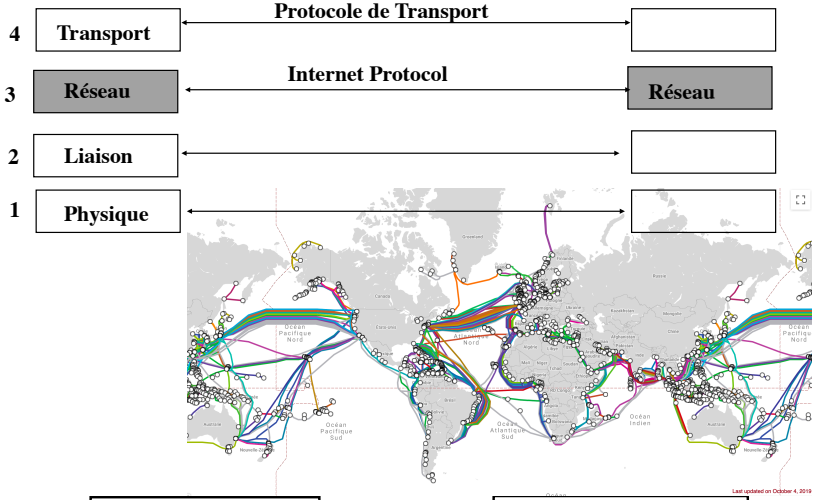
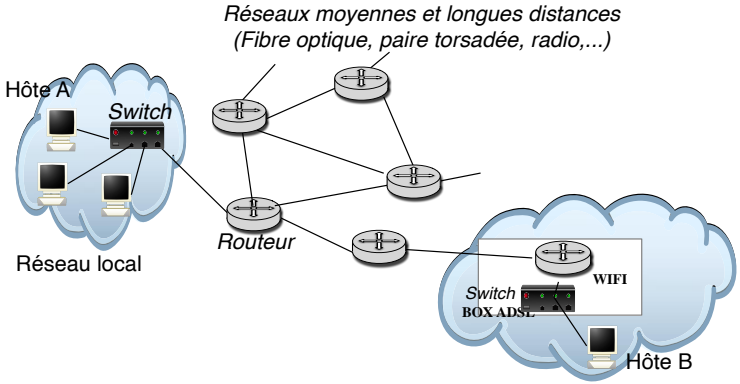
La couche réseau



Fonctionnalités de la couche réseau



- **Le routage: déterminer le chemin (la route) des paquets à travers le réseau**
- **Sous réseaux hétérogènes:**
  - Segmentation peut être nécessaire
- **Le contrôle de congestion: éviter les embouteillages**

Comment trouver une route ?

- Comparable au réseau routier : tronçons de route et carrefours
- Aujourd'hui on utilise le GPS, avant on utilisait une carte

Comment déterminer la route à suivre?

- **Avant le départ on consulte une carte et on décide de notre route au vue de la carte et de l'adresse de notre destination**
- **Autre idée:**
  - On mets une personne à chaque carrefour et on demande notre chemin au fur et à mesure que l'on avance
  - Intérêts
    - Moins de carte à distribuer (une par carrefour)
    - Aucun travail de l'utilisateur, il donne juste l'adresse de sa destination
  - Mais l'utilisateur ne décide pas de sa route
- **Dans le protocole IP la route est décidée dans les routeurs**

Deux politiques différentes possibles pour le routage

- **Le routage concerne la plupart du temps un flux de nombreux paquets (échange de fichier, pages web...) entre deux machines**
- **Deux façons de décider de la route à suivre**
  1. Faites pour chaque paquet (mode Datagramme)
  2. Faites pour l'ensemble des paquets d'un flux
     - On décide de la route avant l'envoi des paquets de donnée (établissement d'un ***circuit virtuel***)
     - Tous les paquets appartenant au même flux vont suivre ce circuit virtuel
- **La deuxième solution semble plus économique en terme de calcul**
  - On peut aussi plus facilement garantir des QoS pour un flux donné

Mode *Datagramme*

- Chaque paquet est «routé» indépendamment des autres
- La décision du routage doit être prise pour chaque paquet
- Des paquets successifs peuvent donc suivre des routes différentes et il peut y avoir alors déséquencement des paquets
- **Intérêts**:
  - Temps d'acheminement plus rapide
  - Défaillance d'un routeur: pertes seulement de quelques paquets, possibilité de réagir à une panne plus rapidement
- **Inconvénients**:
  - Calcul du routage à chaque paquet
  - Qualité de services difficile à garantir et prévention des congestions difficile
- **Le protocole de routage d'Internet (IP) fonctionne en mode *Datagramme***

Retour sur les adresses IP

- **IPv4: s**ur 4 octets donnés en décimal séparés par des points (Ex: 201.123.45.10)
- **Deux parties distinctes:**
  - Une partie désigne le réseau (sous-réseau)
  - Une partie désigne la machine dans le réseau
- **Différentes classes suivant les besoins (définies par les 4 premiers bits du 1er octet):**
  - classe A: (0.....) ; partie machine: 3 derniers octets (R.M.M.M)
  - classe B: (10.....) ; partie machine: 2 derniers octets (R.R.M.M)
  - classe C: (110.....) ; partie machine: le dernier octet (R.R.R.M)
  - Classe D: (1110.....) Multicast (G.G.G.G)
  - **Exemple**:
    - 195.0.0.4 et 195.0.0.5 : deux adresses de classe C de deux machines appartenant au même réseau 195.0.0.0
    - On désigne un réseau en mettant la partie machine à 0
  - Partie machine à 0 interdit pour ne pas confondre une adresse de réseau et une adresse de machine

Les adresses IP

- **Exemple:**
  - adresse 160.1.2.3 : 160= 10100000 donc classe B
  - Réseau 160.1.0.0 ; adresse de machines possibles: 160.1.0.1 à 160.1.255.254
- **Adresses particulières:**
  - 127: boucle locale (*Loopback*); interface virtuelle
  - 0.0.0.0 : Utilisé dans protocole d'apprentissage d'adresse
  - *Broadcast*: Pour s'adresser à toutes les machines d'un sous-réseau
    - Partie machine ne comporte que des 1 (en binaire)
      - Exemple: 192.0.0.255
    - 255.255.255.255
    - Broadcast Ethernet : FF:FF:FF:FF:FF:FF (en Héxadécimal)

Les adresses IP sans classe

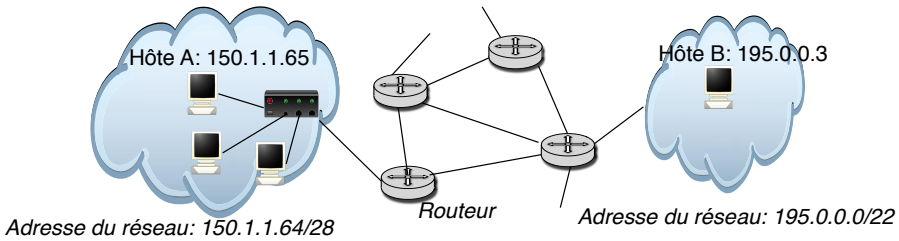
- **Depuis le début des années 1990 :**
  - Sans classe (*classless*) pour économiser des adresses
  - Adresse réseau / Nombre de bit de la partie réseau
  - Permet de limiter le gaspillage des adresses
  - Exemple: 192.0.0.64 / 26, les deux bits de poids fort du dernier octet font aussi partie de la partie réseau
  - On parle de **PREFIXE** pour désigner la partie réseau
- **IPV6**
  - Adresses sur 16 octets données en Héxadécimal: 2001:660:5301:8000:0:0:0:1aed/64
  - 6,67. 10^{17} adresses au millimètre carré de la surface terrestre !
  - Même principe de préfixe que IPv4
  - Saturation des adresses IPv4 (2012)
  - Co-habitation IPV6-IPV4 à l'aide de tunnels

Le masque de réseau (*Netmask*)

- **Comment calculer la partie réseau d'une adresse IP de machine ?**
- **Netmask**
  - Composé de bits à 1 sur la partie réseau et de bit à 0 sur la partie machine
  - Exemples de ***netmask***:
    - **Classe A (/8): 255.0.0.0; Classe C (/24): 255.255.255.0**
    - **Réseau/ 26 : 255.255.255.192; Réseau /28: 255.255.255.240**
  - Pour calculer la partie réseau (donc l'adresse du réseau) à partir d'une adresse de machine il suffit de faire le ET bit à bit avec le Netmask
    - **Exemple:**
      - **160.1.64.69/27, Netmask= 255.255.255.224**
      - **1010 0000. 0000 0001. 0100 0000. 0100 0101**
      - ***ET* 1111 1111. 1111 1111. 1111 1111. 1110 0000**
      - **1010 0000. 0000 0001. 0100 0000. 0100 0000= 160.1.64.64**
- **On peut aussi calculer l'adresse Broadcast à l'aide du Netmask**
  - Adresse Réseau ***Ou*** Complément du Netmask

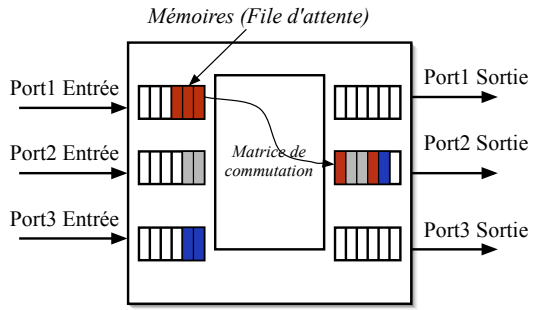
Principe du routage dans IP

- Chaque réseau («sous-réseau») possède une adresse IP permettant de l'identifier de manière unique
- Les routeurs sont connectés à plusieurs réseaux
- Ils possèdent donc une adresse IP par réseau connecté



Principe du routage dans IP

- Chaque paquet reçu par un routeur est mémorisé dans une file d'attente
- L'entête IP est analysé et le paquet re émis vers la destination
- L'adresse IP de la machine destination figure dans l'entête du protocole IP. C'est grâce à elle que les paquets vont pouvoir être «routés»
- L'adresse IP source figure aussi pour que le récepteur puisse répondre ou pour avertir l'émetteur lors de problème lors de l'acheminement



Le routage principe

- Pour chaque paquet reçu un routeur doit être capable de décider de la direction du paquet
- **Le routage :**
  - Au vue de l'adresse IP de la machine destination, comment décider de la route à prendre ?

  c'est à dire dans un routeur (ou un hôte) comment répondre à la question : ***quel est le prochain routeur à qui envoyer le paquet (sous entendu sur quelle interface re-émettre le paquet)?***

- **Deux fonctions distinctes**
  1. Décider à l'aide d'informations locales ***(table de routage***) et de l'adresse destination du paquet quel est le prochain routeur à qui envoyer le paquet (et donc sur quel réseau le re-émettre)
  2. Construire la table de routage : « à la main » ou à l'aide ***d'algorithme de routage***

Le routage dans Internet

- L'adresse IP destination se trouve dans l'entête IP
- Le routage se fait en fonction de la partie réseau de l'adresse IP
  - Nécessité de calculer la partie réseau d'où l'intérêt du Netmask
- Une fois arrivé sur le réseau destination, l'adresse de la machine destination est trouvée à l'aide d'un protocole de résolution d'adresse locale (par exemple ARP (Address Resolution Protocol) sur un réseau Ethernet
- Il n'y a pas de correspondance adresse-emplacement géographique jusqu'en 93
  - Ne simplifie pas le "calcul des routes"
- Depuis des efforts sont faits pour simplifier le routage

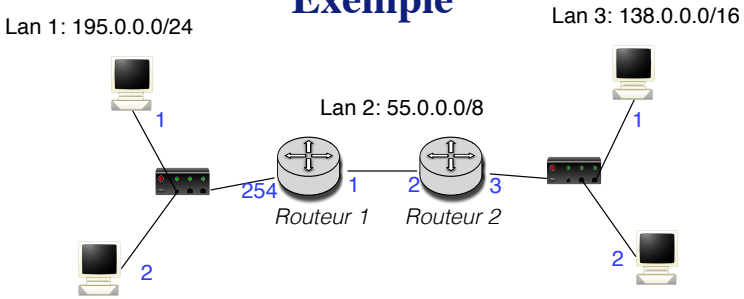
Le routage dans Internet

- **La décision dans IP du routage:**
  - Table de routage:
    - Adresse destination (partie réseau), netmask, adresse routeur voisin
  - Consultation de la table de routage par IP à l'arrivée d'un paquet:
    - Pour chaque ligne de la table de routage (***Adr, netmask, AdrRouteur***) faire

      ★Si (***adresse destination du paquet*** AND ***netmask***) = ***Adr*** alors
      - envoyer le paquet au routeur voisin d'adresse ***AdrRouteur***
      - Pour cela faire appel à ARP pour connaître son adresse Ethernet

      ★Sinon passer à la ligne suivante
    - Si l'adresse n'est pas dans la table alors
      - renvoyer un paquet ICMP: "*destination inaccessible*" à la machine source ou afficher un message «*no route to host*» si on est sur la machine source

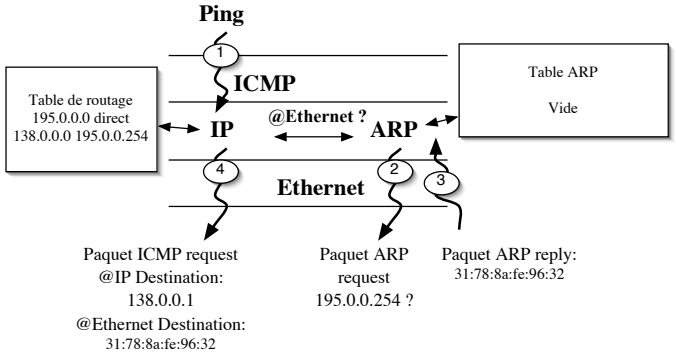
Exemple



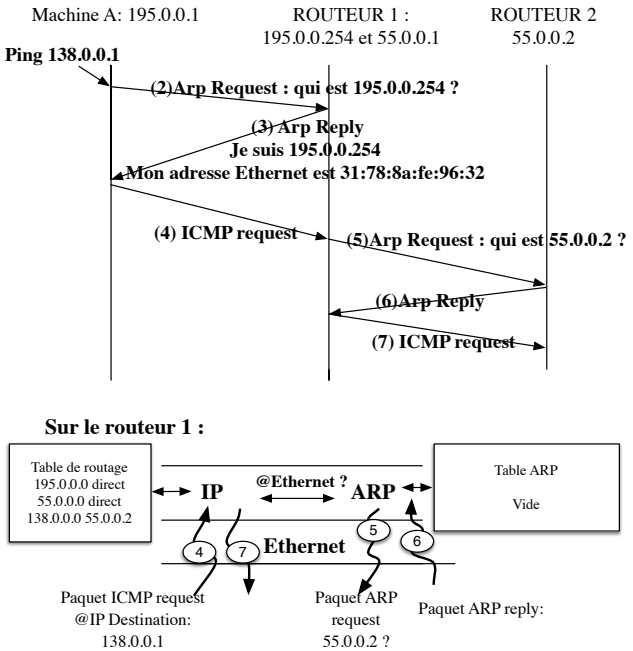
- **Tables de routage :**
  - Routeur 1:
    - 195.0.0.0 255.255.255.0 direct
    - 55.0.0.0 255.0.0.0 direct
    - 138.0.0.0 255.255.0.0 55.0.0.2
  - Machine d'adresse 195.0.0.1:
    - 195.0.0.0 255.255.255.0 direct
    - 55.0.0.0 255.0.0.0 195.0.0.254
    - 138.0.0.0 255.255.0.0 195.0.0.254
- Que se passe t-il lors d'un ping de 195.0.0.1 vers 138.0.0.1 ?

Dialogue IP /ARP dans les hôtes et les routeurs

- **Sur la machine source: au départ du paquet ICMP request (issu du ping)**



- La table ARP contient après le ARP reply : 195.0.0.254 31:78:8a:fe:96:32



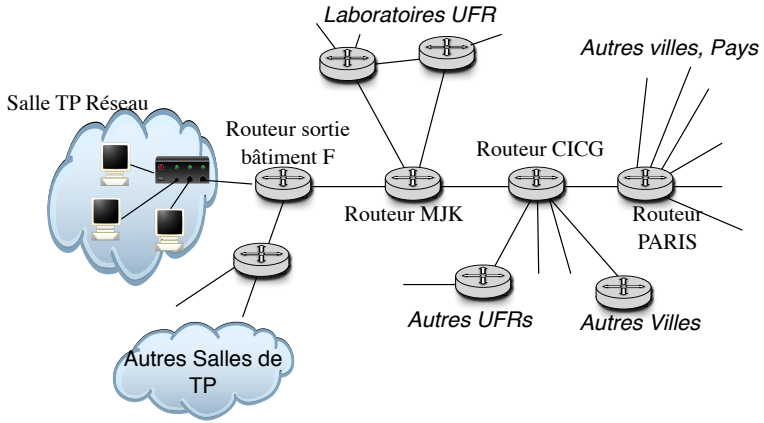
Internet: un routage hiérarchique

- **Problème : nombre de réseau aujourd'hui très grand**
  - table de routage trop importante
- **Solution:**
  - Regroupement par zone des routeurs
  - Malheureusement l'adresse Internet ne "donne" pas la zone
- **Chaque table de routage contient le moyen d'accéder**
  - aux routeurs et machines de sa zone
  - à au moins un routeur de niveau supérieur (ligne de la table de routage particulière « ***default*** »)
  - Les routeurs de plus haut niveau possèdent des tables de routage quasi-complète
- Attention les lignes "par défaut" implique un choix statique qui est fait parfois au détriment du choix du chemin optimal

Routage par *défaut*

- Dans les machines utilisateurs une seule ligne (hormis la ligne d'accès direct) peut suffire
- La table de routage contient une ligne par défaut :
  - Adresse Réseau Netmask Routeur
  - 0.0.0.0 0.0.0.0 192.0.0.1
  - Lors du calcul X.X.X.X and 0.0.0.0 = 0.0.0.0 quel que soit X.X.X.X
- Tous les paquets seront envoyés vers le routeur par ***défaut*** (d'adresse 192.0.0.1 dans l'exemple)
- Remarque : les serveurs **DHCP** envoie l'adresse des routeurs (« passerelle ») par ***défaut***

Un exemple concret



- Table de routage de plus en plus grandes
- Exemple: Routeur de sortie de l'UFR (MJK) : contient tous les réseaux appartenant à l'UFR IM2AG et une ligne par défaut vers un routeur du CICG
- Essayer ***traceroute***

Agrégation d'adresse

- On peut parfois diminuer le nombre de ligne d'une table de routage
  - Exemple:

Adresse Réseau	Netmask	Routeur	
200.0.0.0	255.255.255.0	192.0.0.1	/24
200.0.1.0	255.255.255.0	192.0.0.1	
200.0.2.0	255.255.255.0	192.0.0.1	
200.0.3.0	255.255.255.0	192.0.0.1	

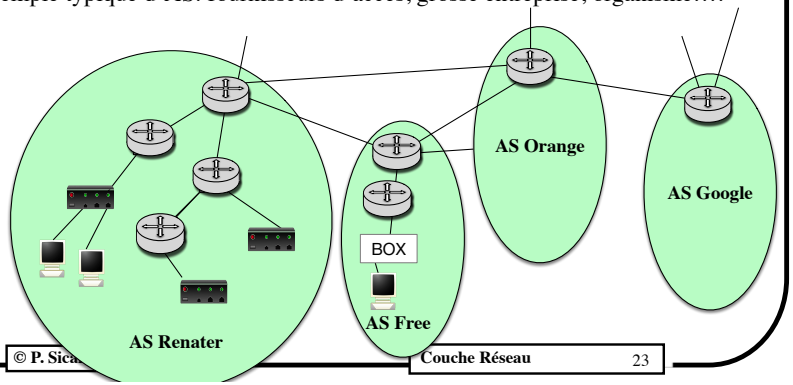
  - **Est équivalent à:**

Adresse Réseau	Netmask	Routeur	
200.0.**0**.0	255.255.**252**.0	192.0.0.1	/22

- **On comprend l'intérêt de bien attribuer les adresses suivant le lieu: l'agrégation d'adresse peut être très importante**

Organisation d'Internet

- **Système autonome (AS):**
  - Géré par une autorité administrative
  - Décide des protocoles de routage utilisés : domaine de routage
  - Attribue les adresses IP
  - Un AS est identifié par un numéro sur 32 bits (ASN)
  - Aujourd'hui plus de 90 000 AS dans le monde
  - Exemple typique d'AS: fournisseurs d'accès, grosse entreprise, organisme….



Les algorithmes de routage

- **Comment remplir les tables de routages ?**
- **Centralisés ou décentralisés ?**
  - Centralisés : les chemins sont calculés par un nœud particulier (irréaliste à l'échelle d'Internet)
  - ***Décentralisé*** : chaque nœud calcule les chemins
- **Statique ou dynamique ?**
  - Statique : les chemins sont fixes. Il faut les changer "à la main" en cas de modifications du réseau
  - ***Dynamique*** : le choix des chemins s'adapte plus ou moins rapidement à des pannes réseaux et machines. L'adaptation en temps réel à la charge des routeurs est très délicate et non implémentée

Critères de choix des routes

- **Métrique**
  - La plupart des algorithmes associent un coût à un chemin (on parle de métrique).
  - Le but est de minimiser ce coût pour chaque route
  - Ce coût peut faire intervenir plus où moins de paramètres:
    - Le plus simple : nombre de réseaux traversés
    - Temps moyen de traversé d'un routeur, débit moyen, charge moyenne (mais cela reste des valeurs statiques)
- **Autres informations prises en compte**
  - Critère politique, de sécurité, interdiction …
  - Par exemple interdiction de passer par certaines routes pour atteindre une destination

Classification suivant la connaissance de la topographie du réseau

- Pour calculer les routes et remplir la table de routage un routeur doit avoir connaissance de la topographie du réseau
- On distingue deux types d'algorithme
  - Connaissance partielle du réseau
    - Algorithme à "**vecteurs de distance**"
    - Le routeur connait seulement la distance à laquelle il est de chaque réseau
    - Très simple mais possibilité d'incohérence
  - Connaissance de la topographie exacte du réseau
    - Algorithme "d'**état de lien**"
    - Beaucoup plus couteux

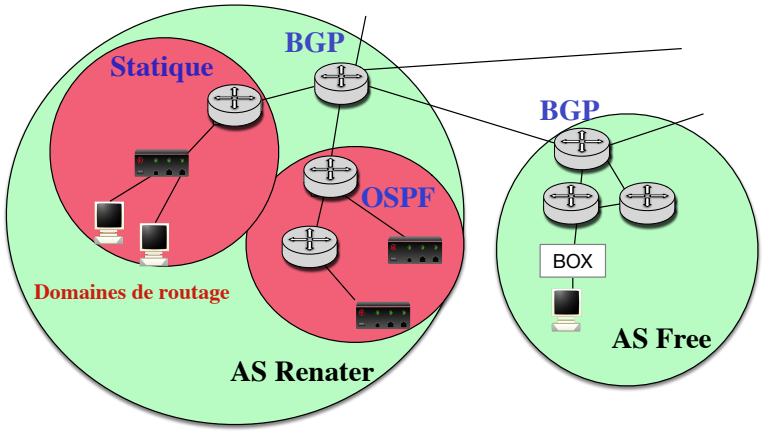
Protocoles de routage associés à IP

- **Dans les systèmes autonomes : protocole interne (IGP Interior Gateway Protocol)**
  - **Routeurs/hôtes:**
    - Hôte : souvent statique une route par défaut distribué par DHCP
    - Routeurs: Topologie variable, interêt des algorithmes Dynamiques
    - Type **"vecteur de distance"** :
      - Vision partielle du réseau, "léger" mais peut poser des problèmes
      - Protocole **RIP** (Routing Information Protocol)
  - **Inter-routeurs:**
    - Topologie plutôt fixe mais plus d'adresses à connaître
    - Métrique: Débit, charge, temps de traversée ...
    - On peut utiliser les protocoles **RIP**, IGRP (Cisco)
    - Mais on préfère des protocoles à "Information globale" :
      - Type **"état de lien"**, chaque routeur connait l'ensemble du réseau et calcul les plus courts chemins de façon exacte
      - Protocole **OSPF** (Open Shortest Path First)
      - Plus "stable" mais plus coûteux (information échangée et calculs)

Protocoles de routage associés à IP

- **Inter domaine: protocole externe (EGP Exterior Gateway Protocol)**
  - Topologie fixe
  - Nombre d'adresses à gérer très important
  - En relation avec les protocoles de routage utilisés dans les AS (exportation/importation)
  - Routeurs internationaux donc critère aussi politique :
    - Par exemple: Interdiction de passer par un AS pour aller à telle destination
  - Protocole **BGP** (Border Gateway Protocol)

Exemple d'AS et algorithmes de routage



Un exemple de protocole de routage Le protocole RIP (Routing Information Protocol)

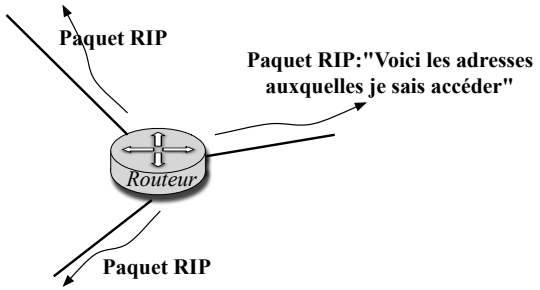
- Type «vecteur de distance»: les routeurs s'échangent les distances aux réseaux qu'ils connaissent
- Un des premiers protocoles utilisés dans Internet pour le routage interne au domaine, toujours très utilisé (version 2)
- Utilisé de routeur à routeur et de routeur à machine hôte
- **Dynamique**
  - Adaptation aux modifications du réseau
- **Distribué et vision partielle:**
  - les routeurs n'ont qu'une vision partielle de la topographie du réseau
  - Basé sur l'échange entre routeurs voisins du contenu de leur table de routage et des métriques associées

Fonctionnement de RIP

- **Association d'un coût aux lignes de la table de routage:**
  - Le coût est égal au nombre de réseaux traversés pour arriver à destination
  - RIP détermine le chemin à mettre dans la table de routage en fonction de ce coût
  - Pas de mémorisation de chemins multiples
- **Echange des informations entre les routeurs:**
  - Application particulière (démon) qui utilise le protocole UDP (*User Datagram Protocol* de la couche transport) et modifie la table de routage en fonction des informations reçus des routeurs voisins
  - Un paquet RIP contient une liste (Adresse réseau, coût)
  - Les paquets sont émis en broadcast

Principe général de RIP

- Périodiquement, les routeurs envoie sur les réseaux auxquels ils sont connectés leur connaissance actuelle du réseau (contenu de la table de routage + coûts associés)
- A la réception de ces paquets RIP, les routeurs et les hôtes complètent leurs tables de routage
- La "connaissance du réseau" va se propager ainsi de routeur en routeur



Fonctionnement de RIP

- Pour les équipements terminaux seule la partie réception de l'algorithme est effectuée
- A l'initialisation des routeurs les tables de routage sont initialisées avec l'ensemble des adresses des réseaux auxquels le routeur est directement connecté
- Le coût minimum (1) est alors associé à ces adresses destinations
- **Exemple**:

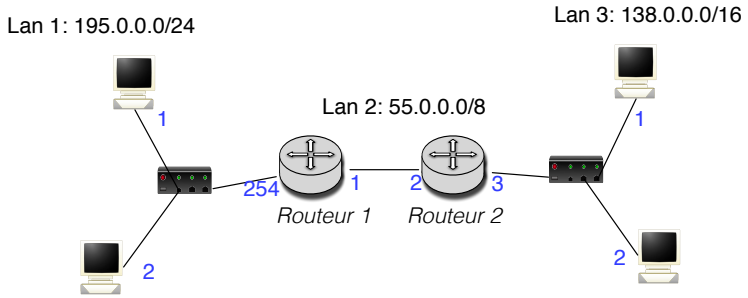


Table de routage

- **Initiales (après la configuration des interfaces):**
  - Routeur1:
    - 195.0.0.0 255.255.255.0 direct 1
    - 55.0.0.0 255.0.0.0 direct 1
  - Routeur2:
    - 55.0.0.0 255.0.0.0 direct 1
    - 138.0.0.0 255.255.0.0 direct 1
  - Machine 195.0.0.1:
    - 195.0.0.0 255.255.255.0 direct 1

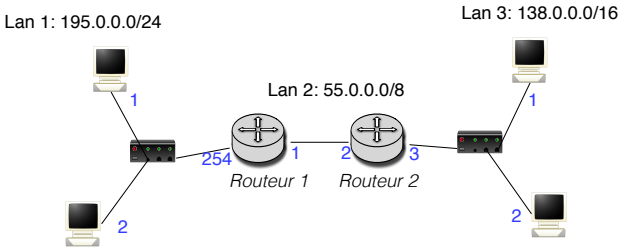
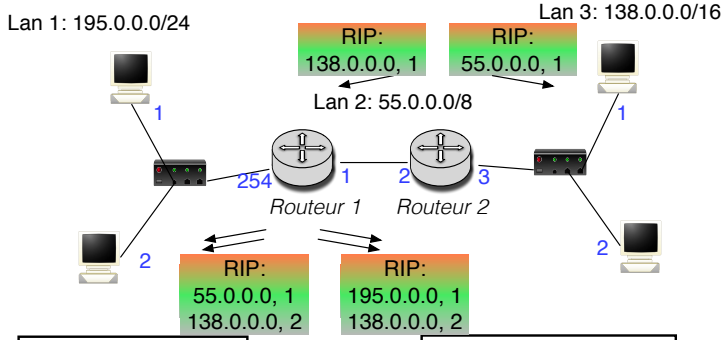


Table de routage

- **Paquets RIP:**
  - Envoyés par le routeur1 sur LAN1: (55.0.0.0, 1)
    - **La machine 195.0.0.1 va ajouter (55.0.0.0, 195.0.0.254, 2) dans sa table de routage**
  - Envoyés par le routeur1 sur LAN2: (195.0.0.0, 1)
    - **Le routeur 2 va ajouter (195.0.0.0, 55.0.0.1, 2) dans sa table de routage**
  - ....



Algorithme de RIP

- Chaque routeur envoie à tous ses voisins périodiquement (30 s) la liste (***@ réseau, coût***) qu'il connaît d'après sa table de routage (voir limitation plus loin)
- A la réception d'un paquet RIP arrivant d'un routeur voisin d'adresse ***@routeur*** pour chaque destination (***@destination, cout***) contenue dans la paquet faire
  - **Si** @destination inconnue **alors**
    - rajouter dans la table de routage (***@destination, @routeur, cout +1***)
    - armer timer
  - **Si** @destination connue **alors**

    (apparait dans la table (***@destination, @routeur_voisin, cout_présent***))
    - **Si @routeur_voisin = @routeur alors**
      - changer table (***@destination, @routeur, cout + 1***)
      - relancer timer
    - **Si @routeur_voisin ≠ @routeur et cout+ 1 < cout_présent alors**
      - changer table (***@destination, @routeur, cout + 1***)
      - relancer timer
- **Si** sonnerie d'un timer (2mn 30) **alors** supprimer de la table la destination correspondante

Convergence de RIP

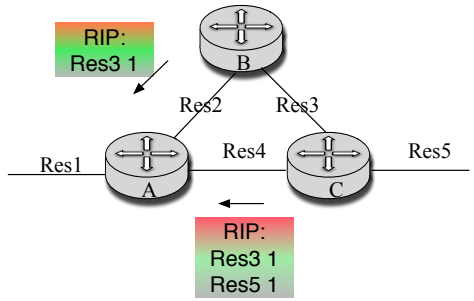
- **Pour arriver à un état stable il faut que les informations se propagent de proche en proche**
- **Le temps de stabilisation dépend de la largeur du réseau:**
  - ~ largeur * durée du timer (30 secondes)

Pour diminuer le temps de la phase d'«apprentissage»:

- Au lancement d'un démon il envoie tout de suite un paquet ***RIP Request*** afin que les routeurs voisins lui envoient immédiatement leurs tables de routage
- En cas d'ajout d'une nouvelle adresse à la table, RIP émet un paquet comportant cette modification immédiatement (il n'attend pas la sonnerie du timer d'émission), on parle de « **Mise à jour déclenchée » (triggered update)**

Exemple

On utilise des noms au lieu des adresses IP



- **Tables de routage (Adresse réseau, adresse routeur, coût)**

On suppose les tables de routage de A, B et C entièrement remplies
Exemple: Table du routeur A:

- Res1, direct, 1
- Res2, direct, 1
- Res4, direct, 1
- Res3, B2, 2 (autre solution : Res3, C4, 2)
- Res5, C4, 2

Différentes solutions sont possibles suivant l'ordre de réception des paquets RIP mais ne sont pas mémorisées

Adaptation dynamique de RIP

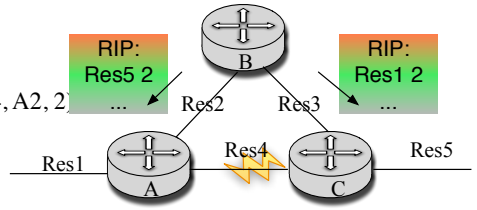
- **Adaptation dynamique aux modifications du réseau:**
  - Ajout/suppression d'un réseau ou d'un routeur
  - Panne d'une machine ou coupure d'un lien

  ➡ Changement dynamique des tables de routage

  - **Suppression de ligne grâce au timer associé à chaque ligne de la table de routage**
  - **Changement des chemins suivant les nouveaux coûts**
  - Attention la convergence n'est pas instantanée

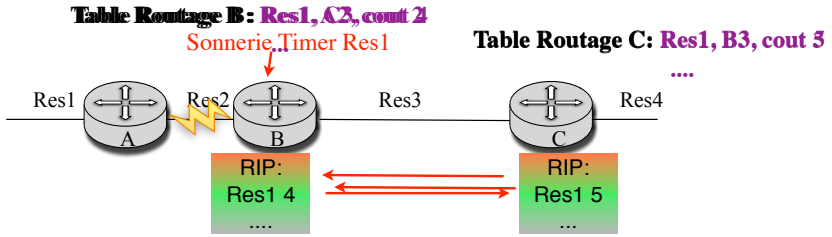
Adaptation dynamique de RIP

- **A :**
  - Res1, direct, 1
  - Res2, direct, 1
  - Res4, direct, 1
  - Res3, B2, 2 (autre solution : Res3, C4, 2)
  - ~~Res5, C4, 2~~ (sonnerie timer)
    Res5, B2, 3
- **B :**
  - Res2, direct, 1
  - Res3, direct, 1
  - Res1, A2, 2
  - Res5, C3, 2
  - Res4, C3, 2 (autre solution : Res4, A2, 2)
- **C :**
  - Res3, direct, 1
  - Res4, direct, 1
  - Res5, direct, 1
  - Res2, B3, 2 (autre solution R2, A4, 2)
  - ~~Res1, A4, 2~~ (sonnerie timer)
    Res1, B3, 3
- Que se passe t-il si le lien entre A et C est coupé ?



Problème du comptage à l'infini

- **L'algorithme de RIP tel qu'il a été donné auparavant peut être mis en défaut**
- Exemple:
  Que se passe-t-il sur le réseau suivant dans le cas où la ligne de A vers B est coupée ?

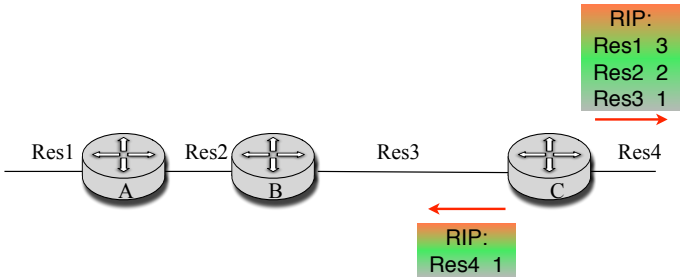


Solution de l"horizon coupé" (split horizon):

Les paquets RIP ne contiennent pas toute la table de routage. Ils ne contiennent que les adresses qui n'ont pas été apprises par la ligne sur laquelle ils sont émis

Exemple horizon coupé

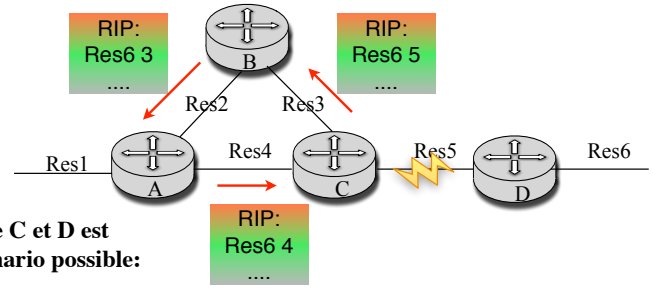
- **Les paquets RIP n'ont donc pas le même contenu suivant les réseaux sur lesquels ils sont émis**
- Exemple**:**



- **Remarque:** Sous Free BSD (en TP) ces adresses apparaissent mais avec une métrique de 16 (équivalent à inaccessible), ce qui revient au même.

Problème du comptage à l'infini (2)

- L'algorithme peut toujours être mis en défaut dans le cas de boucle dans le réseau
- **Exemple :**



Si la ligne entre C et D est coupée, un scénario possible:

- C élimine de sa table de routage Res6 (plus de paquet RIP provenant de D)
- Puis A élimine de sa table de routage Res6 car C ne lui envoie plus rien sur Res6
- B n'a pas encore éliminer Res6 de sa table de routage (timer non synchronisé)
- B envoie donc à A qu'il peut accéder à Res6 avec un coût de 3
- Ensuite A va donc envoyer à C qu'il peut accéder à Res6 avec un coût de 4
- Et ainsi de suite, on tourne en rond et à chaque paquet RIP le coût augmente de 1

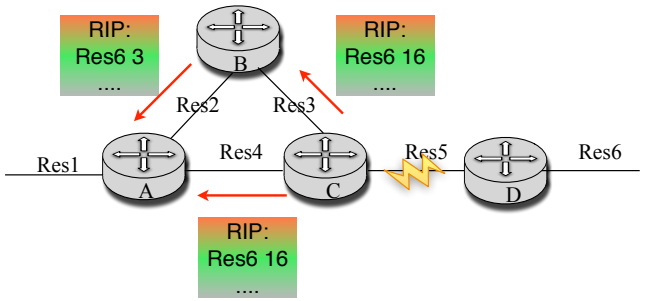
Solutions au comptage à l'infini

- Le problème survient dès que le réseau possède des boucles. Pour remédier complètement au problème il faudrait avoir une vision globale du réseau
- Le fait de limiter l'infini à un entier relativement petit limite les dégâts. Mais il doit être supérieure au nombre maximum de saut dans le réseau complet (RIP : 16)
- Pour éviter que les paquets tournent trop longtemps en rond, champ durée de vie de l'entête IP (TTL)

Limitation du problème du comptage à l'infini

- **Empoisonnement de route** (Route poisoning): A la sonnerie d'un timer le routeur ne supprime pas la ligne immédiatement mais lui associe un coût de 16 (inaccessible) qui sera propagé dans les prochains paquets RIP
  - Cette information est ainsi propagée plus rapidement mais n'évite pas le problème du "comptage à l'infini" dans tous les cas

Exemple d'empoisonnement de route



Si la ligne entre C et D est coupée:

- C élimine de sa table de routage Res6 (plus de paquet RIP provenant de D)
- Il avertit donc tout de suite ses voisins que Res6 est inaccessible (métrique à 16)
- A et B vont donc noter tout de suite que Res6 est inaccessible via C
- Cela n'empêche pas que B est eu le temps de donner à A une fausse information sur Res6 mais le risque est plus petit que sans empoisonnement

Variante de l'horizon coupé

- Certaines versions de RIP implémente une variante de l'horizon coupé appelé « horizon coupé avec empoisonnement inverse » (split horizon with poison reverse).
- **C'est le cas sur Free BSD utilisé en TP.**
- Au lieu de ne pas annoncer une destination apprise depuis un réseau, cette destination est annoncée avec une métrique de 16.
- Cela améliore la rapidité de la convergence après des modifications de la topologie dans des cas particuliers

RIP Version 2

- Améliore la première version de RIP
- Problème des adresses sans classe: il faut envoyer aussi les **Netmasks**
- On peut définir des domaines de routage : un champ domaine permet à un routeur de ne prendre en compte que les paquets RIP propre à ce domaine
- Champ routeur destination permet de spécifier un autre routeur que celui à l'origine du paquet RIP. Intéressant pour ajouter des contraintes de routes dans certaines configurations
- Pour limiter les diffusions, les paquets RIP sont émis sur l'adresse de groupe 224.0.0.9 (utilisation du protocole de gestion du multicast IGMP)
- Pour des raisons de sécurité, une authentification sommaire est ajouté aux paquets RIP:
  - Mot de passe en clair contenu dans les messages
  - MD5 :
    - mot de passe secret connu de l'ensemble des routeurs,
    - *empreinte* calculée à partir du mot de passe et de contenu du paquet (comme un CRC ou un Checksum),
    - envoie du résultat de ce calcul (16 octets) dans le paquet