

IPv6

- Présentation d'IPv6
- Méthodes d'accès à l'internet v6
- IPv6 avec GNU/Linux
- Problèmes potentiels
- Ressources en ligne

Présentation d'IPv6

- Motivations
- Description et comparaison avec IPv4
- Adressage IPv6
- IPv6 et DNS

Motivation de la transition vers IPv6

- En principe : une adresse IP publique par machine connectée à internet
- Mais pénurie d'adresses publiques IPv4
- En pratique : attribution d'une adresse publique par connexion et utilisation d'adresses privées en combinaison avec les techniques de translation d'adresse NAT (Network Address Translation)
- Mais le NAT présente des inconvénients

Inconvénients du NAT

- ajoute de la complexité
- casse la connectivité de bout en bout
- perturbe certains protocoles et applications :
 - protocoles FTP, SIP, H.323, RTSP...
 - applications peer-to-peer au sens large : messagerie instantanée, échange de fichiers, jeux en réseau...
 - IPSec incompatible avec la modification des paquets

Contourner les inconvénients du NAT

- Techniques pour contourner les inconvénients
 - NAT « intelligent »
 - redirection de ports
 - mode passif FTP
 - STUN pour la VoIP
 - Hamaichi pour les jeux
 - NAT-T pour IPSec
- Mais : cela ajoute encore de la complexité

Motivations d'IPv6

- Agrandir l'espace d'adressage pour éviter le recours au NAT
- Corriger des défauts de conception d'IPv4 (simplifier l'en-tête, la gestion de la fragmentation et des checksums)
- Ajouter de nouvelles fonctionnalités
 - sécurité
 - mobilité
 - autoconfiguration

Description d'IPv6

- IPv6 reprend les fondements d'IPv4
 - transmission de paquets
 - connectivité de bout en bout
 - routage hiérarchique par préfixe de proche en proche
 - protocoles de transport TCP, UDP...
- Mais formats d'en-tête de paquet différents
- Pas de compatibilité de la couche réseau
- IPv4 et IPv6 sont traités comme deux protocoles différents (principe de « double pile »)

Comparaison IPv4 - IPv6

	IPv4	IPv6
Ethertype	0x0800	0x86dd
Taille d'adresse	32 bits (4 octets)	128 bits (16 octets)
Taille de préfixe de sous-réseau	variable /8 à /31	fixée /64
Taille de paquet minimum	68 octets	1280 octets
Taille de l'en-tête	variable 20 à 60 octets	fixe 40 octets

- Suppression des adresses de broadcast
- Remplacement du broadcast par du multicast
- Evolution d'ICMP en ICMPv6
- Remplacement d'ARP par un sous-ensemble d'ICMPv6, « Neighbor Discovery »
- Remplacement d'IGMP par un sous-ensemble d'ICMPv6
- Suppression de la fragmentation par les routeurs intermédiaires et généralisation de Path MTU Discovery

Neighbor Discovery

- 5 types de messages ICMPv6
 - Neighbor Solicitation (NS)
 - Neighbor Advertisement (NA)
 - Router Solicitation (RS)
 - Router Advertisement (RA)
 - Redirect
- utilise des adresses multicast IPv6 et ethernet

Adressage IPv6

- Représentation textuelle des adresses et préfixes IPv6
- Visibilité (« scope ») des adresses
- Types d'adresses
- Description de l'espace d'adressage

Notation hexadécimale

décimal	hexadécimal	binaire
0	0	0000
1	1	0001
2	2	0010
3	3	0011
4	4	0100
5	5	0101
6	6	0110
7	7	0111
8	8	1000
9	9	1001
10	A	1010
11	B	1011
12	C	1100
13	D	1101
14	E	1110
15	F	1111

Représentation des adresses IPv6

- Représentation d'une adresse de 128 bits sous forme de 8 groupes de 4 chiffres hexa (16 bits) séparés par « : »
 - 2001:0db8:0000:0000:0123:4567:89ab:cdef
- Représentation compacte
 - On peut supprimer les zéros en début de groupe :
2001:db8:0:0:123:4567:89ab:cdef
 - on peut remplacer une fois une suite de groupes à 0 par « :: » : 2001:db8::123:4567:89ab:cdef

Représentation des adresses IPv6

- Représentation embarquant une adresse IPv4
 - `::ffff:192.0.2.1 = 0:0:0:0:0:ffff:c000:0201`
- Représentation entre [] dans les URL
 - `http://[2001:db8::123:4567:89ab:cdef]:80/`
- Représentation embarquant l'interface avec une adresse link-local
 - `fe80::2e0:29ff:fe04:b661%eth0` (Linux, BSD)
 - `fe80::2e0:29ff:fe04:b661%5` (Windows)

Visibilité (« scope ») et types

- Visibilité
 - hôte (host)
 - lien (link local)
 - site (site local)
 - global
- Types
 - unicast
 - multicast
 - anycast

Adresses unicast

- **adresse indéfinie (unspecified)**
 - `0:0:0:0:0:0:0:0 = ::`
- **adresse d'hôte local (localhost, loopback)**
 - `0:0:0:0:0:0:0:1 = ::1`
- **adresses compatibles IPv4**
 - `0:0:0:0:0:0:x.y.z.t/96 = ::x.y.z.t/96`
- **adresses mappées en IPv4**
 - `0:0:0:0:0:0:ffff:x.y.z.t/96 = ::ffff:x.y.z.t/96`

Adresses globales unicast

- **actuellement allouées**
 - 2000::/3
- **adresses Teredo**
 - 2001::/32
- **adresses d'exemple et documentation**
 - 2001:db8::/32
- **adresses 6to4**
 - 2002::/16
- **adresses anciennement allouées au 6bone**
 - 3ffe::/16 et 5f00::/8

Autres adresses unicast

- Adresses locales unique (ULA)
 - fc00::/7
- Adresses link-local unicast
 - fe80::/10
- Adresses site-local
 - fec0::/10
 - obsolètes, remplacées par les adresses locales uniques

Adresses locales uniques (ULA)

- plage fc00::/7 (fd00::/8 actuellement)
- usage équivalent aux adresses privées IPv4
10.0.0.0/8, 172.16.0.0/12, 192.168.0.0/16
- un préfixe /48 par site (65536 sous-réseaux)
- 2^{41} soit 2000 milliards de préfixes créés par un algorithme pseudo-aléatoire
- but : éviter les collisions
- générateur : <http://www.kame.net/~suz/gen-ula.html>

Adresses link-local unicast

- préfixe fe80::/10
- les 64 bits de droite sont constitués de l'EUI-64 modifié dérivé de l'adresse MAC de l'interface
- adresses de service utilisées dans certains types ICMPv6 comme Neighbor Discovery
- exemple :
 - adresse MAC : 00:e0:29:04:b6:61
 - EUI-64 modifié : 0**2**:e0:29:**ff:fe**:04:b6:61
 - adresse link- local : fe80::2e0:29ff:fe04:b661

Adresses multicast

- plage multicast ff00::/8
- adresses link-local multicast notables utilisées par Neighbor Discovery et Router Discovery :
 - ff02::1 : « all nodes » (hôtes et routeurs)
 - ff02::2 : « all routers » (routeurs seulement)
 - ff02::3 : défini à tort comme « all hosts » dans certains fichiers /etc/hosts, défini dans un draft mais non retenu dans le RFC final

Adresses anycast

- Définies dans l'espace unicast
- Exemple : l'adresse de sous-réseau constituée du préfixe du sous-réseau et de l'identificateur d'interface nul est une adresse multicast destinée à n'importe quel routeur directement connecté à ce sous-réseau

IPv6 et DNS

- Toujours dans la classe IN (Internet)
- Enregistrement d'adresse IPv6 : type AAAA
- Exemple :

```
www.example.net. IN AAAA 2001:db8::123:4567:89ab:cdef
```

- Enregistrement inverse : type PTR au format « nibble » sous ip6.arpa

- Exemple :

```
f.e.d.c.b.a.9.8.7.6.5.4.3.2.1.0.0.0.0.0.0.0.0.0.8  
.b.d.0.1.0.0.2.ip6.arpa IN PTR www.example.net.
```

IPv6 et DNS

- Anciens types et formats abandonnés :
 - enregistrement d'adresse de type A6
 - enregistrement inverse au format bitstring
 - zones inverses sous `ip6.int` (vieilles versions de glibc, Debian woody, patch IPv6 pour Windows 2000 SP1)
- Des enregistrements A et AAAA peuvent être définis pour un même nom de domaine
- Les autres types DNS (NS, MX...) ne sont pas impactés par IPv6 et restent inchangés

Méthodes d'accès à l'internet IPv6

- IPv6 natif par le FAI/hébergeur : ethernet, PPP...
- accès avec encapsulation IPv6 dans IPv4
 - tunnel point à point, tunnel broker
 - tunnel automatique 6to4
 - Teredo

FAI avec IPv6 natif

- Nerim : PPP en ADSL, ethernet ou IPoA en SDSL, préfixe /48
- Free dégroupé avec Freebox : ethernet, préfixe /64
- FDN : PPP
- Orange : expérimentation IPv6 sur PPP entre 2005 et 2007 terminée ; une offre IPv6 pour Orange Business serait prévue pour le 1^e semestre 2009
- Autres ?

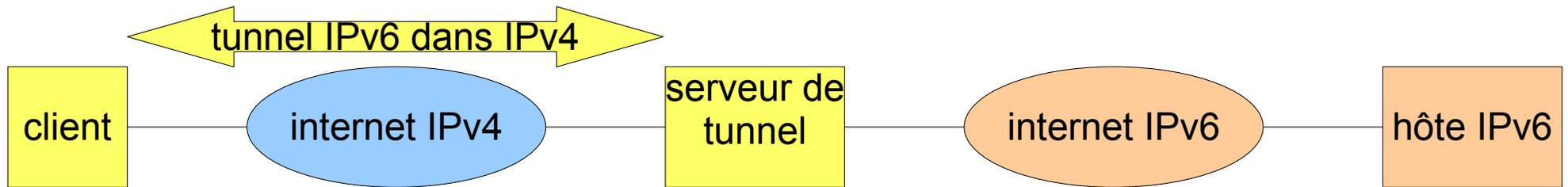
Hébergeurs avec IPv6 natif

- Dedibox : une seule adresse IPv6 autoconfigurée
- OVH : un (gros) bloc d'adresses (selon offre)
- Autres ?

Tunnel brokers

- Tunnel point à point transportant IPv6 dans IPv4
- Plusieurs protocoles d'établissement du tunnel et d'encapsulation :
 - 6in4 (protocole IP 41) IPv6 dans IPv4
 - AYIYA (Anything In Anything), IPv6 dans UDP
 - TIC (Tunnel Information and Control)
- Tunnels brokers :
 - SixXS
 - Hurricane Electric
 - Freenet/Hexago

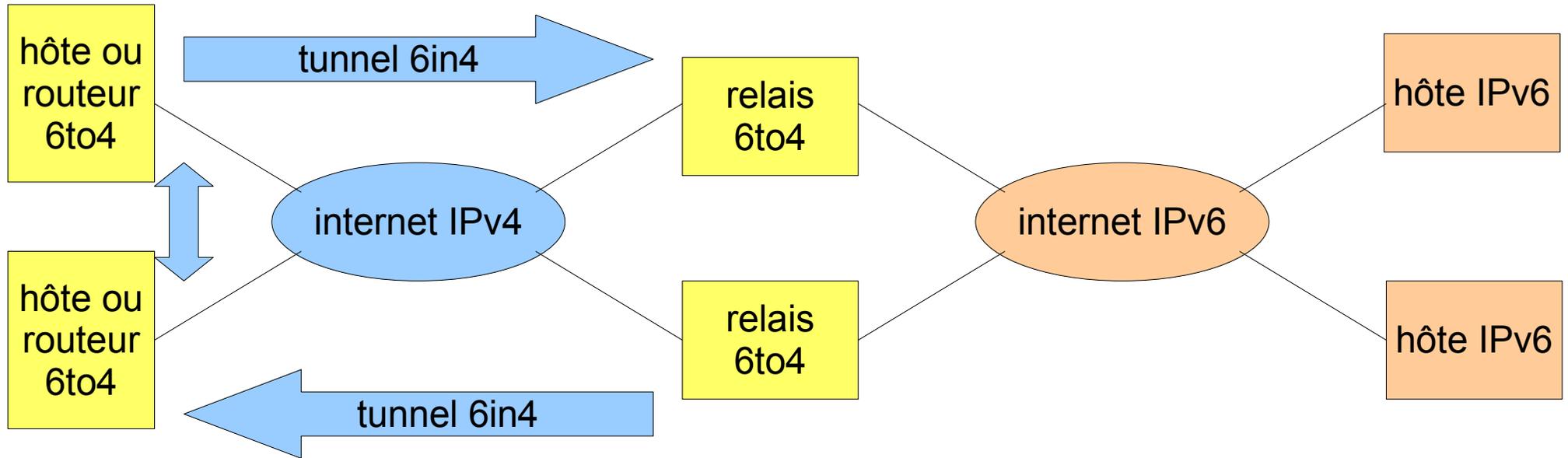
Tunnel point à point



Tunnel automatique 6to4

- Encapsulation 6in4 (protocole IP 41)
- Un préfixe IPv6 global 6to4 est associé à chaque adresse IPv4 publique : `2002:<adresseipv4>::/48`
- Un routeur 6to4 gère l'encapsulation pour un réseau 6to4
- Communication directe entre réseaux 6to4
- Communication entre un réseau 6to4 et un hôte IPv6 via un relais anycast 6to4 `192.88.99.1`

6to4



Teredo

- Encapsulation IPv6 dans UDP dans IPv4
- Une adresse IPv6 dans $2001::/32$ est attribuée à chaque client Teredo, basée sur son adresse IPv4 publique, son port UDP et l'adresse IPv4 du serveur Teredo utilisé
- Protocole compliqué conçu pour traverser les pare-feu et NAT
- Intégré dans Windows Vista

IPv6 et GNU/Linux

- Cf. Linux IPv6 HOWTO sur TLDP :
<http://www.tldp.org/HOWTO/Linux+IPv6-HOWTO/>