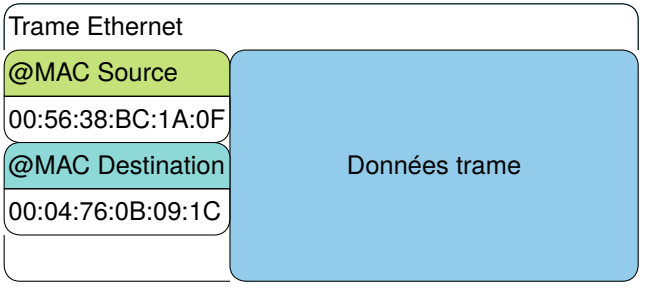
Chaque machine est identifiée par :

- une adresse de niveau 2 (@MAC) ;
- une adresse de niveau 3 (@IP) ;
- un réseau d'appartenance connu :
  - à l'aide du **préfixe** «`/n` indiquant n le nombre de bits de l'identifiant réseau»
  - ou à l'aide du **masque réseau**, *netmask*, adresse où chaque bit de l'identifiant réseau est à 1, les autres sont à 0.

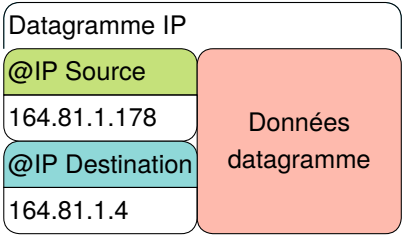


Pour être échangé, les datagrammes IP sont encapsulés dans des trames Ethernet

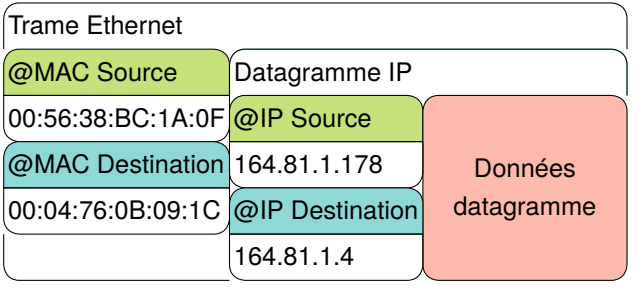
- la **trame** contient :
  - une @MAC source ;
  - une @MAC de destination ;



- le **datagramme** contient :
  - une @IP source ;
  - une @IP destination et des **données** ;



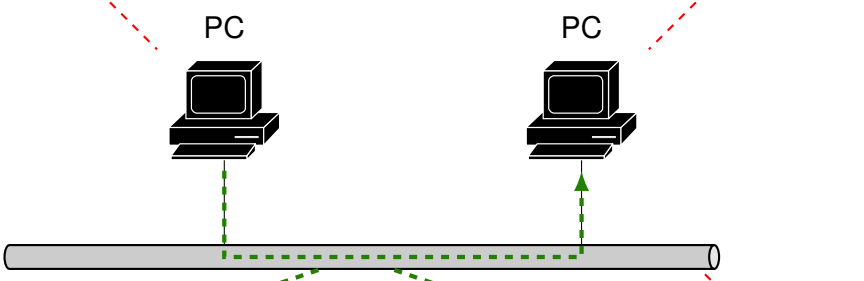
- la **trame encapsule** le datagramme IP :



Routage direct illustré

***Interface réseau*:**
- *niveau 3 (IP) :*
  - *@IP : 164.81.1.178/24 ;*
- *niveau 2 (Ethernet) :*
  - *@MAC : 00:56:38:BC:1A:0F*

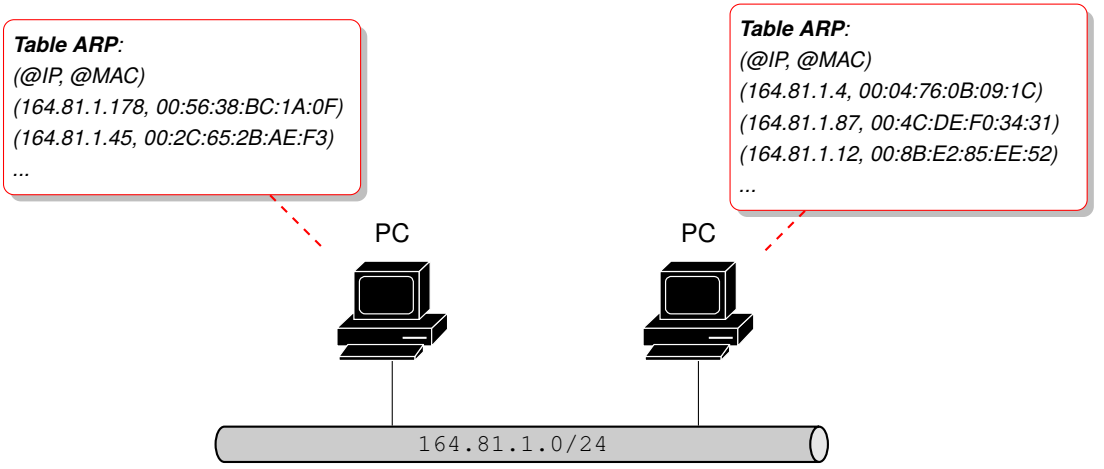
***Interface réseau*:**
- *niveau 3 (IP) :*
  - *@IP : 164.81.1.4/24 ;*
- *niveau 2 (Ethernet) :*
  - *@MAC : 00:04:76:0B:09:1C*



Trame Ethernet		
@MAC Source	Datagramme IP	
00:56:38:BC:1A:0F	@IP Source	Données datagramme
@MAC Destination	164.81.1.178	
00:04:76:0B:09:1C	@IP Destination	
	164.81.1.4	

Pour connaitre la correspondance entre adresse IP et adresse MAC :

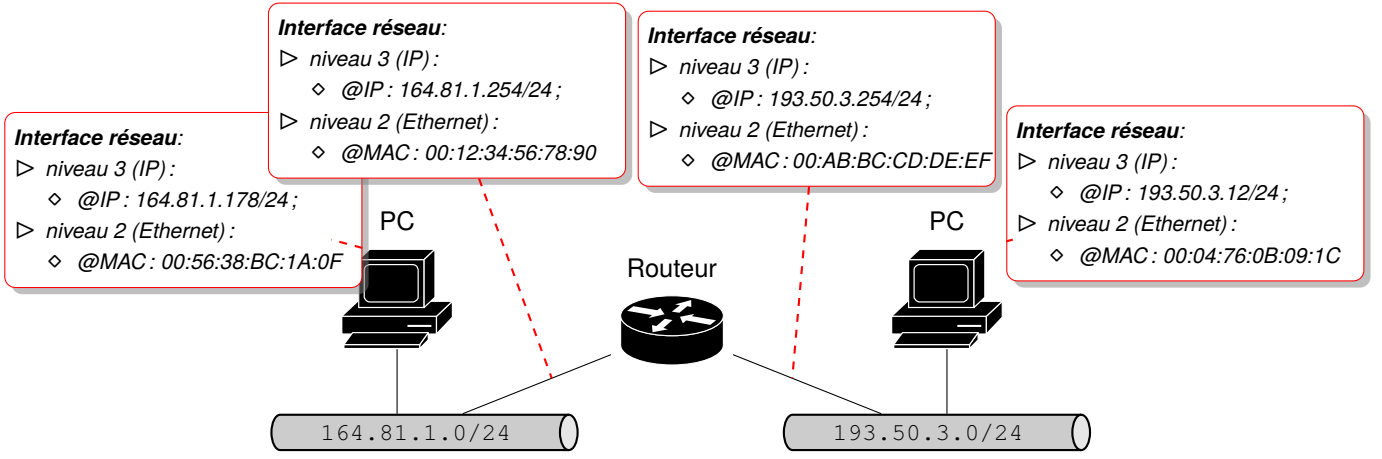
- ▷ mise en oeuvre du protocole ARP (Address Resolution Protocol) ;
- ▷ construction d'une table de correspondance entre @ IP et MAC sur chaque machine (cache ARP).



La **modification malveillante** *de cette table est possible : «ARP Spoofing» (usurpation d'identité), «ARP Cache Poisoning» (insertion d'association erronée).*

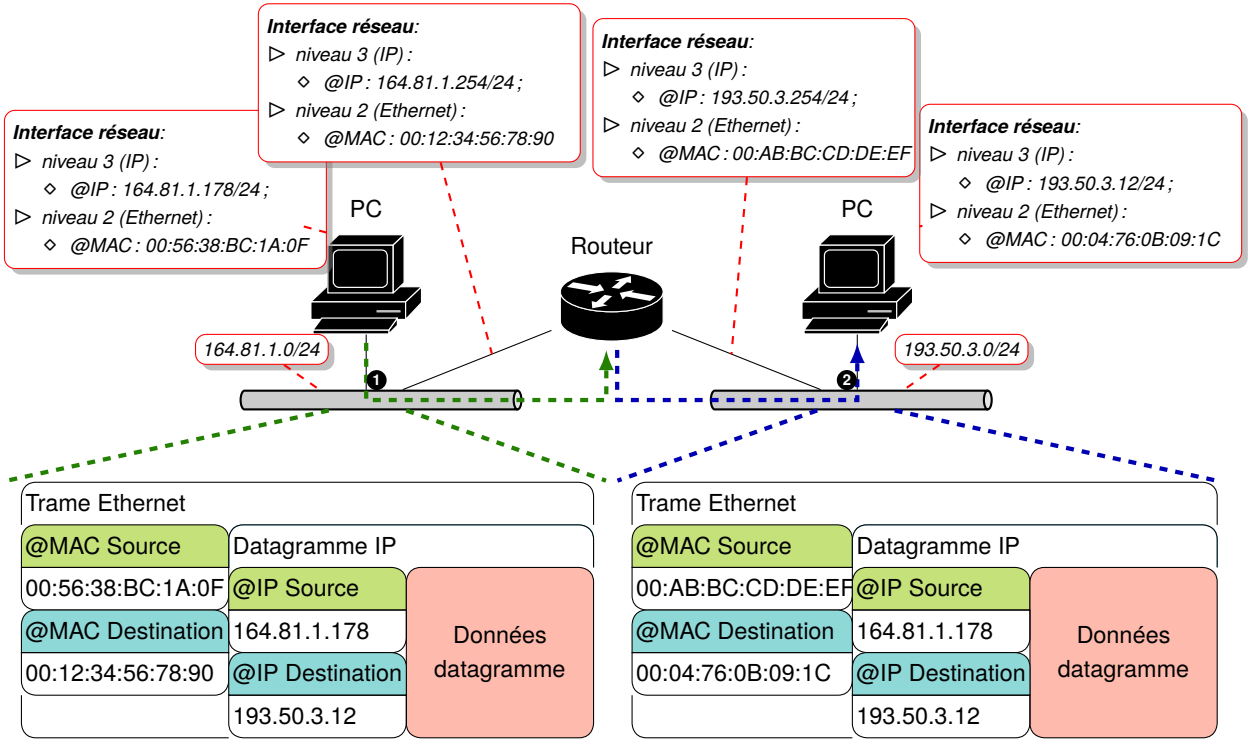
Routage indirect illustré

Le paquet de la machine `164.81.1.178` est routé par l'intermédiaire du routeur vers la machine `193.50.3.12`.



Le datagramme IP est **encapsulé** :

- par la machine `164.81.1.178`, dans une trame à **destination du routeur** ;
- puis, par le routeur, dans une nouvelle trame à destination de la machine `193.50.3.12`.

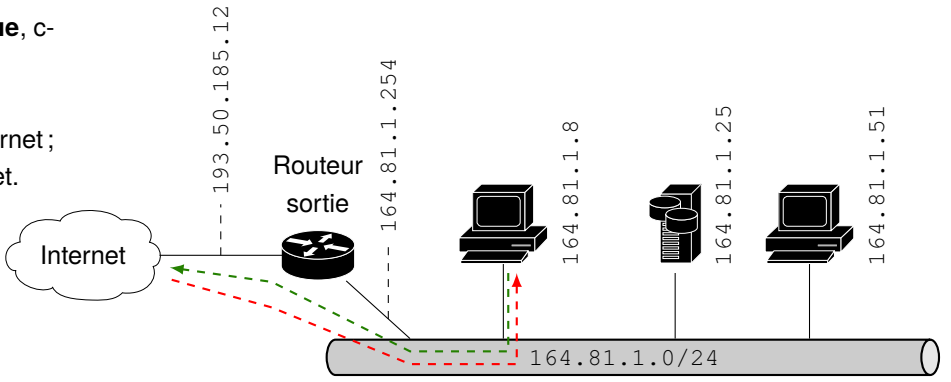


L'encapsulation permet la redirection vers le routeur sans modifier les @IP du datagramme.

Utilisation d'adresses réseaux publiques

Le réseau utilise une **adresse publique**, c-à-d «*routable*» sur Internet.
Une machine :

- ▷ communique directement **vers** Internet ;
- ▷ **peut** être contactée **depuis** Internet.



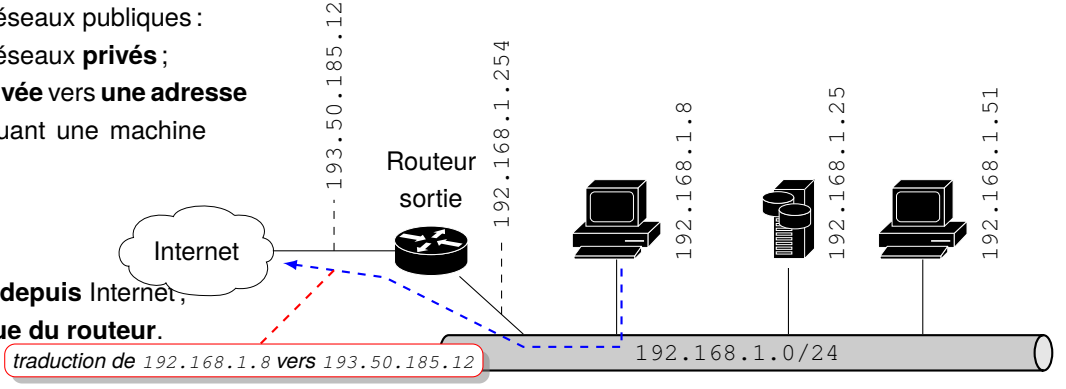
Utilisation d'adresses réseaux privées

On **économise** des adresses réseaux publiques :

- ▷ on utilise des adresses de réseaux **privés** ;
- ▷ on «traduit» une **adresse privée** vers **une adresse publique** et inversement quant une machine veut accéder à Internet.

Une machine :

- ▷ communique **vers** Internet ;
- ▷ **ne peut pas** être contactée **depuis** Internet ;
- ▷ emprunte **l'adresse publique du routeur**.

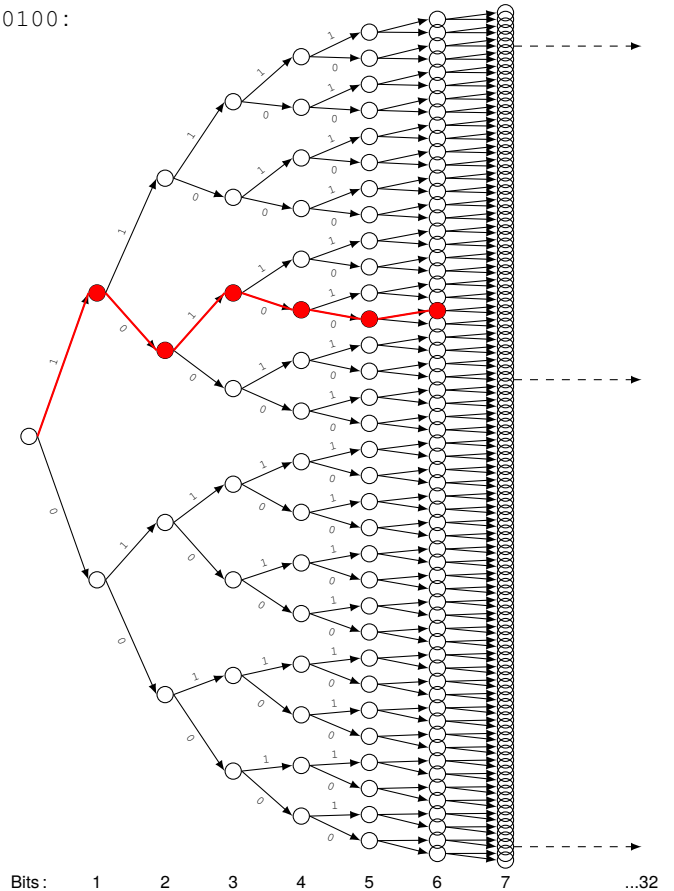


Le NAT :

- ▷ **Économise** les adresses publiques car **il n'existe plus** d'adresses IPv4 libres ;
- ▷ **Protège** les machines du réseau : elles **ne sont pas accessibles** directement depuis Internet.

Soit l'adresse IP `164.x.y.z`, `164`$\Longrightarrow$`10100100`:

Chaque bit de l'adresse définit un chemin différent et permet d'atteindre la destination...



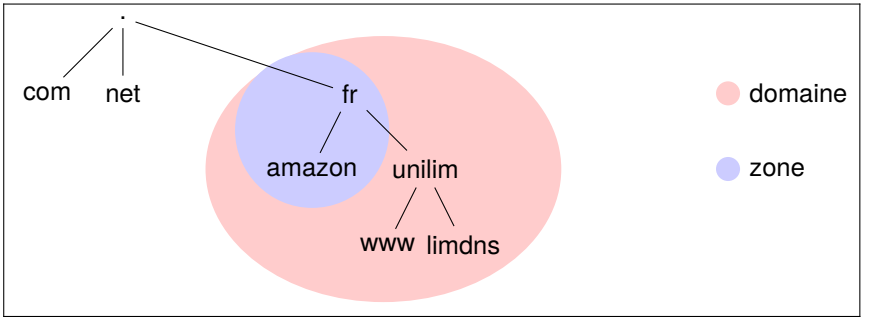
Le système DNS est entièrement distribué au niveau planétaire en utilisant la **délégation de domaine**.

À tout domaine est associé une **responsabilité administrative**.

Une organisation responsable d'un domaine peut :

- ▷ **découper** le domaine en sous-domaines ;
- ▷ **déléguer** les sous-domaines à d'autres organisations :
  - ◇ qui deviennent responsables du (des) sous-domaine(s) qui leurs sont délégué(s) peuvent, à leur tour, déléguer des sous-domaines des sous-domaines qu'elles gèrent.
  - ◇ *Le domaine parent contient alors seulement un pointeur vers le sous-domaine délégué;*

- ⋆ Les serveurs de nom enregistrent les données propres à une partie de l'espace nom de domaine dans une **zone**.
- ⋆ le serveur de nom à **autorité administrative** sur cette zone ;
- ⋆ un serveur de nom peut avoir **autorité** sur plusieurs zones ;
- ⋆ une **zone** contient les informations d'un domaine **sauf** celles qui sont **déléguées** :

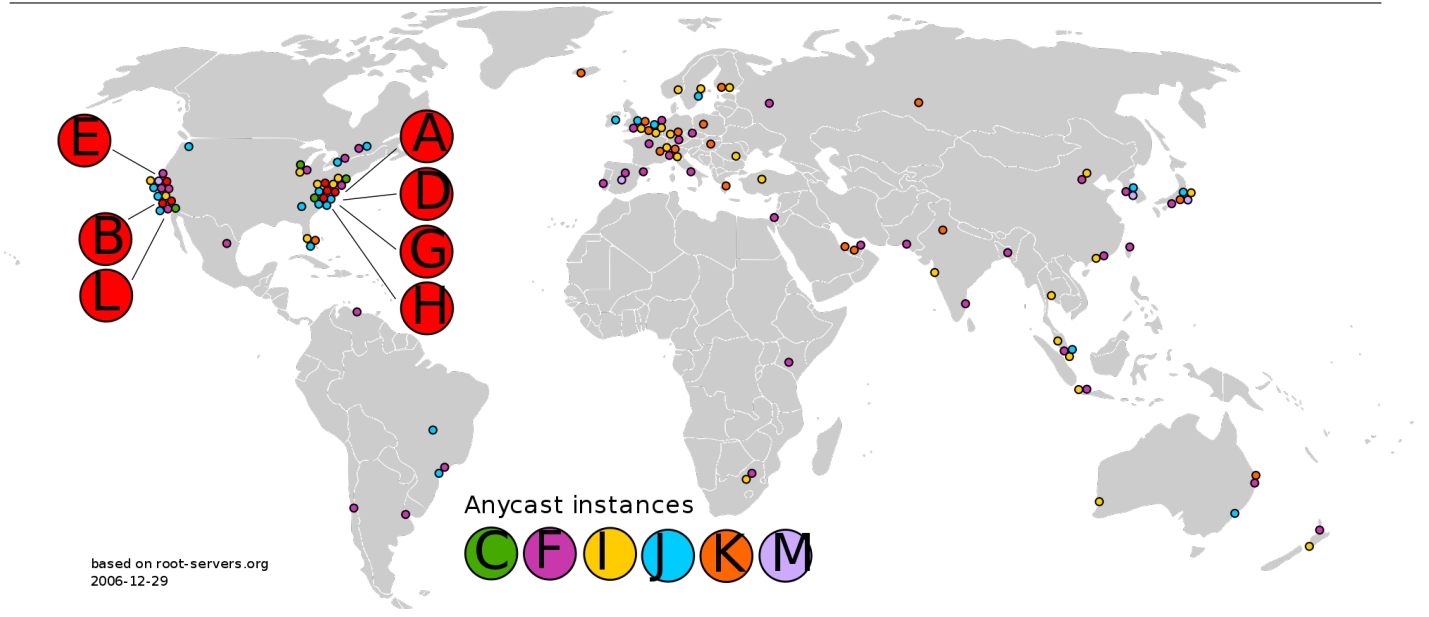


Whols «unilim.fr»

```
darkstar:~ pef$ whois unilim.fr
%%
%% This is the AFNIC Whois server.
%%
%% complete date format : DD/MM/YYYY
%% short date format    : DD/MM
%% version              : FRNIC-2.5
%%
%% Rights restricted by copyright.
%% See http://www.afnic.fr/afnic/web/mentions-legales-whois_en
%%
%% Use '-h' option to obtain more information about this service.
%%
%% [2a01:0e35:8a71:bec0:66b9:e8ff:fed2:23ba REQUEST] >> unilim.fr
%%
%% RL Net [##########] - RL IP [#########.]
%%
domain:      unilim.fr
status:      ACTIVE
hold:        NO
holder-c:    UDL3-FRNIC
admin-c:     JPL1325-FRNIC
tech-c:      GRST1-FRNIC
tech-c:      NV70-FRNIC
tech-c:      GU245-FRNIC
zone-c:      NFC1-FRNIC
nsl-id:      NSL5796-FRNIC
registrar:   GIP RENATER
Expiry Date: 01/01/2016
created:     01/01/1995
last-update: 15/12/2014
source:      FRNIC

ns-list:     NSL5796-FRNIC
nserver:     limdns.unilim.fr [164.81.1.4]
nserver:     limdns2.unilim.fr [164.81.1.5]
nserver:     cnudns.cines.fr [193.48.169.40 2001:660:6301:301::2:1]
source:      FRNIC

registrar:   GIP RENATER
type:        Isp Option 1
address:     23-25 Rue Daviel
address:     PARIS
country:     FR
phone:       +33 1 53 94 20 30
fax-no:      +33 1 53 94 20 31
e-mail:      domaine@renater.fr
website:     http://www.renater.fr
anonymous:   NO
registered:  01/01/1998
source:      FRNIC

nic-hdl:     JPL1325-FRNIC
type:        PERSON
contact:     Jean-Pierre Laine
address:     Unvi. de Limoges
address:     123, Aveneu Albert Thomas
address:     87060 Limoges
country:     FR
phone:       +33 5 55 45 77 08
e-mail:      jean-pierre.laine@unilim.fr
registrar:   GIP RENATER
changed:     24/10/2013 nic@nic.fr
anonymous:   NO
obsoleted:   NO
source:      FRNIC
```

Le responsable du domaine : Jean-Pierre Laine

Les 13 serveurs racines



Le terme «anycast» permet d'offrir des services DNS de proximité à l'aide de cette capacité offerte par IPv6.

On peut remarquer que ce service de proximité permet même de délocaliser géographiquement les serveurs et améliore la disponibilité et la sécurité du système DNS.

Le serveur de courrier

MX = Mail eXchanger Permet l'adressage email sur la base du nom de domaine plutôt que sur l'adresse du (des) serveur(s) de mail :

- `bonnefoi@unilim.fr` plutot que `bonnefoi@msi.unilim.fr` ;
- permet à l'émetteur d'ignorer quelle est la machine serveur de mail ;
- permet le deplacement du gestionnaire de mail vers une autre machine ;
- permet la gestion de plusieurs serveurs de mail avec priorité dans l'ordre de consultation des serveurs

L'enregistrement MX est utilisés par les MTA, «*Mail Transfer Agent*», en tenant compte des priorités :

Exemple pour l'Université de Limoges :

xterm

```
bonnefoi@msi:~$ dig +short mx unilim.fr
50 mail.unilim.fr.
```

le serveur d'envoi de courrier de l'Université

Exemple pour Google :

xterm

```
bonnefoi@msi:~$ dig +short mx google.com
30 alt2.aspmx.l.google.com.
10 aspmx.l.google.com.
20 alt1.aspmx.l.google.com.
50 alt4.aspmx.l.google.com.
40 alt3.aspmx.l.google.com.
```

Le MTA utilise le protocole **SMTP**, «*Simple Mail Transfer Protocol*», en tant que client.

Une **interface de programmation** définie pour mettre en place **simplement** des communications :

- chaque communication a lieu avec :
  - **un interlocuteur** : communication «point à point», ou «unicast» ;
  - **plusieurs interlocuteurs** : communication par «diffusion» ou «multicast» ;
- la communication correspond à l'échange de données entre les interlocuteurs :
  - des **données en continu** : flux d'octets de taille indéfinie, non connue à l'avance ;
  - des **paquets** : données de taille fixe et réduite connue à l'avance.

Deux types de communication uniquement en TCP/IP

1. mode «**connecté**»
   - elle ne concerne que **deux interlocuteurs** : un de chaque côté (point à point) ;
   - les données arrivent les unes après les autres dans «l'ordre d'émission» ;
   - la communication est **bi-directionnelle** (dans les deux sens) ;
   - elle est «full-duplex», les deux interlocuteurs peuvent échanger **simultanément** ;
   - il y a une **garantie contre la perte de données**.
     *C'est le mode offert par le protocole **TCP**, «Transmission Control Protocol».*
2. mode «**datagramme**»
   - elle peut concerner un ou plusieurs interlocuteurs (unicast ou multicast) ;
   - les données sont groupées dans des **paquets de taille limitée** ;
   - il peut y avoir des **pertes de paquets**.
     C'est le mode offert par le protocole UDP, «User Datagram Protocol».

Attention

Le mode «connecté» est simulé par TCP sur un réseau en mode «datagramme».

Modèle Client/Serveur

Un logiciel «serveur» **attend** la communication en provenance d'un logiciel «client».

Localisation du logiciel serveur

- un ordinateur est localisable sur Internet grâce à son adresse IP ;
- un ordinateur ne possède habituellement qu'une adresse IP joignable ;
- un ordinateur peut exécuter plusieurs programmes qui peuvent vouloir communiquer simultanément ;
- il faut **multiplexer ces communications** en «sachant» avec quel programme communiquer : notion de «port» !

À chaque processus communiquant est associé un port

Pour une communication en «mode connecté» :

- un Serveur qui **attend** la connexion du client ;
- un Client qui **effectue** la connexion au serveur.

Pour localiser le Serveur ? Connaître le numéro de port où attend la communication !

Comment connaître le numéro de port ?

Le point sur les communications sur un ordinateur :

- chaque communication est associée à **un seul programme donné** (logiciel de messagerie, navigateur web, client de chat, *etc*) ;
- chaque communication se fait suivant un **protocole donné** (SMTP, POP pour récupérer le courrier, HTTP, *etc*) ;
- chaque protocole est associé à un «serveur» particulier : serveur SMTP pour l'envoi de courrier, serveur Web, serveur FTP, *etc*.
- un **numéro de port identifie un serveur donné** : il faut rendre **standard** les numéros de port !
  Exemple : http : 80, ftp : 21, smtp : 25, DNS : 53 *etc*, la liste dans le fichier `/etc/services`.

Le client veut communiquer avec un serveur donné ? il utilise le port standard associé !

La notion de numéro de port : «multiplexer» les communications

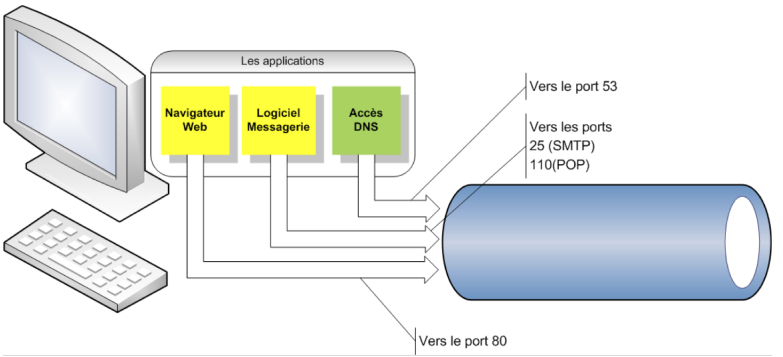
Notion de numéro de port

- différentes communications peuvent avoir lieu pour des protocoles différents, donc des programmes différents, donc des numéros de port différents ;
- chaque communication sur une machine est identifiée par un **TSAP**, «*Transport Service Access Point*», c-à-d un couple (@IP, numéro de port).

Comment un ordinateur peut-il voir plusieurs communications simultanément ?

On ajoute également la notion de **numéro de port** :

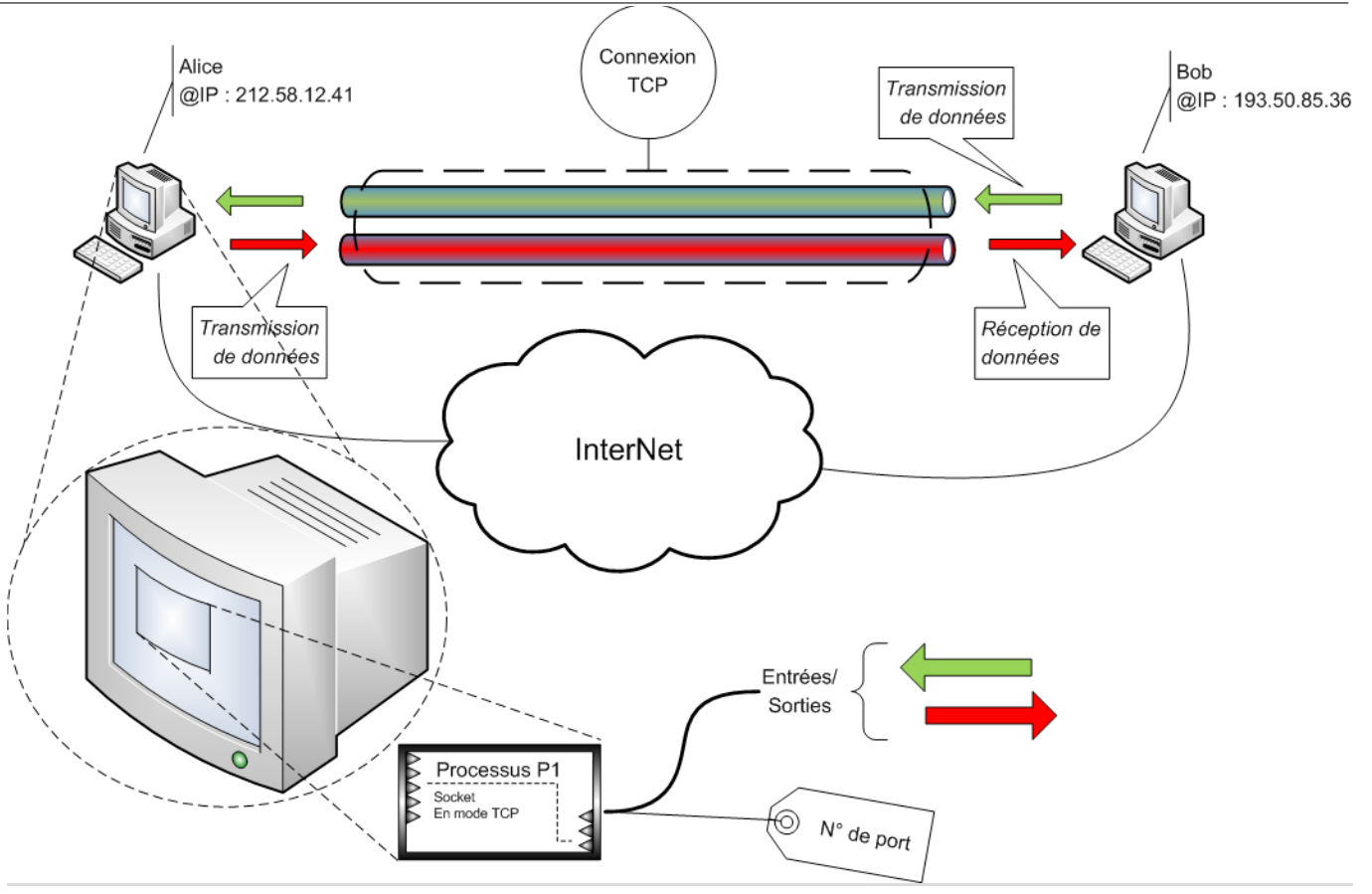
- il varie de 1 à 65535 (sur 16 bits) ;
- il est associé à un seul programme ;
- du côté de la machine *cliente*, il peut prendre n'importe quelle valeur ;
- du côté de la machine *serveur*, il permet à la machine cliente de désigner le programme que l'on veut contacter ;



Le port permet de **multiplexer** les communications :

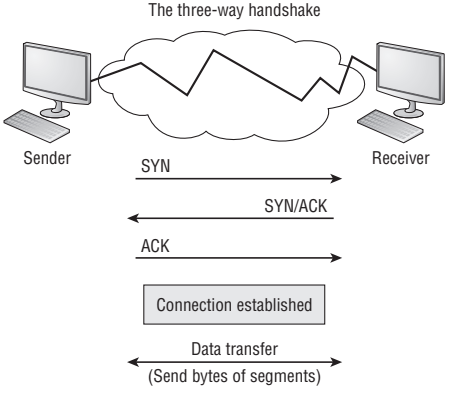
- chaque datagramme sera identifié par le TSAP_{source} duquel il transporte les données ;
- tous les datagrammes utilisent le même lien de communication ;
- lors de leur arrivée sur la machine destination, ils sont identifiés par leur $\text{TSAP}_{destination}$ et remis au bon processus.

Le protocole TCP : connexion et échanges



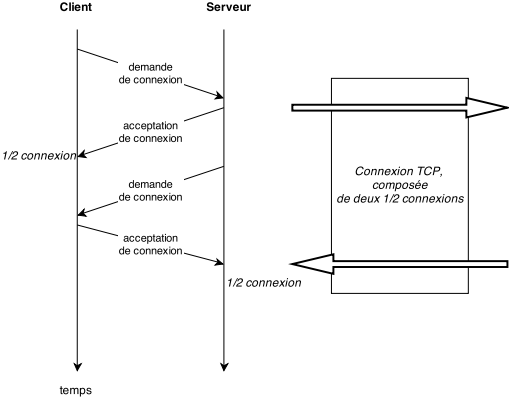
Une communication «orientée connexion» correspond à :

- une **demande d'accord** de la part de l'interlocuteur **avant de lui envoyer des données**, c-à-d une $1/2$ connexion ;
- un envoi de données **sans perte** et **sans erreur** ;
- une communication **bi-directionnelle** (d'un interlocuteur vers l'autre et vice-versa) et «*full duplex*» (chaque interlocuteur peut communiquer simultanément avec l'autre) ;



«*The three-way handshake*», correspond à l'établissement d'une communication depuis le client vers le serveur :

- une demi-connexion du client vers le serveur ;
- une demi-connexion du serveur vers le client ;



Celui qui **initie** la communication est le client.
Celui qui **attend** la communication est le serveur.

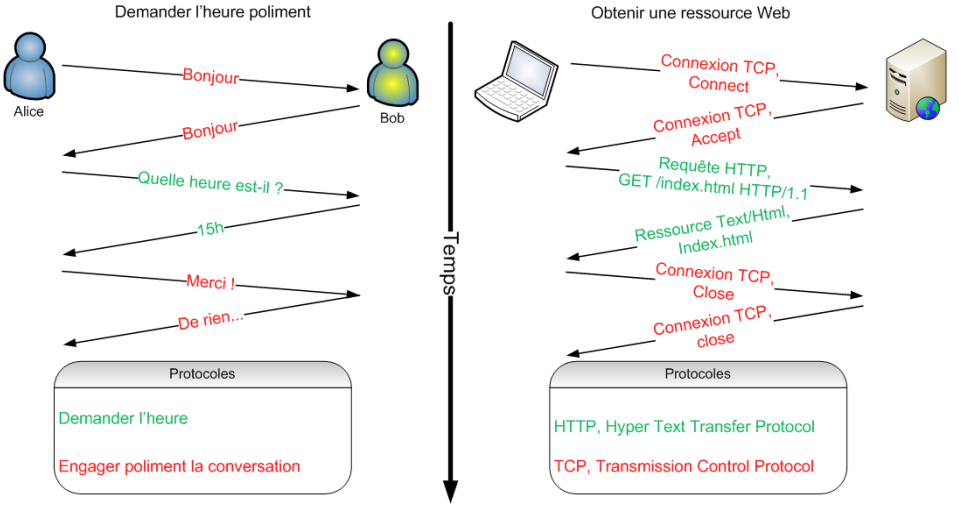
Firewall & Filtrage

Communication autorisée (non filtrée) :
Si le client est dans le LAN et le serveur sur Internet $\Longrightarrow$ Communication Intérieur$\longrightarrow$Extérieur
L'appartenance du Client au LAN est déduite grâce à la présence du «`SYN`».

À noter : la demande de 1/2 connexion du serveur (`SYN`), ainsi que son acceptation de la demande de 1/2 connexion du client (`ACK`), sont transmises dans le même message, d'où la simplification en 3 échanges seulement.

Un protocole humain et un protocole machine

« demander l'heure à quelqu'un » et « demander une ressource sur un serveur Web ».



Les protocoles définissent :

- le **format** des données échangées ;
- l'**ordre** des messages émis et reçus entre les entités réseaux;
- ainsi que les **réactions** à ces messages.

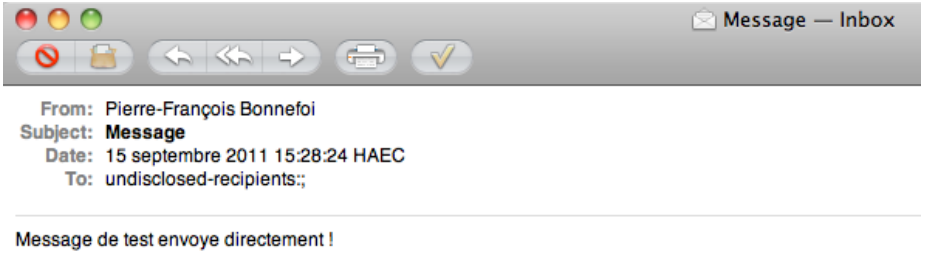
Un protocole correspond à un **comportement** qui **évolue** en fonction des données échangées.

Le protocole SMTP, «Simple Mail Transfer Protocol»

xterm

```
bonnefoi@msi:~$ socat - tcp:smtp.unilim.fr:25
220 smtp.unilim.fr ESMTP Sendmail 8.13.1/8.13.1; Thu, 15 Sep 2011 15:28:24 +0200
HELO msi.unilim.fr
250 smtp.unilim.fr Hello www.msi.unilim.fr [164.81.60.6], pleased to meet you
MAIL FROM: <bonnefoi@unilim.fr>
250 2.1.0 <bonnefoi@unilim.fr>... Sender ok
RCPT TO: <bonnefoi@unilim.fr>
250 2.1.5 <bonnefoi@unilim.fr>... Recipient ok
DATA
354 Enter mail, end with "." on a line by itself
Subject: Message

Message de test envoye directement !
.
250 2.0.0 p8FDSOJe031646 Message accepted for delivery
QUIT
221 2.0.0 smtp.unilim.fr closing connection
bonnefoi@msi:~$
```

La commande «`socat`» permet de simplement établir une connexion TCP avec le serveur que l'on a désigné sur le port indiqué.



Le protocole HTTP, «*Hyper Text Transfer Protocol*»

xterm

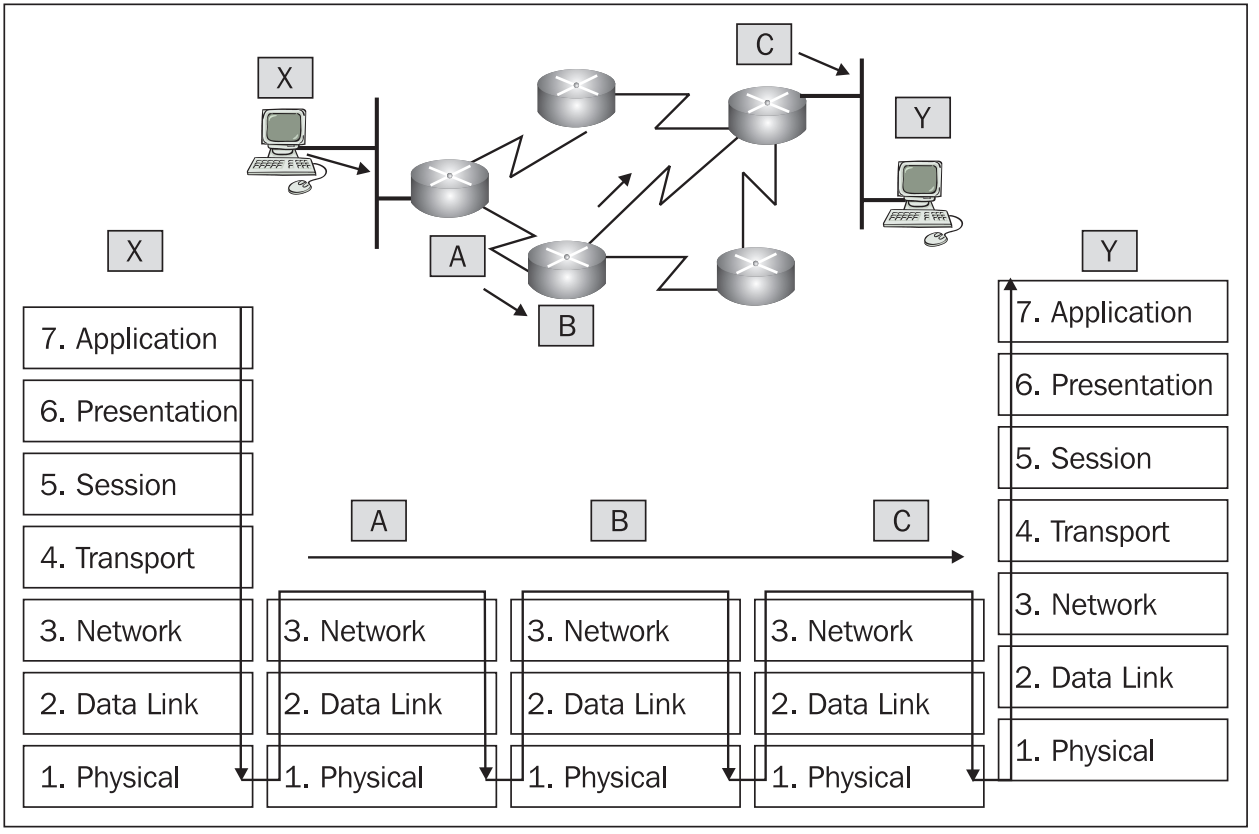
```
pef@darkstar-8:/Users/pef  socat - tcp:www.unilim.fr:80
HEAD / HTTP/1.0

HTTP/1.1 301 Moved Permanently
Server: nginx
Date: Mon, 11 Sep 2017 10:53:33 GMT
Content-Type: text/html
Content-Length: 178
Connection: close
Location: http://www.cryptis.fr/
```

Le protocole POP, «*Post Office Protocol*»

xterm

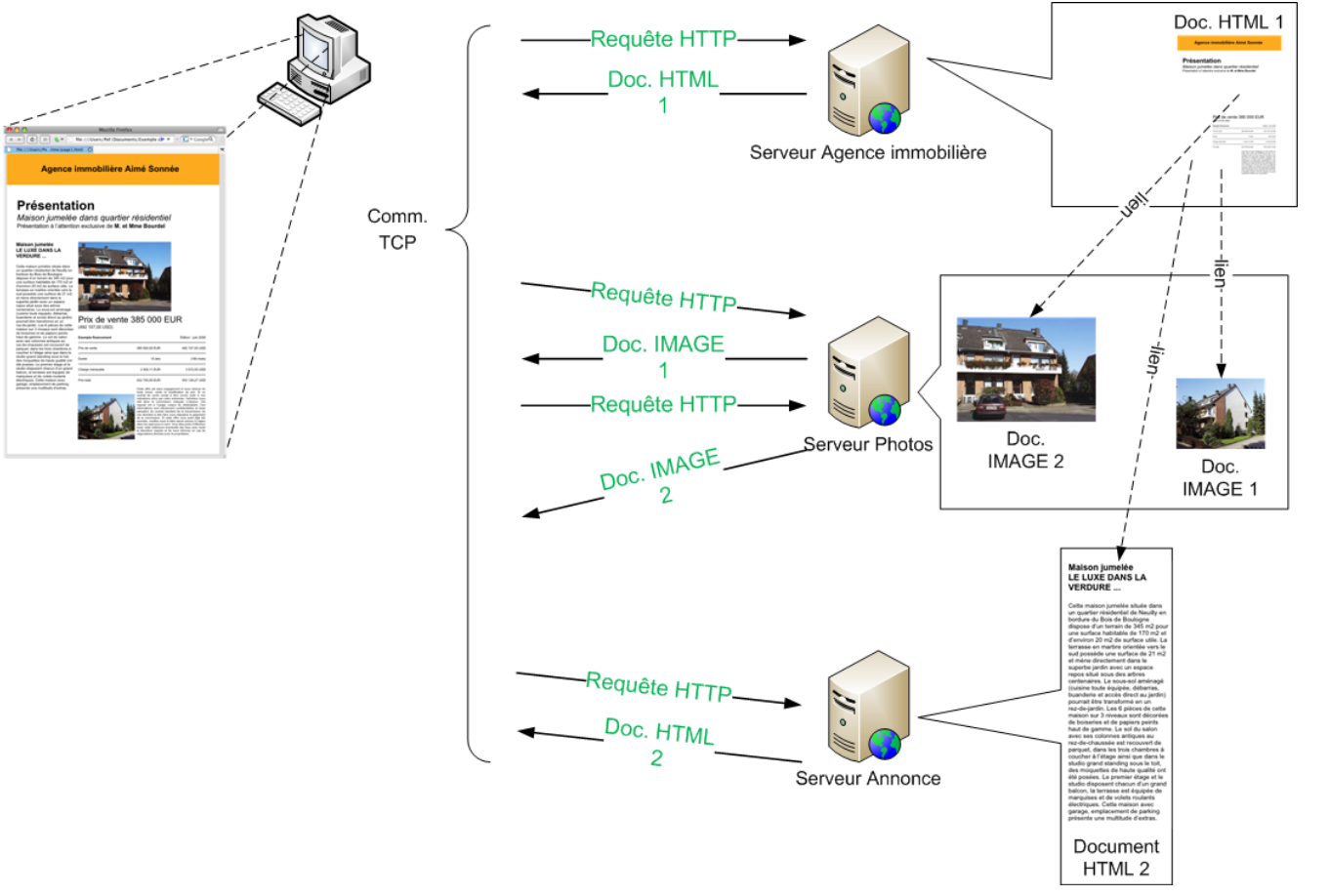
```
bonnefoi@msi:~$ socat - tcp:pop.unilim.fr:110
+OK courriel Cyrus POP3 v2.2.13-Debian-2.2.13-14.xm.1 server ready <299345444.1316380363@courriel>
USER bonnefoi
+OK Name is a valid mailbox
PASS bob
-ERR [AUTH] Invalid login
^C
bonnefoi@msi:~$
```

Ce protocole est utilisé pour consulter son courrier et le récupérer dans son logiciel de messagerie.
On lui préfère le protocole IMAP, port 143 en version non sécurisée, qui permet de consulter son courrier tout en le laissant sur le serveur.

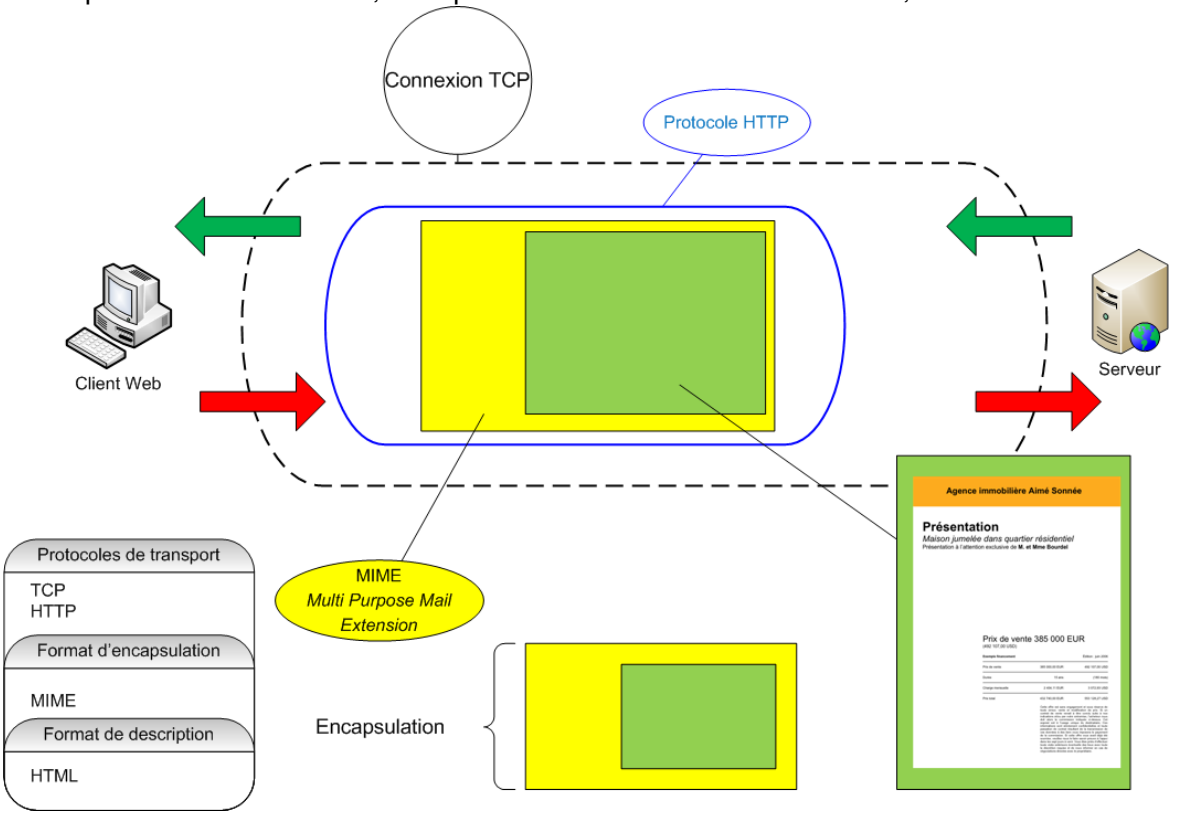


Chaque matériel dispose d'une pile TCP/IP, plus ou moins complète.

Le protocole HTTP, «HyperText Transfer Protocol», RFC 1945

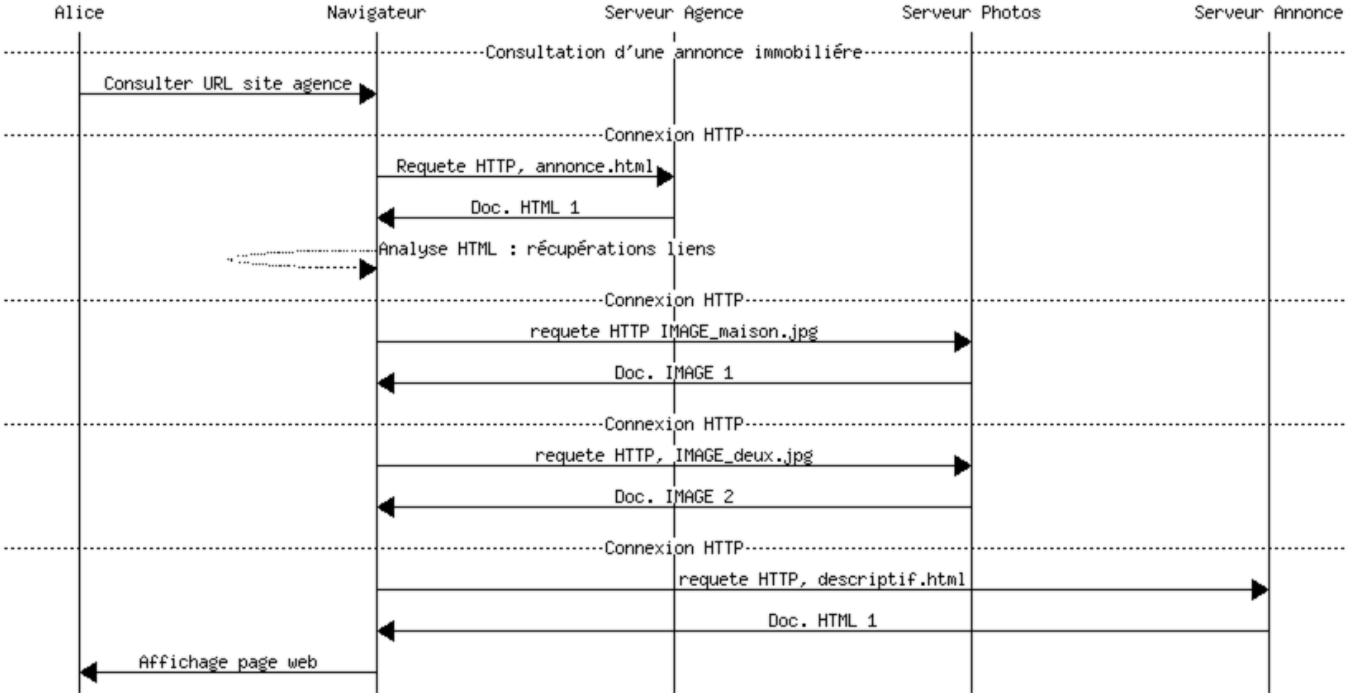


Utilisation du «mode connecté» : protocole de transport TCP, encapsulant le protocole HTTP pour échanger un descripteur de format MIME, encapsulatn un contenu formaté HTML, …



Une pile de protocole

- protocole « utilisateur » ou abstrait : consultation d'une page web ;
- protocole de transport : TCP ;
- protocole d'échange : HTTP ;
- format d'échange : MIME.



Rapport entre les différents protocoles

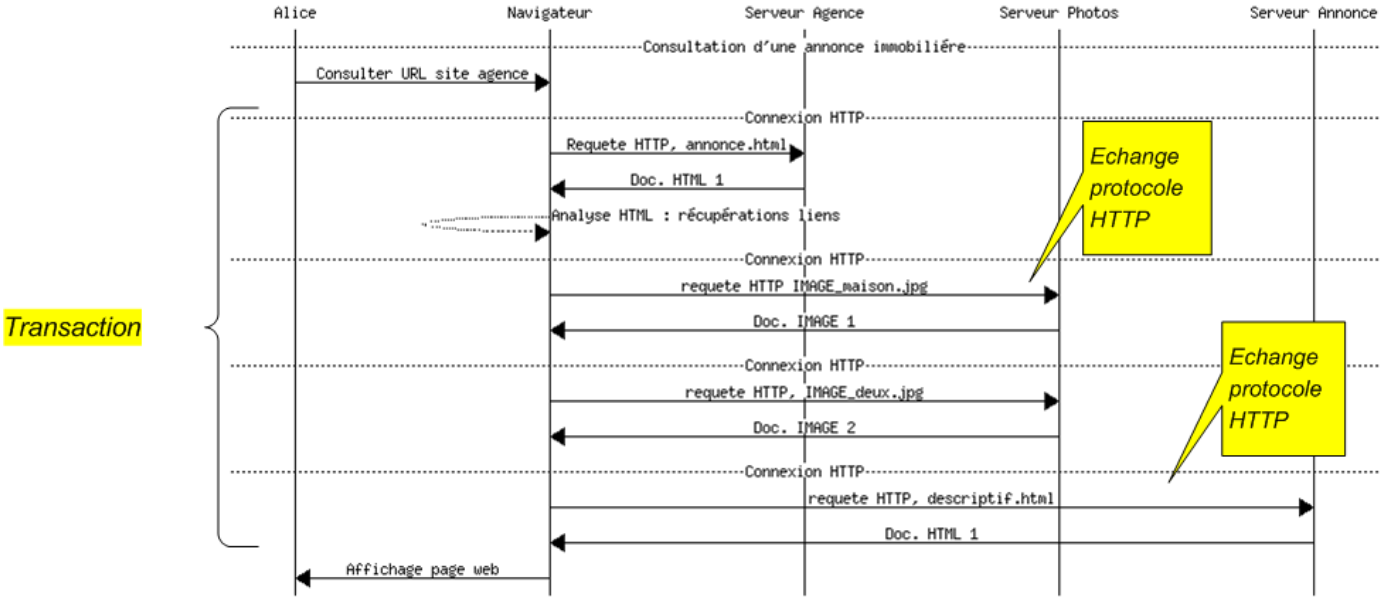
Le protocole abstrait est celui qui intéresse l'utilisateur (ici, Alice), c-à-d. « naviguer sur le Web ».

L'unité élémentaire de ce protocole est la transaction (Alice charge une page Web).

Le navigateur d'Alice réalise plusieurs échanges au format HTTP pour récupérer les contenus multimédia.

La notion de session décrit l'ensemble des transactions qui ont un certain lien entre elles.

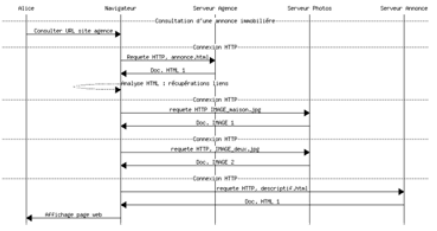
Par exemple : entrer dans le magasin virtuel, s'identifier, remplir son caddie, payer et quitter le site.

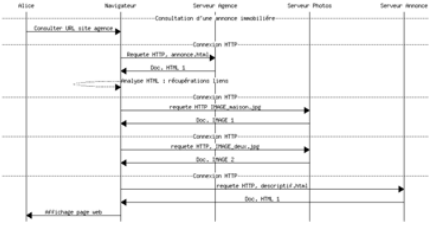


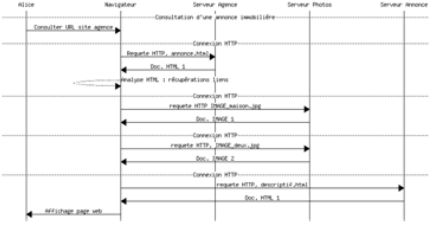
Utilisation de cookies, de données de formulaires, de contenus JSON, d'URL particulière (REST), *etc.*

Début de la session : navigation sur le site de l'agence immobilière

Temps





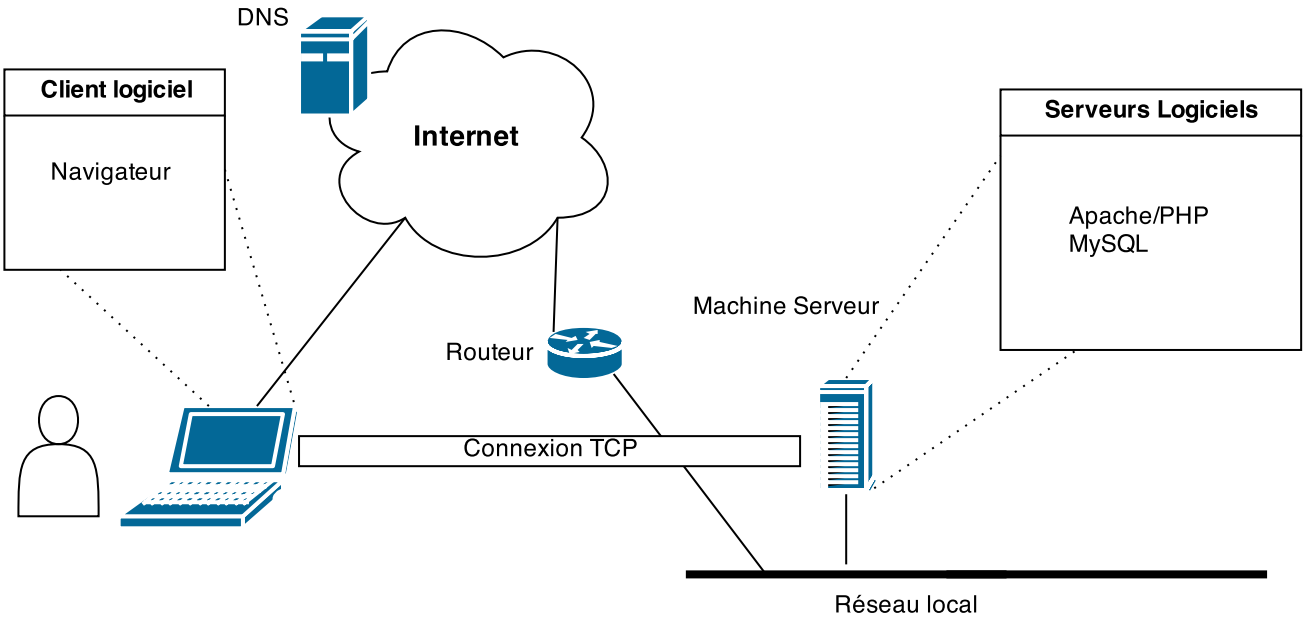


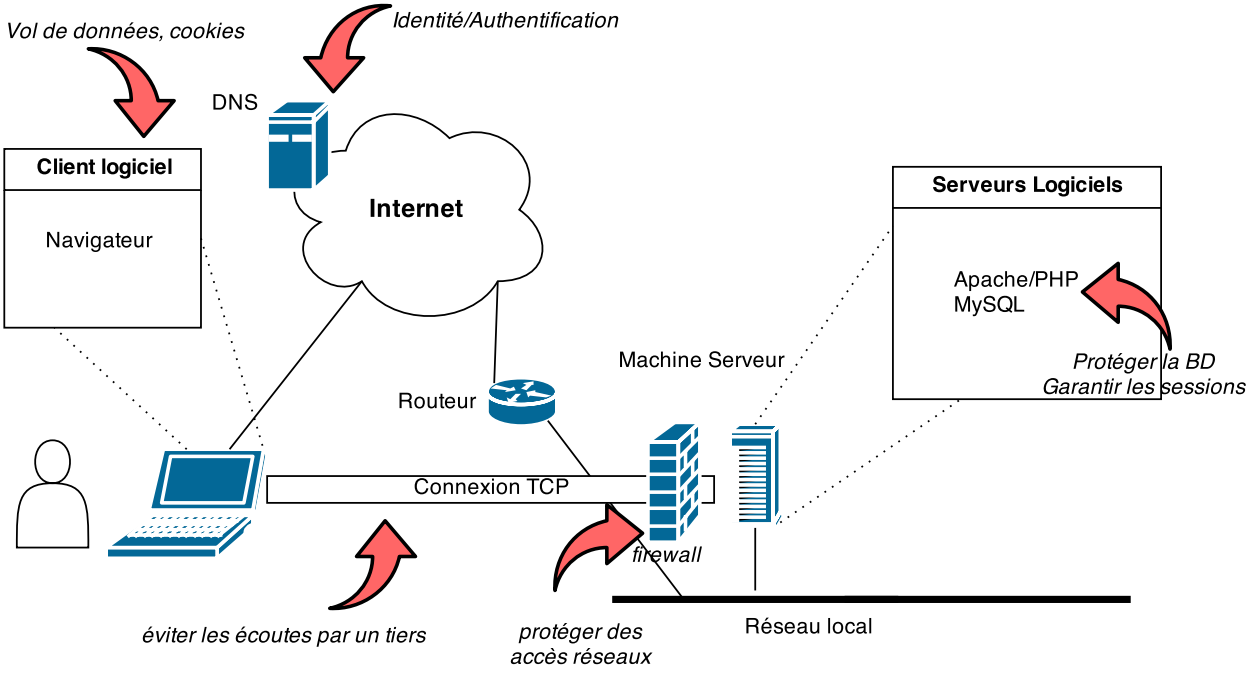
Session : plusieurs transactions

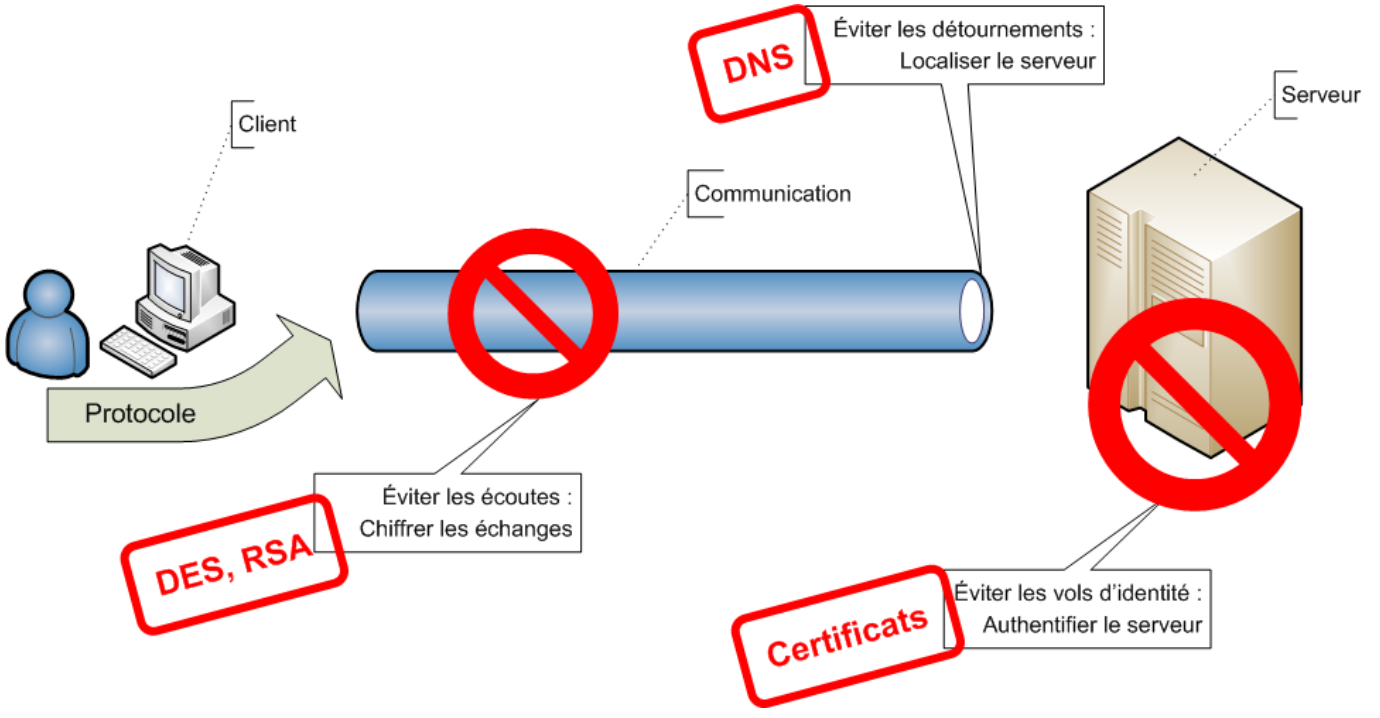
Fin de la session

Une **session** est composée de plusieurs **transactions**.

Chaque **transaction** peut donner lieu à un ou plusieurs **échanges**.


DNS
Client logiciel
Navigateur
Internet
Serveurs Logiciels
Apache/PHP
MySQL
Machine Serveur
Routeur
Connexion TCP
Réseau local


Vol de données, cookies
Identité/Authentification
DNS
Client logiciel
Navigateur
Internet
Serveurs Logiciels
Apache/PHP
MySQL
Protéger la BD
Garantir les sessions
Machine Serveur
Routeur
Connexion TCP
firewall
éviter les écoutes par un tiers
protéger des
accès réseaux
Réseau local


DNS
Éviter les détournements :
Localiser le serveur
Client
Serveur
Communication
Protocole
DES, RSA
Éviter les écoutes :
Chiffrer les échanges
Certificats
Éviter les vols d'identité :
Authentifier le serveur