

Administration Réseau

Adressage IPv4 :

une adresse comporte deux parties : partie réseau et partie hôte

Classe A : premier bits à 0, les 7 autres déterminent le réseau et les 24 derniers les hôtes

→ adresses de 0. à 127.

Classe B : deux premiers bits à 10, les 14 suivants déterminent le réseau et les 16 derniers les hôtes

→ adresses de 128. à 191.

Classe C : trois premiers bits à 110, les 21 suivants déterminent le réseau et les 8 derniers les hôtes

→ adresses de 192. à 223.

Classe D : quatre premiers bits à 1110

→ ne correspond pas à des réseaux mais à des adresses *multicast*

→ utilisé par IGMP

→ adresses de 224. à 239.

Classe E : quatre premiers bits à 1111

→ adresses de 240. à 255.255.255.255

ARP (*Adresse Resolution Protocol*) et **RARP** (*Reverse ARP*) :

→ ARP utilise la diffusion (*broadcast*) sur un seul réseau physique

— recherche quelle interface réseau possède telle adresse IP

→ RARP permet à une machine de déterminer son adresse IP à partir de son adresse matérielle

ICMP (*Internet Control Message Protocol*) : permet aux machines d'échanger des informations

→ contrôle de routage (ICMP Redirect) si meilleure route existe

→ notification d'erreur

→ gestion du réseau

UDP (*User Datagram Protocol*) :

→ service en mode non connecté sans reprise d'erreur

→ messages peuvent être perdus, dupliqués, etc.

TCP (*Transfer Control Protocol*) :

→ circuit virtuel en mode connecté

→ établissement d'une connexion bidirectionnelle entre 2 points

→ mécanisme de fenêtre → contrôle de flux et performance de la liaison

Passerelles par défaut :

→ routage direct : la destination fait partie du sous-réseau

→ routage indirect : la destination est joignable *via* une passerelle

DHCP (*Dynamic Host Control Protocol*) :

→ fournit aux clients : adresse IP, masque, passerelle par défaut, DNS, WINS, durée du bail

→ mécanisme

— client diffuse DHCP DISCOVER

— serveur diffuse DHCP OFFER

— client envoie DHCP REQUEST

— serveur envoie DHCP ACK

→ les requêtes ne peuvent traverser les routeur qui si il existe le service *BOOTP Relay Agent*

Application (couche 5, 6, 7)
TCP/UDP (couche 4)
IPv4 - IPv6 (couche 3)
Ethernet (couche 2)
Physique (couche 1)

Routage IP

→ deux types d'algorithmes : *Vector Distance* et *Link-State*

→

Algorithme Link State :

- les passerelles maintiennent une carte (topologie) du réseau
- ne communiquent pas toutes les destinations connues (contrairement à *Vector Distance*)
- test périodique de l'état des liens et diffusion du résultat
- si réception de message, mis à jour de la topologie avec *Dijkstra*

NAT (*Network Address Translation*) :

- full cone : une adresse publique pour une adresse privée
- restricted cone : quelques adresses publiques pour plusieurs adresses privées
- port restricted cone : idem, mais le correspondant doit utiliser le port source du paquet reçu
- symmetric : table de correspondance entre adresse privée, adresse publique et adresse du correspondant
- NATP (?)

CISCO

Séquences de démarrage :

1. chargement du programme d'amorçage général
2. recherche de l'image du système d'exploitation (Cisco IOS) → mémoire flash, réseau
3. chargement du système d'exploitation, composants matériels et logiciels
4. parcourt du fichier de configuration → si inexistant, routine de configuration initiale

Services et outils réseau

DNS (*Domain Name Server*) :

- un domaine est un sous-arbre de l'espace « nom de domaine »
- types de serveur
 - primaire : fait autorité sur le domaine
 - secondaire : collecte les informations de zone

Principe d'un sniffer :

- toutes les machines d'un réseau local partagent le même câble → elles reçoivent toutes les paquets
- passage en promiscuous mode pour remonter au noyau tous les paquets (filtrés par la carte réseau en temps normal)

tcpdump	analyse en direct et enregistrement dans un fichier du trafic
ethereal	analyse de trame
ngrep	recherche des motifs dans des trames
netcat	permet de lire/écrire au travers de connexions TCP/UDP
ntop	statistiques sur l'utilisation du réseau
iptraf	statistiques (rapides) sur le réseau
cheops	analyse des réseaux
nessus	audit de réseaux (recherche active de faille)
etherape	statistiques (rapides) sur le réseau
netperf	calcul de vitesse du réseau
netpipe	calcul de vitesse (selon taille des paquets) du réseau

IPv6

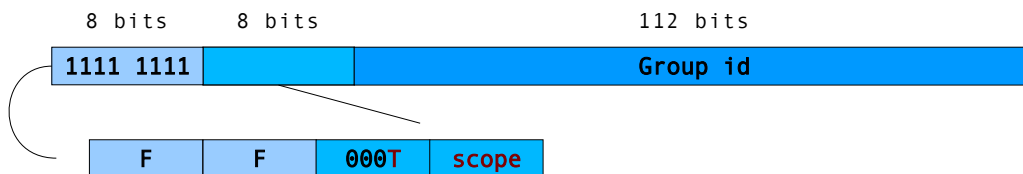
Adressage IPv6

- la taille passe de 32 bits à 128 bits
- découpe des mots de 128 bits en 8 mots de 16 bits, présentés en hexadécimal et séparé par un « : »
- les 0 consécutifs sont abrégés par « :: »
- la notation CIDR est toujours de rigueur : adresse-ipv6/longueur-du-préfixe
- structuration de l'adresse en niveau : FAI, prestataire et site

Type d'adresses :

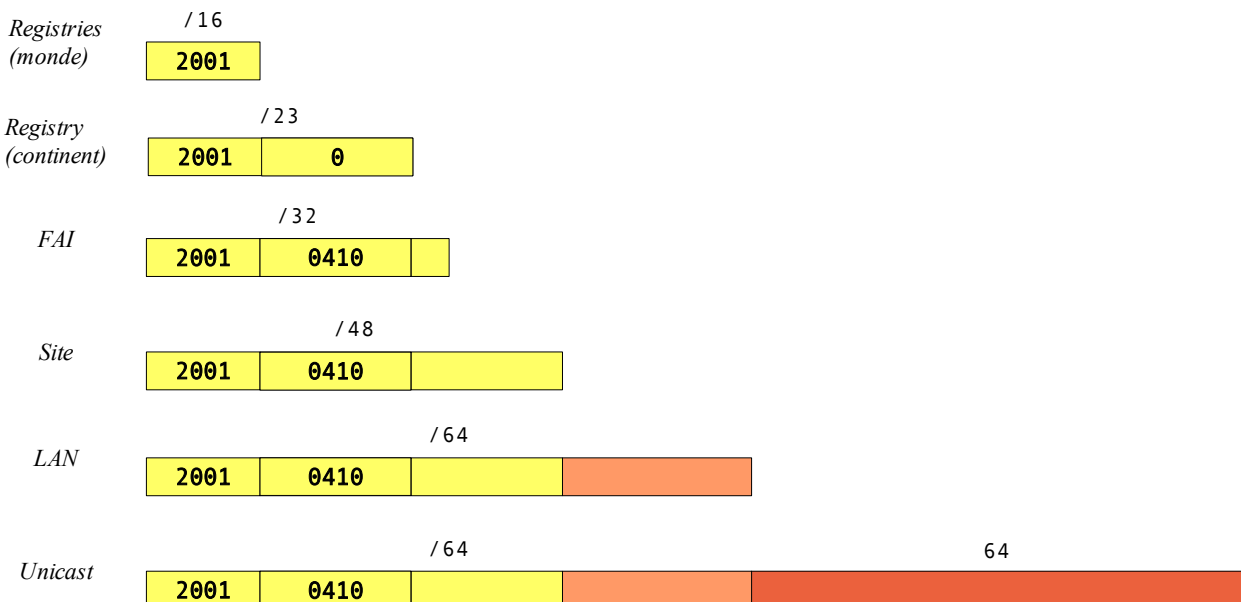
- **unicast** : désigne une interface unique
- **multicast** : groupe d'interfaces (pouvant appartenir à des équipements différents)
 - remplace les adresses de *broadcast* (trop gourmandes)
 - possibilité d'abonnement au groupe *multicast*
 - préfixe FF00::/8
- **anycast** : désigne un groupe d'interfaces, mais transmis à une seule de ces interfaces
 - ces adresses sont dans le même espace d'adressage

Adresse multicast :

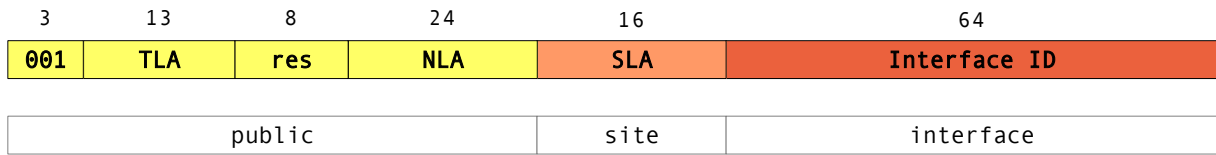


- T pour temporaire (1 pour oui, 0 pour non)
- scope pour la portée de l'adresse

Address allocation policy :



Plan d'adressage agrégé :



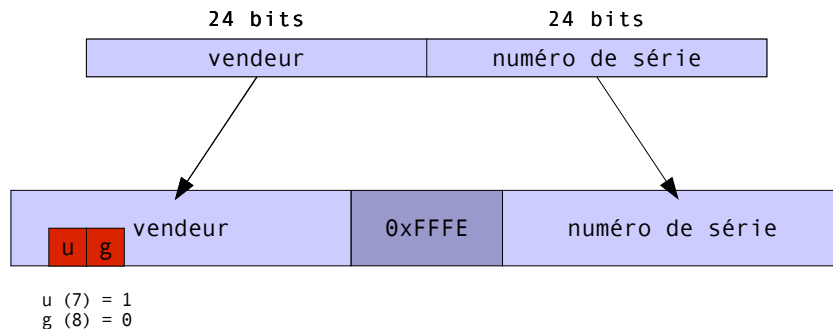
- TLA = *Top Level Agregator*
- NLA = *Next Level Agregator*
- SLA = *Site Level Agregator*
- res = réservé

Autres adresses :

- indéterminée → 0:0:0:0:0:0:0:0 abrégée en ::
- boucle locale → 0:0:0:0:0:0:0:1 abrégée en ::1
- adresse lien-local → concaténation du préfix FE80::/64 à l'identificateur de l'interface
- adresse site-local → concaténation du préfix FEC0::/48, sous-réseau sur 16 bits et 64 bits d'identificateur (interface)

Identificateur d'interface :

- configuration automatique avec l'adresse matérielle (MAC, 48 bits)
- configuration manuelle
- assignation *via* DHCP
- génération automatique
- aléatoire



Dessin 1: Auto-configured IEEE802

Exemple : adresse matérielle 00:A0:24:E3:FA:4B donnera 02A0:24FF:FEE3:FA4B.

Adresse compatible IPv4 :

- *tunneling* IPv6/IPv4



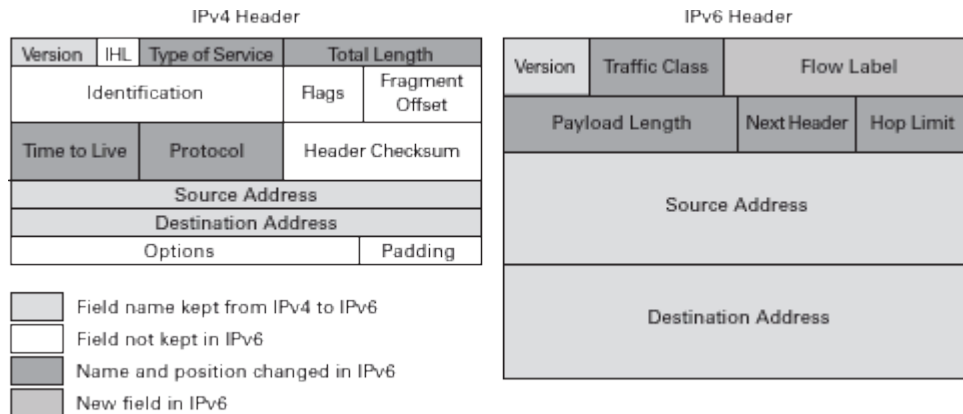
- *IPv4 mapped address*, pour les architecture *dual-stack*



Datagramme IPv6

En-tête :

- l'en-tête ne contient plus de champs checksum → les protocoles de niveau supérieur doivent sans charger
- leur taille est fixe
- champs alignés sur 64 bits
- taille MTU vaut 1280 octets
- plus de fragmentation → les algorithmes de découvertes (Path MTU) doivent empêcher la fragmentation



→ contenu :

- **version** : occupe la même place que pour IPv4, sa valeur vaut 6 (hum)
- **classe de trafic** : permet la différenciation de services (cf. DiffServ)
- **identificateur de flux** : numéro unique choisit par la source, doit permettre un traitement particulier pour le routeur (choix d'une route, temps réel, etc.) → notion de contexte
- **longueur de données** : ne contient **que la taille des données utiles**, donc sans la taille de l'entête
- **entête suivant** : identifie le prochain entête (protocole)
- **nombre de saut** : similaire au TTL

ICMP v6 : évolution de la version 4, meilleure définition

- détection des erreurs (équipement inaccessibles, durée de vie expirée, etc.)
- tests (*ping*)
- configuration automatique des équipements (redirection, découverte)
- gestion des groupes *multicast*
- fonction du protocole ARP

Protocoles et services IPv6

Découverte des voisins : permet à un équipement de s'intégrer à l'environnement

- ne consiste pas à établir une liste complète des équipements, mais seulement ceux qui dialoguent
- la résolution d'adresse est effectuée *via* ICMPv6
 - souplesse d'utilisation
 - construction de tables {@MAC, @IPv6}
 - la requête est appelée « *sollicitation de voisin* » envoyé à l'adresse « *multicast sollicité* »
- détection d'inaccessibilité (*Neighbor Unreachability Detection*)

Autoconfiguration : atout principal d'IPv6, implique des fonctionnalités de la « découverte des voisins »

- découverte des routeurs, *ICMP Router Discovery*
- découverte des préfixes, selon les annonces des routeurs → permet de construire a (les) adresse(s) IPv6
- détection des adresses dupliquées, *Duplicate Address Detection*
- découverte des paramètres : taille MTU, nombre max. de sauts, etc.

Avantages de l'autoconfiguration :

- notion de réseau « *plug-and-play* » (et non *plug-and-pray*)
- la machine obtient toutes les informations nécessaires automatiquement (sans intervention humaine)

→ dans l'idéal, l'utilisateur branche sa machine et pouf, ça marche

Autoconfiguration d'adresse :

- acquisition d'une adresse à la première connexion
- mise à jour si changement de fournisseur d'accès (par exemple)
- implique
 - la création d'une adresse lien-local
 - la vérification de son unicité
 - détermination de l'adresse unicast globale
- autoconfiguration sans état (gestion non stricte) et avec état (gestion stricte)
- le routeur a un rôle important

Découverte de services :

- recherche des ressources disponibles sur le réseau : imprimante, scanner, messagerie, etc.
- **SLP** (*Service Location Protocol*) : envoi d'une requête sur le réseau
- **DNS Service Discovery**

Sécurité : fournit la confidentialité, l'authentification, l'intégrité des données, la protection contre le rejet et le contrôle d'accès

- mécanismes cryptographiques
- *IPSec* (niveau IP) peut être mise en œuvre sur tous les équipements → moyen de protection unique
- *Authentication Header*
- *Encapsulating Security Payload*

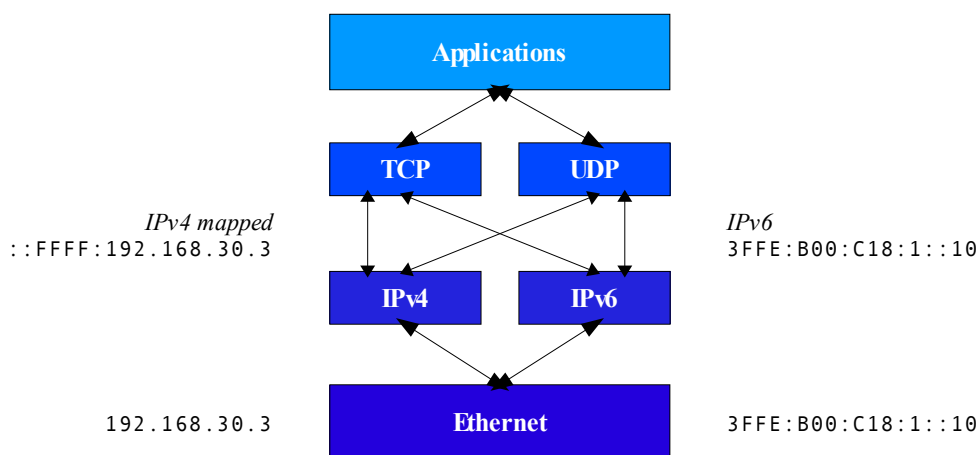
Passage d'IPv4 à Ipv6

Trois grands principes :

- *dual-stack* → deux piles IP coexistent
- *tunneling*
- *translation* → communications entre équipement IPv6 et équipements IPv4

Approche « dual-stack »

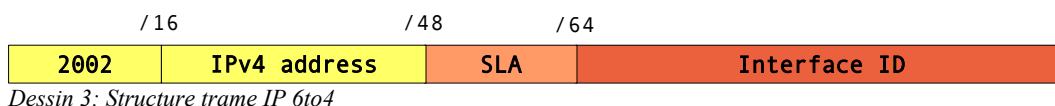
- deux piles IP actives, avec chacune une adresse (?)
- les applications échangent les données avec des clients v4 ou v6
- les applications *full-IPv6* use des adresses v4 mappées



Dessin 2: Approche à double pile IP

Approche « tunneling »

- tunnel Ipv6 via une infrastructure IPv4 → encapsulation des paquets v6 dans des paquets v4
- méthodes
 - tunnel configuré à la mano, avec routeurs *dual-stack*
 - tunnel utilisant des routeur *6to4*
 - tunnel *ISATAP*



Approche « translation »

- **ALG**, *Application Layer Gateway*
- **NAT-PT**, traduit les adresse v6 ↔ v4

Administration avec SNMP

Types d'applications :

- *managers*, gestionnaire, sur les systèmes d'administrations
 - envoi d'ordre, réception des informations, etc.
- *agents*, sur les systèmes administrés
 - contrôle des objets (modem, connexion TCP, table de routage, etc.)

Modèles d'information :

- organisationnel
 - domaine d'administration, répartition des agents/managers
 - système coopératif et distribué
- fonctionnel
 - gestion des erreurs → détecter, isoler, réparer
 - gestion de la configuration
 - gestion (évaluation) des performances
 - gestion des comptes utilisateurs (limitation de l'usage, facturation)
 - gestion de la sécurité (cryptage, authentification, contrôle)
- information
 - **MIB**, *Management Information Base*
 - convention pour la description et l'identification des données

Notation de Syntaxe Abstraite, ASN.1

Type de données

<i>Type</i>	<i>Signification</i>
INTEGER	Entier
BIT STRING	Chaîne de bits
OCTET STRING	Chaîne d'octets
NULL	Aucun type
OBJECT IDENTIFIER	Type de données officiellement défini

Identification des objets :

- chaque objet est situé à un emplacement unique
- un nœud est identifié par un couple d'étiquettes ou le nombre seul
- *exemple* : { iso(1) organisation(3) dod(6) internet(1) admin(2) }

Définition de type

<i>Mot clef</i>	<i>Signification</i>
SEQUENCE	Équivalent à une structure C
SEQUENCE OF	Tableau mono-dimensionnel
CHOICE	Union d'une certaine liste de types

Exemples :

```
compteur INTEGER ::= 100
internet OBJECT IDENTIFIER ::= {iso org(3) dod(6) 1}
status ::= INTEGER { haut(1), bas(2) }
TaillePaquet ::= INTEGER(0..1023)
MonType ::= SEQUENCE {
    index INTEGER,
    adressePhysique OCTET STRING,
}
```

Modèle SNMP

Nœud administré

- entité capable de communiquer des information d'état (hôte, routeur, pont, imprimante, etc.)
- exécute un agent SNMP

Station d'administration

- ordinateur exécutant un logiciel (HP OpenView par exemple)
- communique avec les agents

Protocole d'administration

- communication type question/réponse
- interrogation ou changement de l'état d'un objet

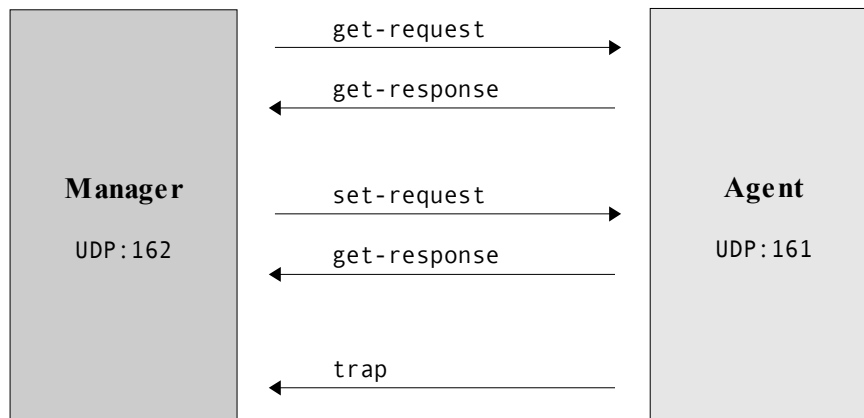
Agent mandataire

- dans le cas où un nœud ne supporte pas SNMP
- équipement ↔ agent mandataire ↔ manager

Protocole SNMP v1

Caractéristiques

- utilise UDP/IP
- simple, rapide, concis → remontée rapide des informations
- cinq types de messages
 - requête `get-request`
 - requête sur le prochain objet `get-next-request`
 - mise à jour `set-request`
 - réponse à une requête `get-response`
 - signal émis par un agent `trap`



Structure SMI

- règles de description de l'information et d'identification de l'objet dans la MIB → chaque objet possède un ident. unique appelé *OID*

Structure de la MIB

- base de données gérée par un agent SNMP
- pré-existence d'une MIB standard
- les objets propres sont dans la MIB privée
- rajout du suffixe .0 à l'OID pour accéder à une valeur d'un objet simple

Protocole SNMP v2

Messages

<i>Message</i>	<i>Signification</i>
get	Demande de la valeur d'une ou plusieurs variables
get-next	Demande de la variable suivante
get-bulk	Chargement d'une grande table
set	Mise à jour d'une ou plusieurs variables
inform	Message d'une description d'une MIB locale
trap	Indication de déroutement provenant d'un agent
response	Réponse

Format de messages

- non sécurisé
- authentifié mais non privé
- privé et authentifié
- context : collection de ressources accessibles par l'entité
- digest : résultat du hachage
- tampon horaire pour la source et pour la destination

Protocole SNMP v3

- sécurité : authentification et cryptage
- administration : nommage des entités, notion de vue, configuration à distance