

Internet Protocol version 4

Julien Montavont

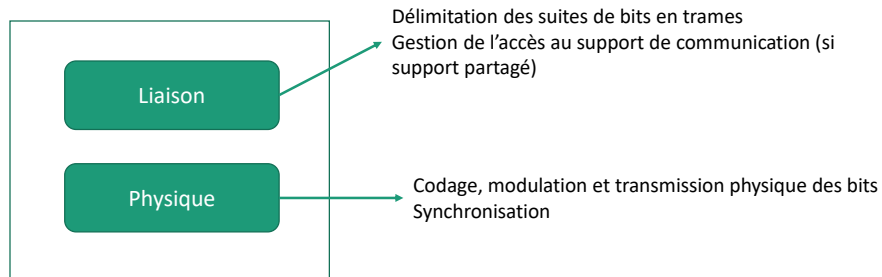
montavont@unistra.fr

Licence 2 Informatique

L'usage de ce support ne peut être qu'académique

Etat des lieux

- Couches 1 et 2 du modèle OSI



⇒ Réseau local implémente ces 2 couches

Ex : Ethernet, WiFi, Bluetooth, 4G, 802.15.4, LoRa, etc.

Réseau local et passage à l'échelle

- Couche physique + liaisons identiques
 - Même support physique, même manière de coder l'information
 - Même format de trames, d'adresses (de niveau 2)
 - ⇒ tout le monde parle le même langage
- Comment interconnecter des réseaux locaux différents ?
 - Support physique différent, codage différent, adresses différentes, etc.
 - ⇒ Langage différent
- Ex : 4G et Ethernet ?

Réseau local et passage à l'échelle

- **Solution** : on impose une et une seule technologie
 - Ex : Ethernet
 - Ethernet supporte le trafic de broadcast
 - Plus de 4 milliards d'utilisateurs sur Internet en 2018 [1]
 - Envoi en boucle de message en broadcast => **réseau tombe**
 - Ethernet est un réseau à diffusion
 - Chaque message unicast va inonder le réseau => **réseau tombe**
 - Ethernet commuté
 - **Explosion du nombre d'entrées dans les tables de commutation**
 - Plus de 4 milliards d'entrées, temps de recherche très long...
- ⇒ **solution inadaptée pour un réseau à (très) large échelle**

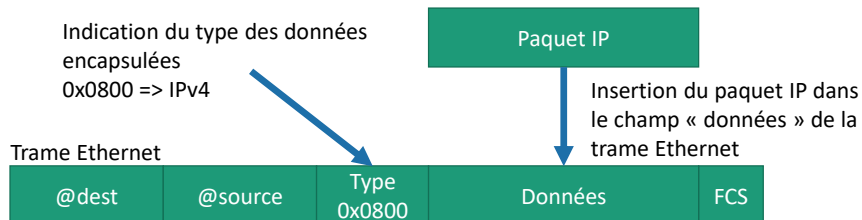
[1] <https://wearesocial.com/blog/2018/01/global-digital-report-2018>

Réseau local et passage à l'échelle

- Chaque technologie de communications possède des **avantages** et des **inconvénients**
 - Les interconnecter permet de tirer parti des avantages de chacune d'entre-elles
- Solution : définir un nouveau langage commun
 - ⇒ **couche réseau et spécification du protocole IP** (*Internet Protocol*)
 - **Abstraction** des couches physique et liaison sous-jacentes (boîte noire)
 - Format **unique des messages** et **adressage commun**
 - Nouveaux équipements intermédiaires jouant le rôle de **passerelles**
 - Le réseau global ne sera plus à diffusion
 - ⇒ Notion de **chemins / routes** pour atteindre chaque destination

Protocole IP version 4 (RFC 791)

- Envoi de datagrammes / paquets IP
 - De bout en bout à travers des réseaux interconnectés (Inter-Network => Internet)
 - Réseaux traversés sont hétérogènes (couches physique et liaison peuvent être différentes)
 - **Règles d'encapsulation du paquet IP pour chaque type de liaison**
 - Ex : paquet IP envoyé sur un réseau Ethernet



Fonctionnement IPv4

- **Plan de données** (*Data Plane*)

- Modèle de service **Best Effort**
 - Pertes possibles et non corrigées / détectées
 - Déséquencements possibles et non corrigés / détectés
 - Peu ou pas de contrôle de flux / congestion

- **Plan de contrôle / signalisation** (*Control Plane*)

- Complexité / intelligence dans le plan de contrôle
- Protocoles de routage dynamique pour acheminer les données
 - Apprentissage, calculs des meilleurs chemins, etc.

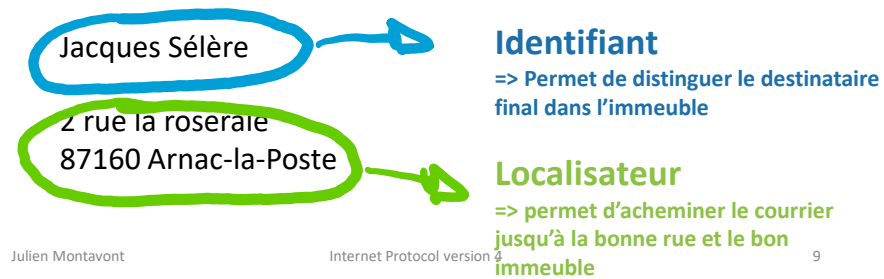
⇒ Étude en Master SIRIS

Routeur

- Equipement **de niveau 3**
 - Implémente couche physique, liaison et réseau
- Joue le rôle de **passerelle** entre 2 (ou plus) réseaux IP différents
 - Se charge d'acheminer les paquets au destinataire final
- Intervient sur
 - **Le plan de données** : **commutation des paquets IP** en fonction de la **FIB** (*Forwarding Information Base*)
 - **Le plan de contrôle** : **calcule les meilleurs chemins** vers les destinations, construction de la **RIB** (*Routing Information Table*) de laquelle découle la FIB

Adressage IP

- Adressage hiérarchique
 - Adresse IP joue 2 rôles
 - **Localisateur** => où se trouve ma destination
 - **Identifiant** => qui est ma destination
- Analogie avec une adresse postale

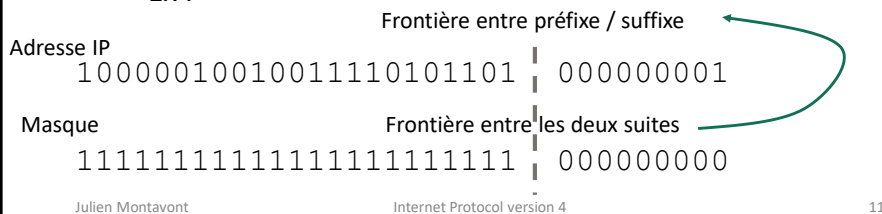


Adressage IP

- Comment distinguer ces deux rôles ?
 - **Préfixe**
 - constitue le **localisateur**, c-à-d l'adresse du réseau où se trouve actuellement l'équipement
 - **Suffixe**
 - constitue l'**identifiant** de l'interface
- Absence de sémantique (@ = suite de bits)
 - Ajout d'un séparateur entre préfixe et suffixe
 - **Masque réseau** (ancienne dénomination)
 - **Longueur du préfixe** (nouvelle dénomination)

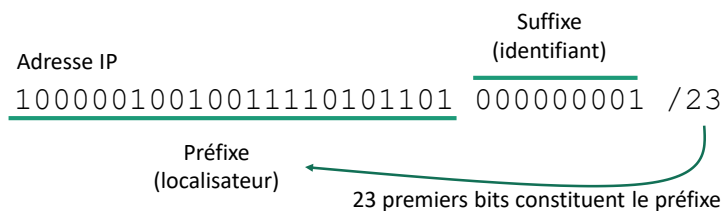
Exemple IPv4

- Adresses sur 32 bits
- Masque réseau sur 32 bits => suite (pouvant être nulle) de 1 suivie par une suite (pouvant être nulle) de 0
 - La **frontière** entre les deux suites indique la **frontière** entre **localisateur** et **identifiant** dans l'adresse correspondante
 - Ex :



Exemple IPv4

- Adresses sur 32 bits
- Notation **CIDR** (*Classless Inter-Domain Routing*)
 - Notation simplifiée **/X**
 - ⇒ Indique que les **X** premiers bits de l'adresse correspondent au préfixe
- Ex :



Adressage IPv4

- Adresses sur **32 bits**
 - $2^{32} = 4\,294\,967\,296$ adresses uniques possibles
- Représentation « humaine »
 - Binaire : difficile à écrire et lire
 - **Décimale à point** : base habituelle pour l'écriture des nombres
 - Groupement de blocs de 8 bits et conversion en base 10
 - ⇒ Chaque bloc est compris entre 0 et 255

10000010 01001111 01011010 00000001

130 79 90 1

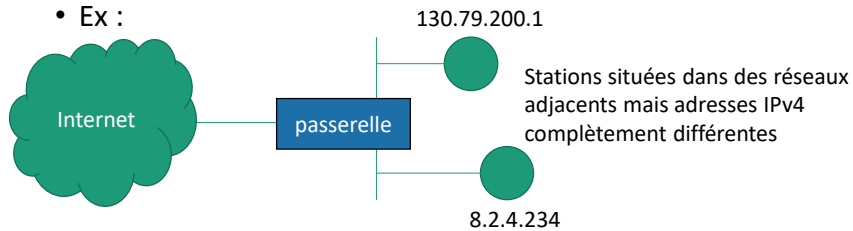
Représentation finale => 130 . 79 . 200 . 1

Adressage IPv4

- Représentation du masque réseau
 - Même représentation qu'une adresse
 - Ex :
 - 11111111111111111111111110000000 => 255.255.255.0
 - 11111111111100000000000000000000 => 255.240.0.0
- Représentation CIDR
 - Ex : 139.79.90.1/24
 indique que les 24 premiers bits de l'adresse constituent le préfixe réseau / l'adresse du réseau

Espace d'adressage

- Plus de 4 milliards d'adresses IPv4 différentes
 - Adressage plat => aucune structure dans l'allocation
 - Ex :



- ⇒ chemins spécifiques pour joindre chaque destination
- ⇒ **explosion du nombre d'entrée dans les tables de routage**
 - Plus de 4 milliards d'entrées, temps de recherche très long, ...

Julien Montavont

Internet Protocol version 4

15

Modèle en classes (*classfull*)

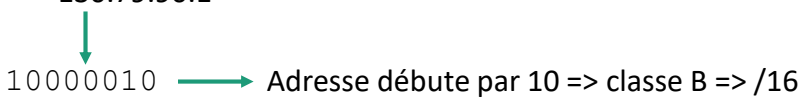
- Découpage de l'espace en classes

Classe	Plage	Longueur préfixe	Bit de poids fort des adresses
A	0.0.0.0 => 127.255.255.255	/8	0
B	128.0.0.0 => 191.255.255.255	/16	10
C	192.0.0.0 => 223.255.255.255	/24	110
D (multicast)	224.0.0.0 => 239.255.255.255	Non défini	1110
E (réservée)	240.0.0.0 => 255.255.255.255	Non défini	1111

5

Modèle en classes (*classfull*)

- Masque réseaux / longueur du préfixe est **implicite** dans ce modèle
 - Les bits de poids fort d'une adresse indiquent la classe et donc la longueur du préfixe contenu dans l'adresse
- Ex :
 - 130.79.90.1



Adresses spécifiques

- IPv4 supporte **trafic unicast**
- IPv4 supporte également le **trafic broadcast**
 - Permet de joindre tous les hôtes IPv4 du réseau local
 - Adresse de broadcast **liée au réseau courant**
 - Préfixe réseau + tous les bits du suffixe à 1
 - Ex : Réseau 130.79.0.0/16
 - Adresse de broadcast => 130.79.255.255
 - Paquet IP envoyé vers adresse de broadcast ne passe pas les routeurs
 - ⇒ inondation du réseau local mais pas des réseaux externes (Internet)
- IPv4 supporte trafic **multicast**
- IPv4 supporte **trafic local** (communication entre des applications d'un même hôte) => **boucle locale**

Problèmes du modèle en classes

- Modèle **trop rigide**
 - Attribution d'un bloc en classe A => unique réseau pouvant accueillir plus de 16 millions de postes...
- Ajout des **sous-réseaux** (1984) [RFC917]
 - Niveau hiérarchique supplémentaire entre le préfixe réseau et l'identifiant de l'interface

`<network-number><subnet-number><host-number>`

- Possibilité de créer des sous-réseaux locaux en redécoupant un bloc en sous-blocs
 - Ex : bloc de classe B découpé en 256 sous-réseaux disposant chacun de 255 adresses

Problèmes du modèle en classes

- Blocs attribués de manière FIFO sans tenir compte des besoins
 - ⇒ **Gâchis** car les adresses non utilisées ne peuvent pas être réutilisées par une autre entité
 - Ex : allocation d'un bloc de classe A (donc plus de 16 millions d'adresses) pour une entité qui dispose de 1000 postes informatiques...
- Début des années 1990, **pénurie d'adresses...**
 - Nouveaux pays / organisations / entreprises avec des besoins fort
 - ⇒ attribution de multiples blocs de classe C à une même entité
 - ⇒ **explosion du nombre d'entrées** dans les tables de routage

Modèle sans classe (*classless*)

- **Classless Inter-Domain Routing** - CIDR (1993)
[RFC1517, RFC1518, RFC1519]

- Abrogation des classes d'adresse
⇒ Le masque / la longueur du préfixe est **explicite**
- Allocation plus flexible
⇒ Utilisation de préfixes de longueur variable (*Variable-Length Subnet Masking* – VLSM)
 - Ex : préfixes de longueur 5, 13, 23, etc.
- Nouvelle représentation de la longueur d'un préfixe
⇒ Notation CIDR : / longueur_préfixe
 - Ex : 130.79.90.1/23
- Agrégation de préfixes
⇒ Réduit le nombre d'entrées dans les tables de routage
 - Ex : 2 entrées agrégées en 1

130.79.90.0/24  130.79.90.0/23
130.79.91.0/24

Julien Montavont

Internet Protocol version 4

21

Allocation des blocs CIDR

- Allocation hiérarchique
 - *Internet Assigned Numbers Authority* (IANA)
 - gère l'allocation des blocs de haut niveau (/8) aux **RIR**
 - *Regional Internet Registry* (RIR)
 - Responsable d'une zone géographique spécifique
 - Redécoupe les blocs pour allocation aux **LIR**
 - *Local Internet Registry* (LIR)
 - Réattribue les blocs à des tiers (clients)
 - Généralement opérateur télécoms (*Internet Service Provider* – ISP)

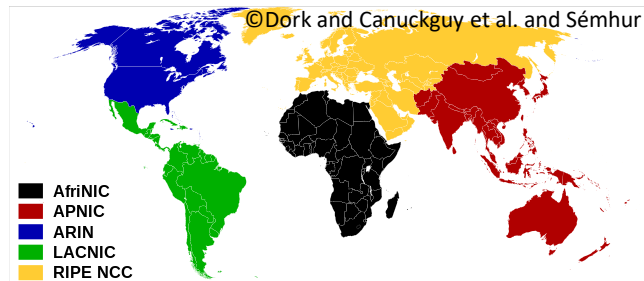
Julien Montavont

Internet Protocol version 4

22

Répartition des RIRs

- African Network Information Center (AfriNIC)
- Asian-Pacific Network Information Centre (APNIC)
- American Registry for Internet Numbers (ARIN)
- Latin America and Caribbean Network Information Centre (LACNIC)
- Réseaux IP Européens Network Coordination Centre (RIPE NCC)



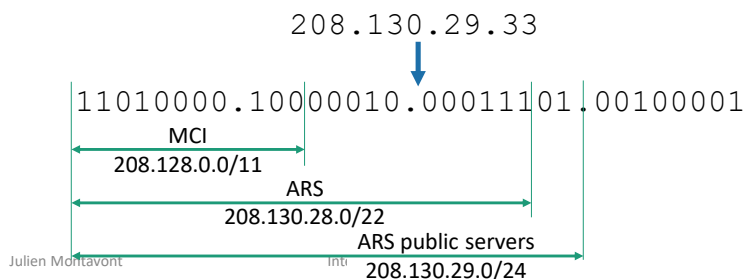
Julien Montavont

Internet Protocol version 4

23

Allocation des blocs CIDR

- Ex (tiré de wikipédia) :
 - 208.130.29.33 était utilisée par le site www.freesoft.org
 - 208.128.0.0/11 avait été attribué par le RIR ARIN à l'opérateur MCI
 - La compagnie Automation Research Systems (ARS) à loué une ligne Internet chez MCI et a reçu la plage 208.130.28.0/22
 - Dans ce bloc, ARS a découpé le bloc 208.130.29.0/24 pour ses serveurs publics, dont www.freesoft.org



Julien Montavont

24

Allocation des blocs CIDR

- IETF recommande l'obtention d'une plage d'adresses IP
 - Après de son opérateur (ISP) en cas d'opérateur unique
 - ⇒ plage d'adresses liée à l'opérateur
 - ⇒ changement d'adresses si changement d'opérateur
 - Directement auprès du RIR si opérateurs multiples
 - *provider-independent address space*
 - ⇒ plage d'adresses indépendante d'un opérateur
 - ⇒ conservation d'adresses si changement d'opérateur

Adressage public / privé

- Adresses publiques
 - Joignables / routables depuis tout point d'Internet
 - Uniques et payantes
 - Allocation via IANA => RIR => LIR => Utilisateur
- Adresses privées [RFC1918]
 - Non joignables / routables depuis tout point d'Internet
 - Non unique
 - ⇒ mêmes plages utilisables simultanément car réseaux non interconnectés
 - Gratuites
 - Allocation ad hoc

Plage	Nombre d'@
10.0.0.0/8	16 777 216
172.16.0.0/12	1 048 576
192.168.0.0/16	65 535

Blocs d'adresses réservés

Bloc	scope	Usage
100.64.0.0/10	Réseau privé	Carrier-grade NAT
127.0.0.0/8	Hôte	Adresse de boucle locale
169.254.0.0/16	Sous-réseau	Adresses de lien-local (cf. IPv6)
192.88.99.0/24	Internet	6to4
198.18.0.0/15	Réseau privé	Benchmark testing
198.51.100.0/24	Documentation	
203.0.113.0/24	Documentation	
224.0.0.0/4	Internet	Multicast IPv4
240.0.0.0/4	Internet	Réservé pour usage futur

Format des paquets IPv4

```

0                               1                               2                               3
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|version| IHL |   DSCP   |ECN|          total length          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|          identification          |flags|  fragment offset  |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
| time to live |  protocol  |          header checksum       |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|          source address          |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|          destination address     |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|          options                  |padding|
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
//                                data                                //
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Champs en-tête IPv4

- **Version** – 4 bits : version du protocole IP
- **IHL** (*Internet Header Length*) – 4 bits : taille de l'en-tête exprimée en nombre de blocs de 32 bits
 - En-tête de taille variable car accepte des options
- **DSCP** (*Differentiated Service Code Point*) – 6 bits : code pour marquer le paquet afin de lui appliquer un traitement particulier (qualité de service) [RFC3260]
- **ECN** (*Explicit Congestion Notification*) – 2 bits : permet de notifier des sources réactives d'une congestion sur le réseau [RFC3168]

Champs en-tête IPv4

- **Total length** – 16 bits : longueur total du paquet (en-tête + données) exprimée en octets
- **Identification** – 16 bits : identifiant du paquet en cas de fragmentation (segmentation du paquet en plus petit fragments)
- **Flags** – 3 bits : drapeaux pour la fragmentation (*More Fragment, Don't Fragment, Reserved*)
- **Fragment offset** – 13 bits : place du fragment dans le paquet original
- **Time to live** – 8 bits : durée de vie du paquet
 - Si expiration alors paquet supprimé pour éviter qu'un paquet ne boucle indéfiniment dans le réseau
 - exprimée en seconde mais interprété en pratique comme un nombre de sauts autorisés

Champs en-tête IPv4

- **Protocol** – 8 bits : type des données véhiculées dans le champs « data »
 - Numéro attribué à chaque protocole de manière standardisé
 - Ex : 0x06 => TCP
- **Header checksum** – 16 bits : contrôle d'erreurs permettant de détecter des erreurs sur le paquet (en-tête + données)
- **Source address** – 32 bits : adresse IPv4 de l'expéditeur original du paquet
- **Destination address** – 32 bits : adresse IPv4 de la destination finale du paquet

Options IPv4

- Environ 30 options différentes
 - Source routing
 - Route recording
 - Router alert
 - Etc.
- Liste complète
 - <http://www.iana.org/assignments/ip-parameters/ip-parameters.xhtml>
- Aujourd'hui
 - Majorité des options est **obsolète** ou **non traitée** par les routeurs

Interactions niveaux 2 et 3

- Couche 3 construit un paquet IP
- Paquet IP doit être encapsulé dans trame de niveau 2 (ex : trame Ethernet)
- Problème : **à qui adresser la trame au niveau 2 ?**
 - Détermination en fonction de **la table de routage** de l'expéditeur
 - 2 cas possibles :
 - Le destinataire IP se trouve dans le même réseau que la source
⇒ **le destinataire au niveau 2 sera le même que celui de niveau 3**
 - Le destinataire IP se trouve dans un réseau distant
 - le paquet IP doit être passé au routeur qui sera en charge de son acheminement vers la destination
⇒ **Le destinataire au niveau 2 sera le routeur**

Julien Montavont

Internet Protocol version 4

33

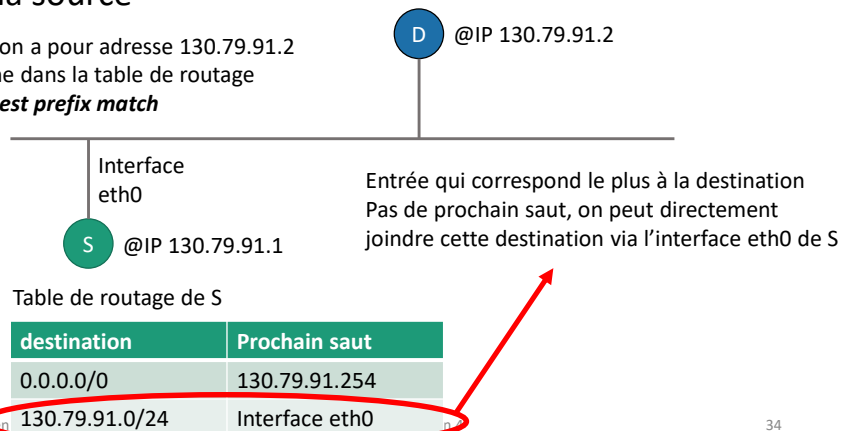
Interactions niveaux 2 et 3

- Exemple : destinataire dans le même réseau IP que la source

Destination a pour adresse 130.79.91.2

Recherche dans la table de routage

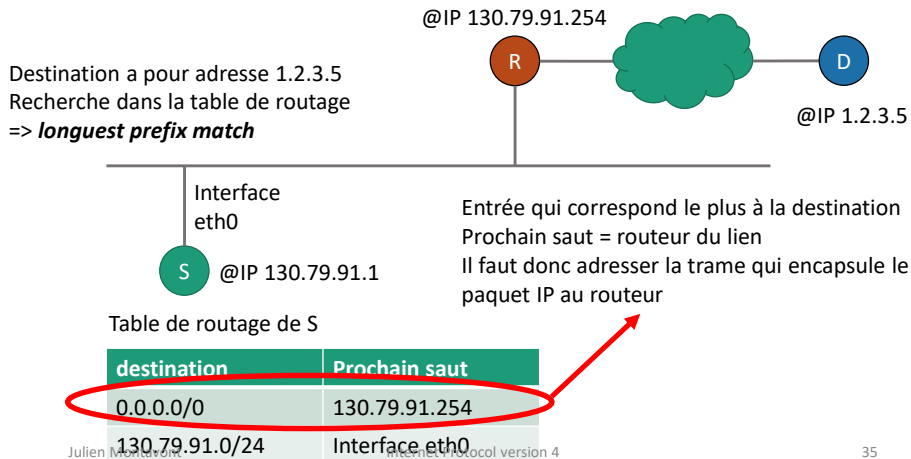
=> **longest prefix match**



34

Interactions niveaux 2 et 3

- Exemple : destinataire dans un réseau distant

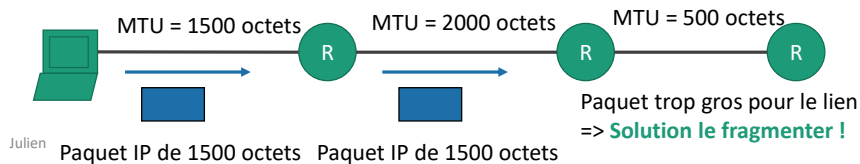


Résolution d'adresse

- Comment déterminer l'adresse de niveau 2 qui correspond à une adresse de niveau 3 ?
 - La source connaît l'adresse IP du destinataire, ou
 - La source connaît l'adresse IP du routeur (depuis la table de routage)
- Address Resolution Protocol (ARP)**
 - Emission d'une requête ARP (broadcast)
 - On pose la question suivante à tout le réseau local :
« *Quelle est l'adresse de niveau 2 qui correspond à telle adresse de niveau 3 ?* »
 - Réponse ARP (unicast)
 - Contient **correspondance adresse de niveau 3 – adresse de niveau 2**
 - Ex : 130.79.91.2 ⇔ A0:BC:DF:12:A2:45
 - Aucune réponse => personne ne possède cette adresse sur le réseau local
 - Enregistrement des correspondances découvertes dans le **cache ARP**
 - Évite de faire systématiquement des requêtes
 - Entrées soft-state (associée à une durée de vie)
 - Ex :
172.16.1.3 at 00:00:C0:5A:42:C1 [ether] on eth0

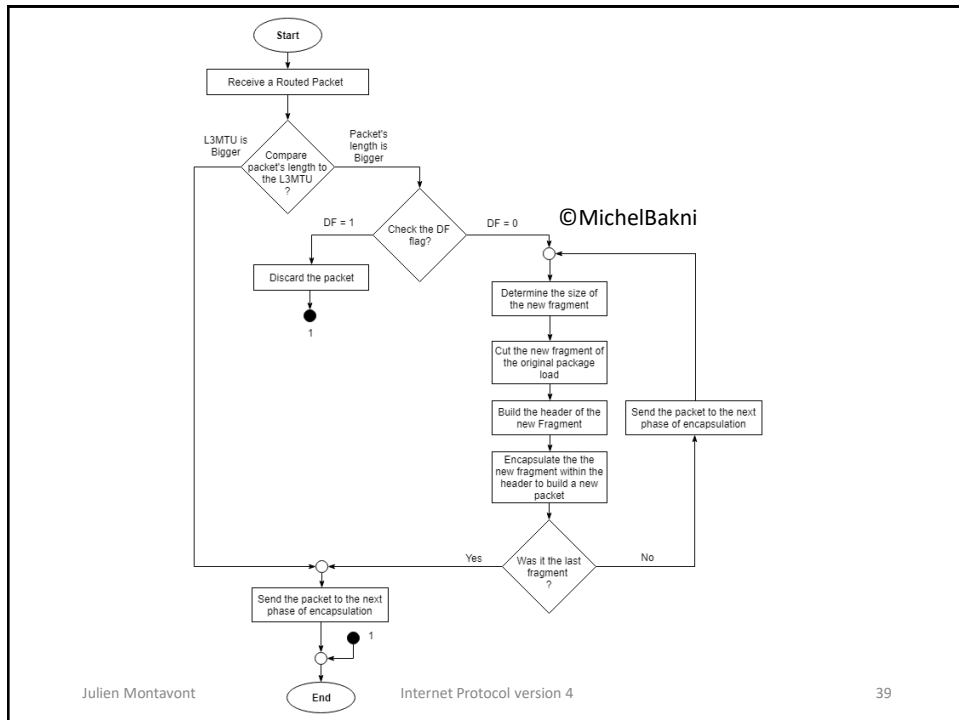
Fragmentation IP

- Paquet IP va traverser des réseaux hétérogènes
 - Support de communication, codage, modulation, format des trames, etc.
 - Le **MTU** (*Maximum Transfer Unit*) peut être différent suivant la technologie sous-jacente utilisée
 - Ethernet = 1500 octets, WiFi = 2272 octets, etc.
 - Que faire si un paquet IP est trop grand pour circuler sur un lien ?
 - Ex :



Fragmentation IP

- Routeur en charge de fragmenter les paquets (IPv4)
 - Découpage du paquet en plus petits paquets IP appelés **fragments**
 - Chaque fragment doit avoir une taille multiple de 8 octets
 - Attribution d'un identifiant de paquet pour retrouver tous les fragments d'un même paquet (champ **identification** de l'en-tête)
 - Flag **More Fragment** positionné à 1 pour tous les fragments sauf le dernier
 - Chaque fragment comporte sa position dans le paquet original (champ **fragment offset**)
- Un fragment est un paquet IP qui peut **à nouveau être fragmenté**
- Destination en charge de **reconstruire le paquet original**



Informations nécessaires pour communiquer sur Internet

- Disposer d'une **adresse IPv4 publique, topologiquement valide** (correspond à notre localisation logique)
- Connaître **l'adresse du réseau local** (pour déterminer la longueur du préfixe)
- Connaître **l'adresse du routeur par défaut** du lien
- Comment configurer ces paramètres ?
 - Configuration **manuelle** (ex : commande « ip » sous UNIX)
 - Configuration **automatique** (protocole *Dynamic Host Configuration Protocol* – DHCP) [RFC2131]

Internet Control Message Protocol (ICMP)

- Protocole de signalisation de niveau 3
- Information sur **erreurs** au niveau IP
 - **Destination non joignable**
 - Réseau non joignable, hôte non joignable, port non joignable, réseau interdit administrativement, paquet trop grand et interdiction de fragmenter, etc.
 - **Durée de vie**
 - Temps de vie d'un paquet expiré pendant l'acheminement, temps pour réassembler les fragments d'un paquet expiré
 - **Pb de paramètre**
 - Mauvaise longueur, option manquante, etc.

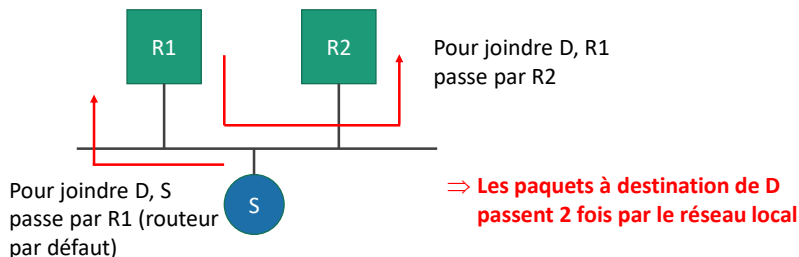
Julien Montavont

Internet Protocol version 4

41

Internet Control Message Protocol (ICMP)

- **Redirection de paquets**
 - **Meilleur chemin** pour le réseau de destination, le destinataire, etc.



Julien Montavont

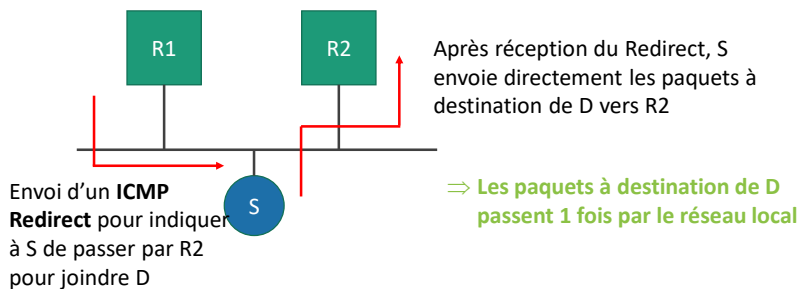
Internet Protocol version 4

42

Internet Control Message Protocol (ICMP)

• Redirection de paquets

- **Meilleur chemin** pour le réseau de destination, le destinataire, etc.



Julien Montavont

Internet Protocol version 4

43

Internet Control Message Protocol (ICMP)

• Test d'accessibilité IP

- Utilitaire « ping »
 - Emission d'un **ICMP Echo Request**
 - Si réception d'un **ICMP Echo Reply**, alors connectivité IP vérifiée entre 2 hôtes

```
ping google.fr
```

```
Envoi d'une requête 'ping' sur google.fr [216.58.204.131] avec 32 octets de données
```

```
Réponse de 216.58.204.131 : octets=32 temps=76 TTL=51
```

```
Réponse de 216.58.204.131 : octets=63 temps=76 TTL=51
```

```
...
```

- D'autres fonctionnalités aujourd'hui obsolètes

Julien Montavont

Internet Protocol version 4

44

Conclusions

- Couche réseau permet l'**interconnexion de réseaux de technologies différentes**
- Protocole IP constitue le « **langage commun** » pour la communication à travers tous ces réseaux
- IP fournit un service « **best effort** »
- Adresse IP possède 2 rôles : **localisateur et identificateur**
 - Dans IPv4 les adresses sont sur **32 bits**
- Aujourd'hui : modèle d'adressage **sans classe** (CIDR) et **longueur de préfixe variable** (VLSM)