Routage

(Z:\Polys\Internet_gestion_reseau\4.RIP.fm- 15 septembre 2008 09:33)

PLAN

- Introduction
- Le Distance Vector
- Quelques problèmes
- Des solutions
- Le protocole RIP
- Conclusion

Bibliographie :

- . C. Huitema, Le routage dans l'Internet, Eyrolles, 1995

1. Introduction

1.1. Présentation

Le Routage est composée de 2 fonctions essentielles :

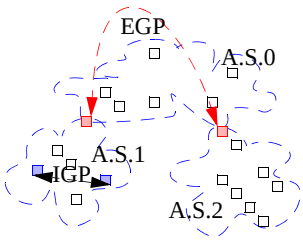
- L'acheminement ("datagram forwarding"),
- La mise à jour des tables de routage

Acheminement :

- réception d'un datagramme
- consultation de la table de routage qui indique le meilleur chemin
- retransmission du datagramme

Mise à jour des table de routage

- base de données répartie des routes
- protocole de mise à jour des tables de routage
- plusieurs classes de protocoles existent :
  - . Distance vector algorithm
  - . Link state algorithm
- domaines d'application de l'algorithme :
  - . domaine interne ("Autonomous System")
  - . domaine externe : interconnexion d'A.S.



1.2. Les deux classes de protocoles de routage

"Distance vector algorithm" :

- algorithme simple,
- par diffusion d'un extrait des meilleurs chemins,
- (sous la forme d'un vecteur où chaque entrée contient une distance)
- entre voisins directs (de proche en proche)
- métrique simple : *hop count*.

"Link state algorithm" :

- 2 phases :
  . diffusion à tous de la connaissance sur les liaisons locales
  . calcul local par chacun des meilleurs chemins sur les informations ainsi rassemblées
- exemple : *Shortest Path First*

=> Distance Vector

2. L'algorithme "Distance Vector"

2.1. Historique

Algorithme (+ protocole) :
- vecteur de distance ("distance vector algorithm")
- algorithme de calcul du plus court chemin
  . décrit par [Bellman - 1957]
  . amélioré par [Bellman & Ford]
- algorithme réparti [Ford & Fulkerson - 1962]

Implémentation :
- première apparition : RIP du réseau XNS de Xérox
- RIP-1 : RFC 1058 - juin 1988.
- RIP-2 : RFC 1388 - juin 1993.

2.2. Principes

Chaque routeur maintient localement une base des meilleures routes (BdR)

=> une entrée de la BdR<@ de destination, distance, @ du prochain routeur>

distance = nombre de noeuds intermédiaires (“hop count”)

Diffusion

- Chaque routeur actif diffuse un message de routage (MdR) :
- Un extrait de sa base de routage
- Périodiquement (30 s)
- A tous ses voisins immédiats
- MdR = une liste de couples <@ de destination, distance>

Réception

- les routeurs mettent à jour leur base de routage, si ce qu’ils recoivent est “meilleur”
- L’adresse du prochain routeur est implicitement celui de l’émetteur du message de routage

Table de routage

- La table de routage (TdR) est produite à partir des infos de la BdR

2.3. Algorithme de mise à jour de la BdR

Lorsqu’un routeur reçoit un message de routage,

chaque couple du MdR est comparé aux entrées de la base de routage (BdR) :

. [1] l'entrée n'existe pas dans la BdR et la métrique reçue n'est pas infinie :

- une nouvelle entrée est crée : prochain routeur = routeur d'où provient le MdR; distance = distance reçue + 1.

. [2] l'entrée existe et sa métrique est supérieure à celle du MdR :

- on met à jour l'entrée : prochain routeur = routeur d'où provient le MdR; distance = distance reçue + 1.

. [3] l'entrée existe et son prochain routeur est celui d'où provient le MdR :

- distance = distance reçue + 1 (augmentation ou diminution de la distance !).

. [4] sinon rien.

Etat initial :

Chaque routeur connaît son environnement immédiat :

. les adresses de ses interfaces,

. les (sous-)réseaux IP auxquels il est connecté directement : distance = 1.

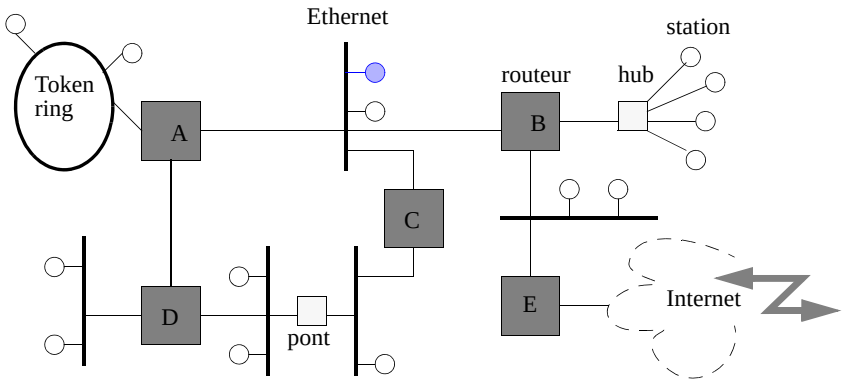
Etat des noeuds :

. Actif (les routeurs diffusent leurs routes),

. Passif (les stations d'extrémité écoutent).

2.4. Le contexte

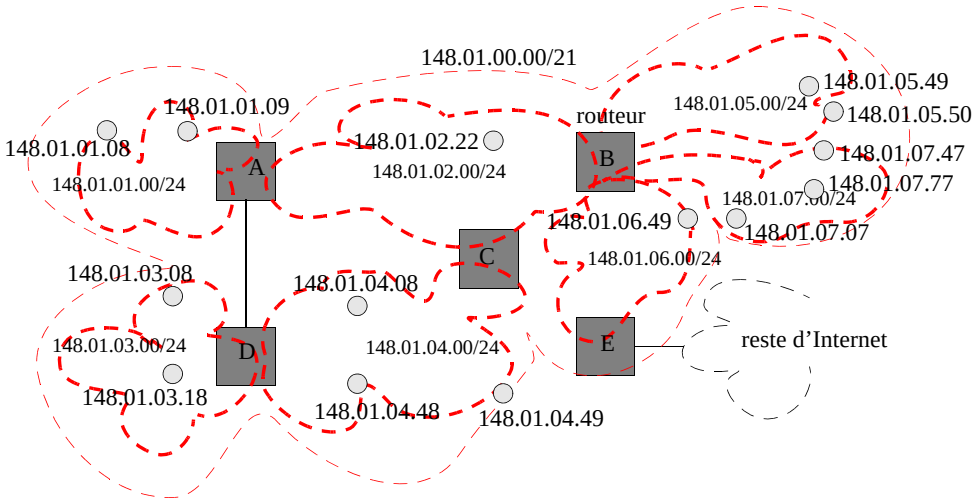
2.4.1 L'infrastructure



Notes : plusieurs routeurs peuvent être connectés au même réseau IP : par ex. A, B, C

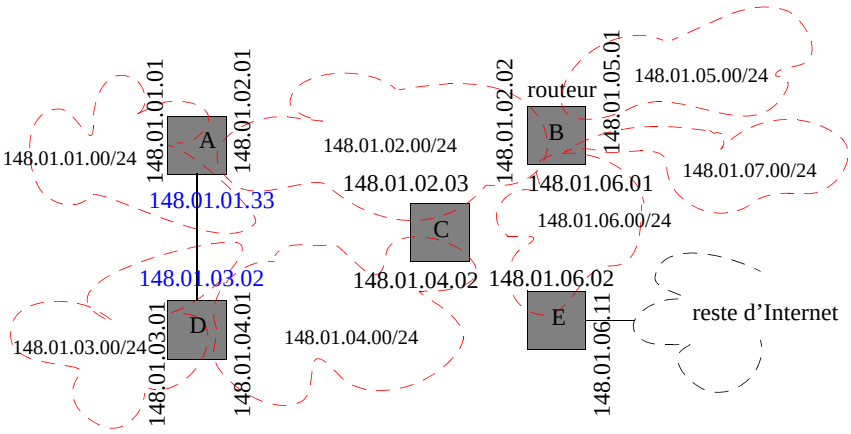
- un routeur peut être connecté à plus de 2 réseaux IP : par ex. B
- les routeurs peuvent être interconnectés par un simple lien (A, D) ou à travers un LAN (A, B)

2.4.2 Les sous-réseaux IP



- Le réseau IP 148.01.00/21 est partionné en plusieurs sous-réseaux IP
- Plusieurs sous-réseaux IP peuvent se partager le même LAN :
  . par exemple 148.01.05.00/24 et 148.01.07.00/24

2.4.3 Les adresses IP



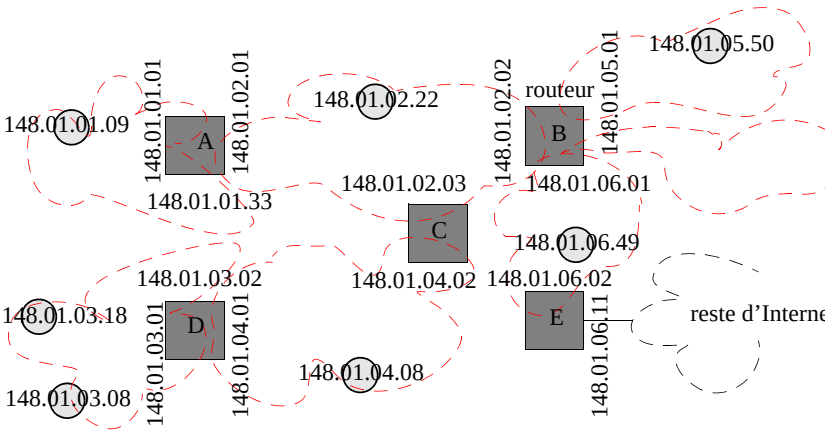
Généralement, on attribue une adresse à chaque interface de chaque routeur :

. en adéquation avec le préfixe du sous-réseau

2.4.4 La table de routage

Tableau 1 : table de routage de D

adresse de destination	prochain routeur
148.01.03.08	148.01.03.01 (*1)
148.01.03.18	148.01.03.01
148.01.04.08	148.01.04.01
148.01.01.09	148.01.01.33 (*2)
148.01.02.22	148.01.04.02
148.01.05.50	148.01.04.02
148.01.06.49	148.01.04.02
0.0.0.0 (*3)	148.01.04.02



Trois types d'entrée dans la table de routage:

- (*1) : sous-réseau à accès direct à partir de D
- (*2) : sous-réseau à accès indirect (par A ou C)
- (*3) : route par défaut

2.4.5 Routes spécifiques

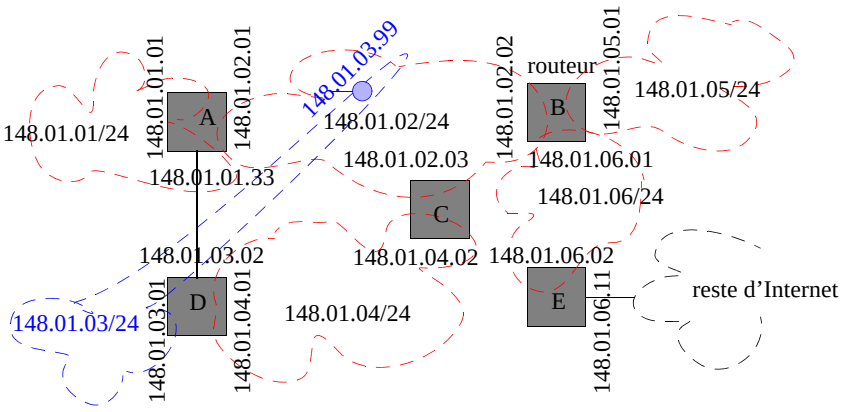


Tableau 2 : table de routage de D

adresse de destination	prochain routeur
148.01.03.99	148.01.04.02
148.01.03.08	148.01.03.01
148.01.03.18	148.01.03.01
…	…
0.0.0.0	148.01.03.02

2.4.6 "Best Prefix Match"

Lors du "datagramme frowarding", on sélectionne l'entrée de la TdR partageant le plus long préfixe avec l'adresse de destination.

Lors de la mise à jour de la TdR, on fusionne les entrées de la TdR:

- adresses consécutives
- et ayant même next hop

Tableau 3 : table de routage de D avant agrégation

adresse de destination	prochain routeur
...	...
148.01.03.08	148.01.03.01
148.01.03.09	148.01.03.01
…	…
0.0.0.0	148.01.03.02

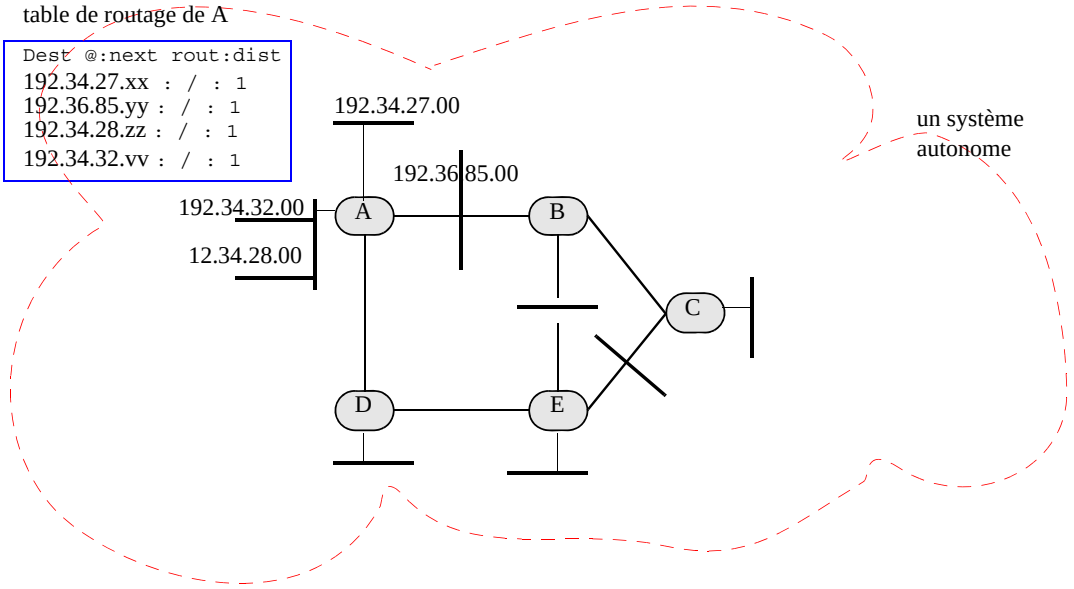
Tableau 4 : table de routage de D aprés agrégation

adresse de destination	prochain routeur
...	...
148.01.03.08	148.01.03.01
…	…
0.0.0.0	148.01.03.02

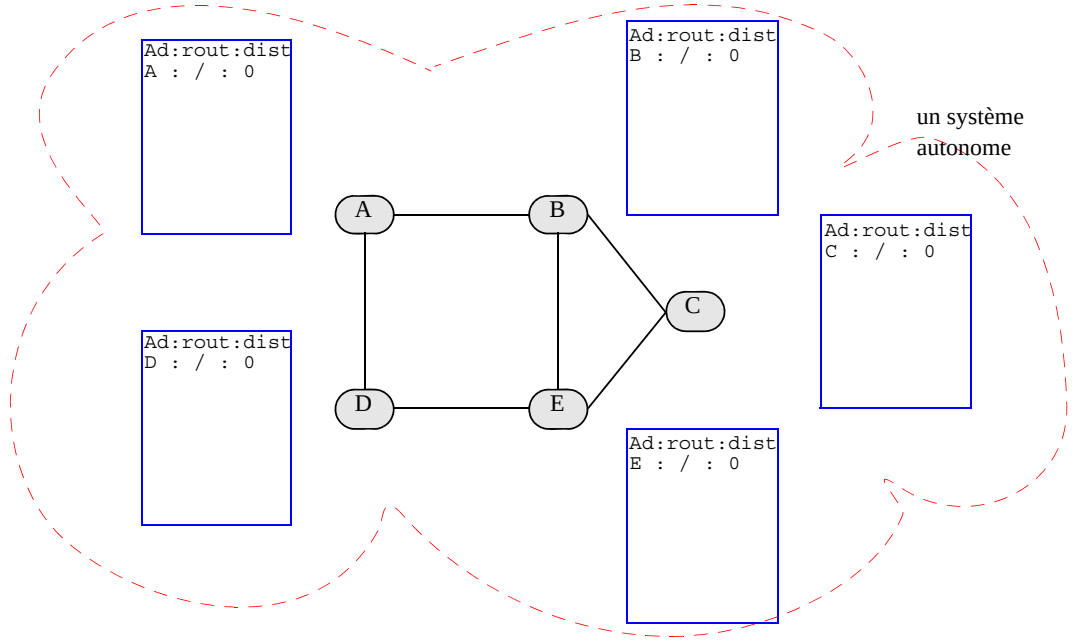
2.5. Illustration des différentes phases de l'algorithme

2.5.1 Initialisation

Lors de son initialisation, un routeur connait tous ses sous-réseaux IP directs



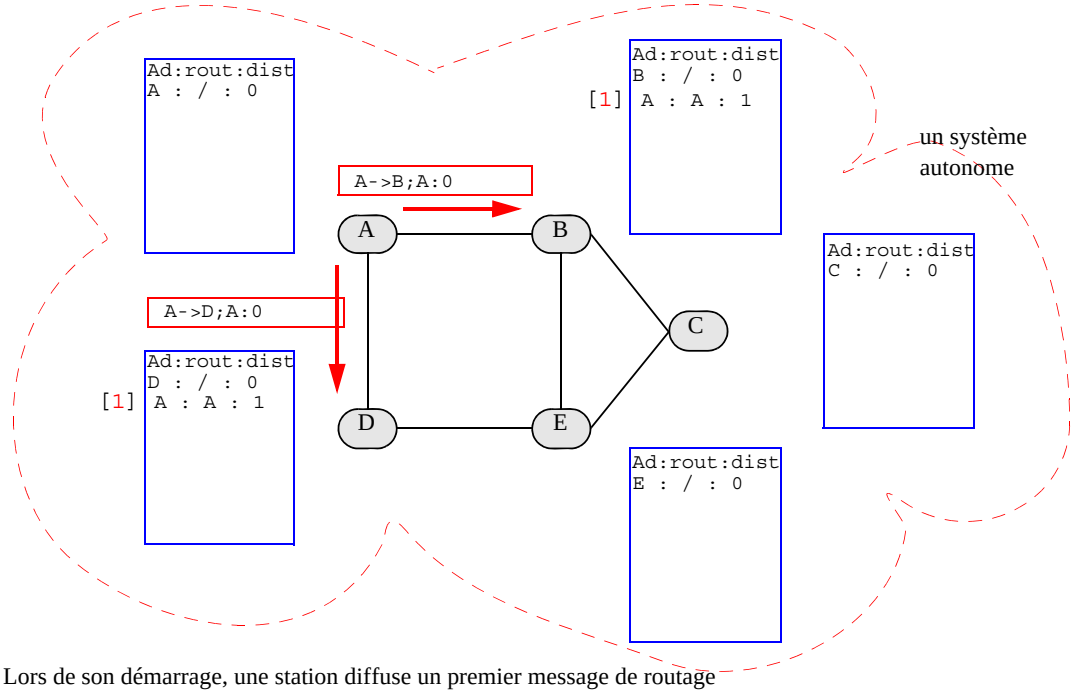
2.5.2 Représentation simplifiée



Simplifications :

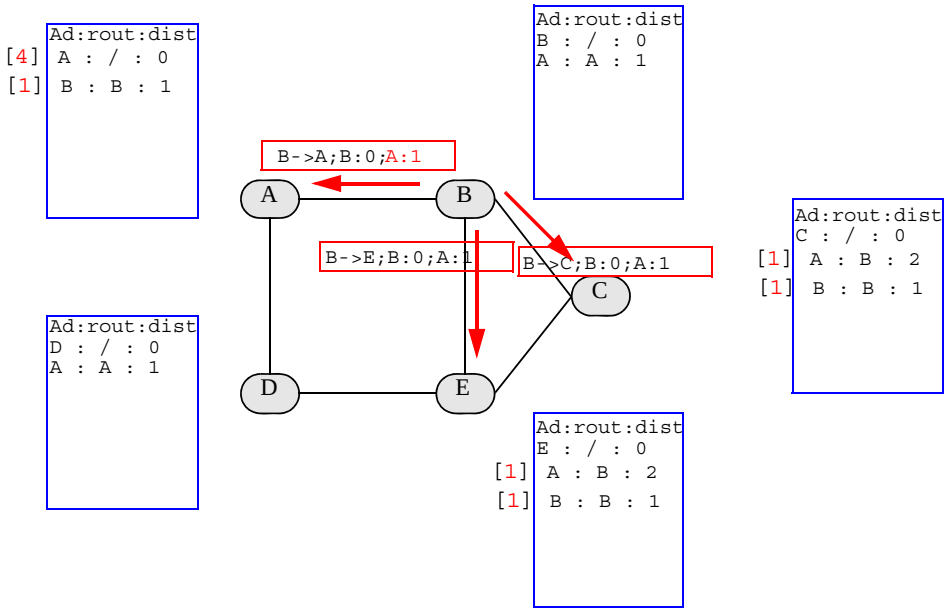
. tous les sous-réseaux IP d'un routeur sont représentés par sa seule adressse
. la distance initiale est notée 0 (précédement c'était 1, et cela pourrait être une fonction inverse du débit !)
. utilisation du nom du routeur et pas de son adresse IP

2.5.3 La première phase



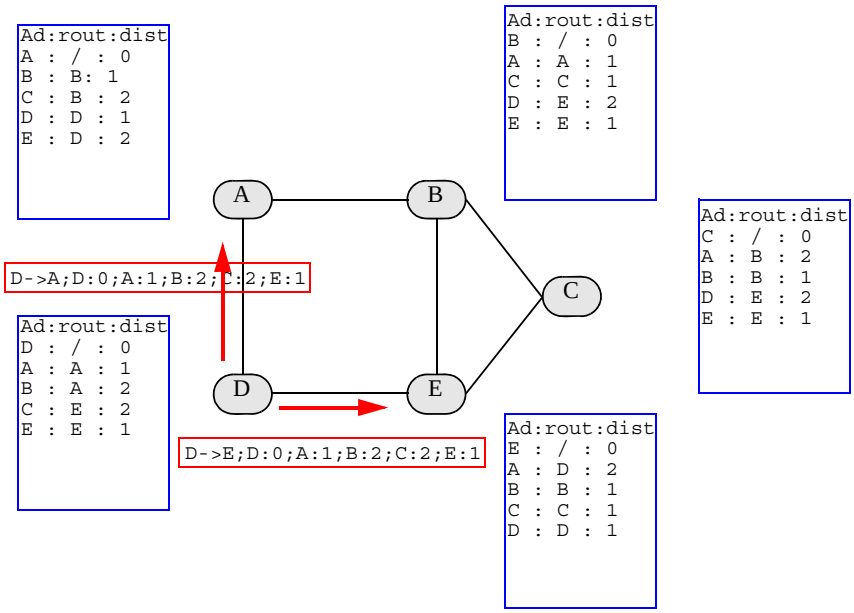
Lors de son démarrage, une station diffuse un premier message de routage

2.5.4 Les phases suivantes



Toute modification de la table locale entraîne la diffusion d'un nouveau message de routage

2.5.5 L'état stable de surveillance



La diffusion des messages de routage est effectuée périodiquement (surveillance de la topologie, perte de message) : gratuitous response (30 s)

3. Quelques problèmes

3.1. Présentation des problèmes

Slow convergence :

Les changements de topologie ne sont pas immédiatement pris en compte :

- il faut que le changement soit détecté et que l'information se propage
  - les routeurs sont nombreux
  - les routeurs sont éloignés

Le rebond :

- des boucles sont créées : certains datagrammes y circulent sans fin (trous noirs)
  - engorgement des liens et des routeurs => destruction des paquets (TTL)

Incrémentation infinie :

- la distance des stations inaccessibles s'accroit (lentement) jusqu'à l'infini

Fiabilité

- détection des pannes de stations
- récupération des pertes et corruptions des messages

3.2. Illustration de quelques problèmes

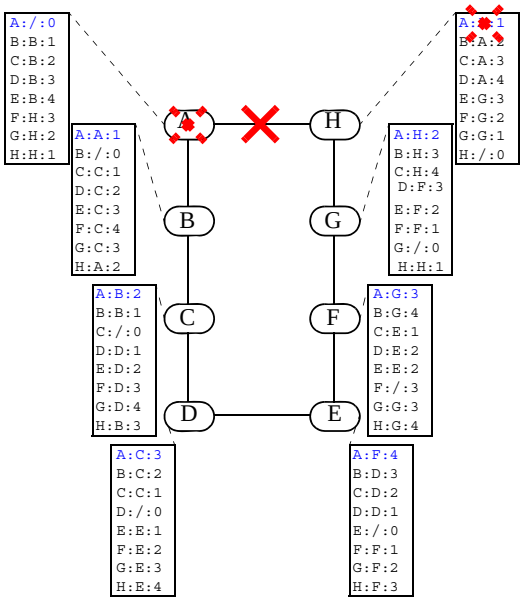
Une topologie simple de 8 stations. On ne s'intéresse qu'à la destination A.

3.2.1 La panne

La destination A devient innaccessible.

- soit l'entrée n'a pas été rafraichie à temps
- soit on a une information explicite que le lien est tombé en panne

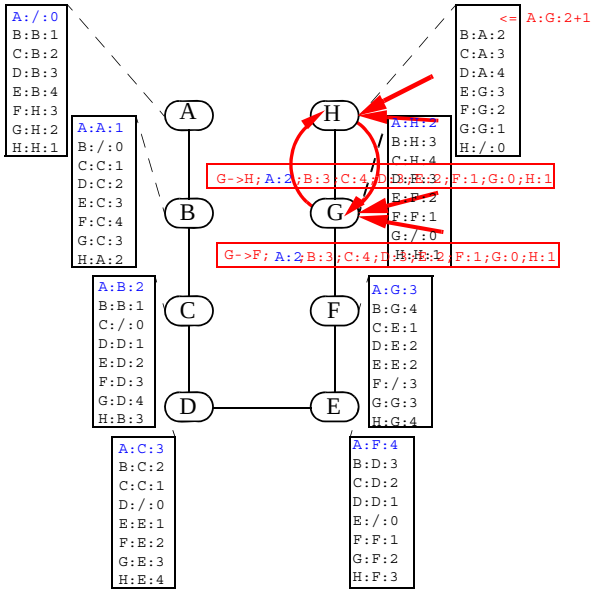
L'entrée correspondante dans la table de routage de H est supprimée



3.2.2 Le rebond

(Périodiquement) G diffuse ses meilleurs routes :

- H reçoit le message de routage de G et met sa table à jour, cas [2] de l'algorithme.

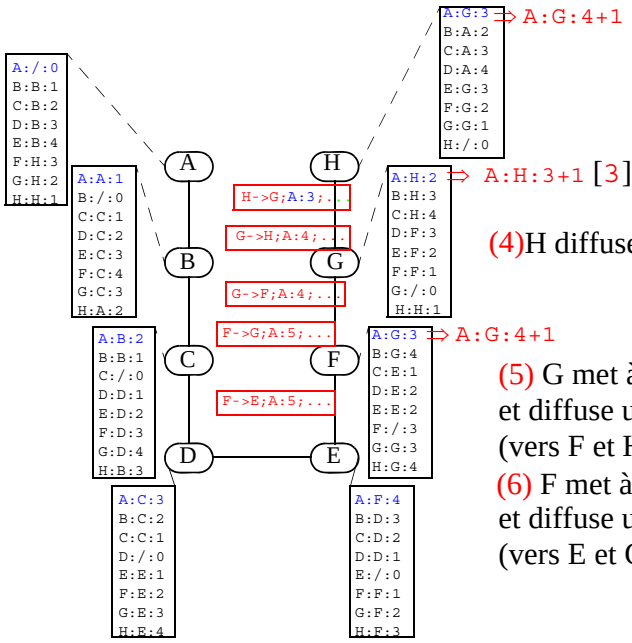


Création d'un circuit G entre H :

- tous les paquets à destination de A passant soit par G soit par H rebondiront :
  - . mauvais routage,
  - . risque de congestion du lien et des routeurs

  ⇒ destruction de paquets (TTL)

3.2.3 La propagation

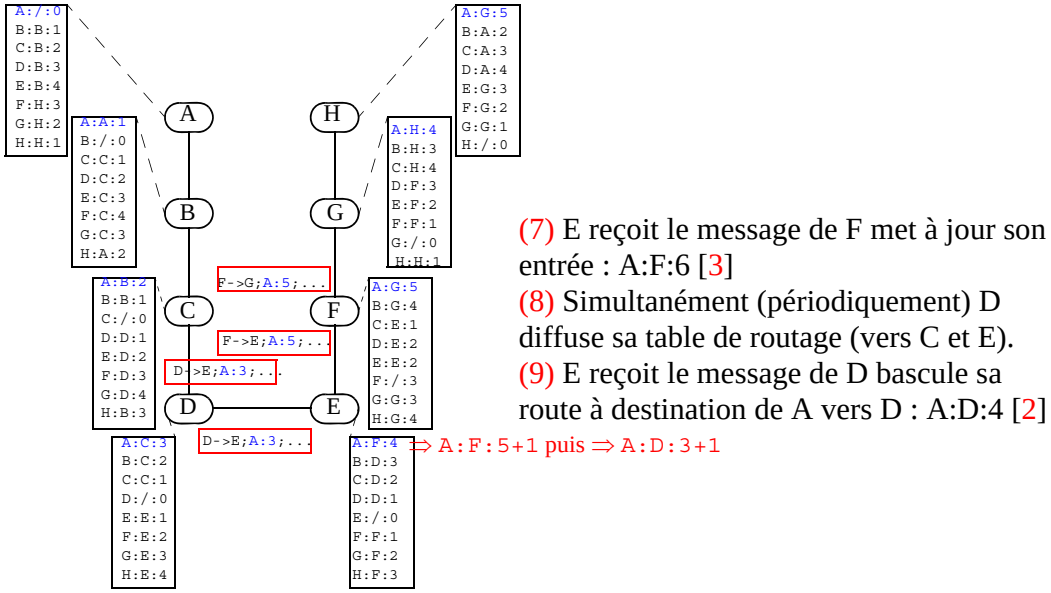


(4)H diffuse un message de routage vers G

(5) G met à jour son entrée : A:H:4 [3] et diffuse un nouveau message de routage (vers F et H)

(6) F met à jour son entrée : A:G:5 [3] et diffuse un nouveau message de routage (vers E et G)

3.2.4 Le basculement

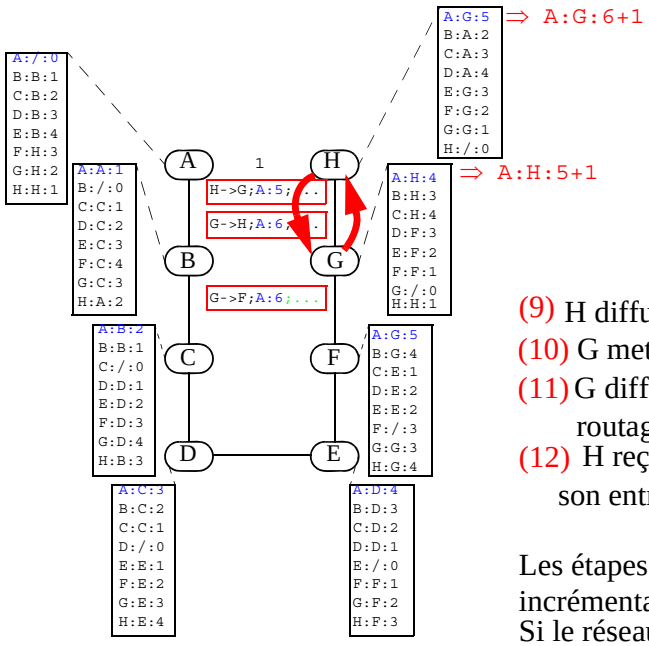


(7) E reçoit le message de F met à jour son entrée : A:F:6 [3]

(8) Simultanément (périodiquement) D diffuse sa table de routage (vers C et E).

(9) E reçoit le message de D bascule sa route à destination de A vers D : A:D:4 [2]

3.2.5 Pendant ce temps-là : le comptage



(9) H diffuse un nouveau message vers G
(10) G met alors à jour sa table : A:F:6 [3]
(11) G diffuse un nouveau message de routage (vers H et F)
(12) H reçoit le message de G et met à jour son entrée : A:G:5 [3]

Les étapes (9) à (11) provoquent une incrémentation continuelle de la métrique. Si le réseau n'est pas connexe, détection tardive : "count to infinity problem"

4. Quelques solutions

4.1. Solutions aux problèmes précédents

Limited infinity

Pour limiter la durée de comptage, la valeur maximale est choisie petite :

- cela a pour conséquence de limiter l'étendu du domaine géré par RIP
- ∞ =16 !

Split horizon update

Une première station n'informe pas une autre station des meilleurs chemins qui passent par cette deuxième station.

- c'était inutile,
- c'était dangereux.
- les messages de routage sont différents en fonction des destinataires
- cela diminue la taille des messages de routage
- cela ne résout que partiellement le problème du rebond :
  . les circuits de plus de 2 stations rebondissent toujours !

4.2. Solutions à l'inaccessibilité

Route time-out

Détection des adresses inaccessibles. Toute destination dont on a plus de nouvelles devient inaccessible :

- durée limitée de validité des entrées de la table de routage (3 mn)
  => 6 pertes de MdR successives

Hold down

On mémorise dans la table de routage les destinations qui ne sont plus accessibles :

- codé ∞
- on conserve cette valeur pendant 4 périodes de mise à jour (2 mn)

Poison reverse

On diffuse les destinations qui deviennent inaccessibles aux voisins

- les messages de routage informent des destinations innaccessibles et non plus seulement des meilleures routes !
- accroît la taille des messages de routage

4.3. Optimisations

Récupération des pertes ou corruptions de message :
Par retransmission périodique des messages de routage (30 s).

- plus la période est grande plus le délai de prise en compte des changements est grand,
- plus la période est petite plus la quantité d'information échangée est importante.

Triggered update :
Un message de routage est diffusé dès que la table de routage a été modifiée.

- prise en compte (presque) immédiate des modifications.

5. Le protocole RIP

5.1. Présentation

Routing Information Protocol :
- RIP-1 : RFC 1058 - juin 1988.
- RIP-2 : RFC 1388 - juin 1993.

routed : Unix RIP routing deamon

commande *netstat -r* : visualise la table de routage
commande *route* : modifie la table de routage
fichier : */etc/hosts* : la table de routage initiale

RIP + UDP + IP
. Port n°520 (service RIP)
. Infini = 16 hops => étendue limitée
. Période de diffusion des message de routage [15-45 s] => moyenne 30 s
. Durée de validité d'un entrée de la TdR (3 mn)
. Délai aléatoire de diffusion immédiate des MdR [0-5 s]
. Split horizon + poison reverse + triggered update + hold down

5.2. Contraintes et avalanches

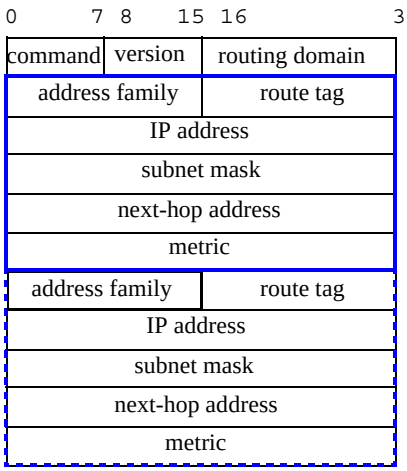
Contraintes

. Les messages de routage ont une longueur limitée : 512 octets
=>le MTU par défaut des datagrammes IP est de 576 octets !
. si les informations à transmettre sont plus longues, on diffuse plusieurs messages de routage.
. le protocole RIP est sans mémoire (“memoryless”), les messages de routage ne sont pas liés (par ex. pas de n°).

Avalanches

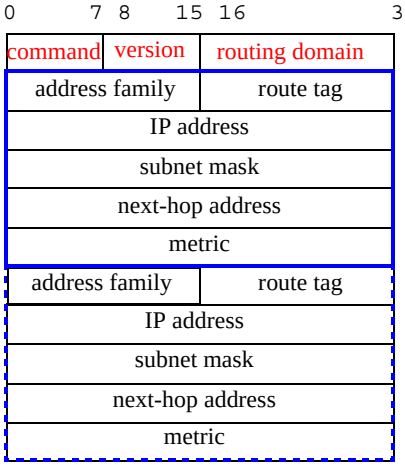
Pour limiter les risques de congestion (avalanche/synchronisation)
les diffusions des MdR sont retardées aléatoirement [RFC 1056] :
- diffusion immédiate [0-5 s]
- diffusion périodique [15-45 s]

5.3. Le format général des messages RIP



. en mots de 32 bits
. longueur < 512 octets
--> limite les traitements de fragmentation au niveau IP
. une entête d'un mot
. autant de blocs de 5 mots que d'entrées à transmettre
- en nombre variable : [1-25]

5.4. L'entête des messages RIP

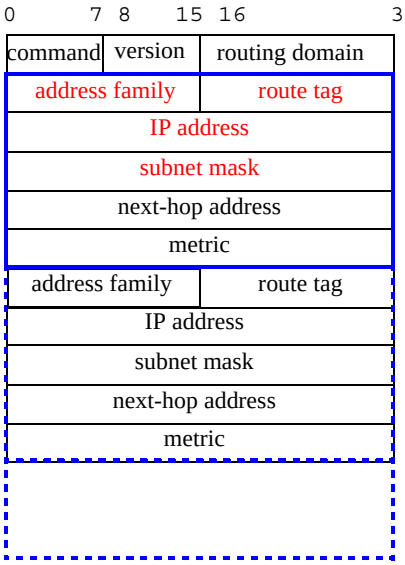


Le champ "command "(8 bits) : code le type du message :
. 1 = demande d'information
- demande partielle pour certaines destinations (dont les entrées figurent dans la demande)
- demande totale (s'il y a une seule entrée associée à la demande tel que "address family"=0 et "metric"=16)
. 2 = réponse
- l'extrait des meilleures routes du routeur
- suite à une demande, envoi périodique, envoi spontané

Le champ "version"(8 bits) :
. 1 = RIP-1 (=>les champs "routing domain", "route tag", "subnet mask", "next-hop address" sont inutilisés = 0)
. 2 = RIP-2

Le champ "routing domain"(16 bits) :
. RIP est générique :
- plusieurs domaines de routage peuvent être gérés simultanément par le même routeur.
. 0 par défaut et obligatoire pour RIP-1

5.5. Les entrées des messages RIP



Le champ “address family “(16 bits) : code le format d'adressage :

. les adresses peuvent être de longueur quelconque

. 2 = IP => 32 bits

Le champ “route tag“(16 bits) :

. transmet des informations utilisées par le routage inter-domaine (EGP). Par ex. le n° d’AS de l’adresse.

. 0 pour RIP-1

Le champ “IP address“(32 bits) : l'adresse de destination

. l'adresse d'un réseau IP (=> netid)

. l'adresse d'un sous-réseau IP (=> subnet mask : subnetid)

. l'adresse d'une station (=>@IP)

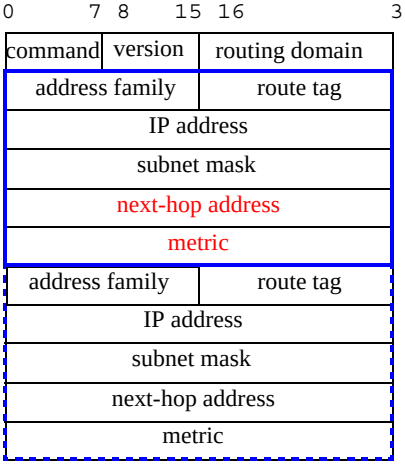
. l'adresse par défaut (=>n’importe quelle destination : 0.0.0.0)

Le champ “subnet mask“(32 bits) :

. 0 pour RIP-1

. spécifie la taille du champ “subnetID” dans l'adresse IP.

Les entrées des messages RIP (suite)



Le champ “next-hop address” (32 bits) :

. contient explicitement l'adresse du prochain routeur qui est associé à l'entrée
(ce n'est plus implicitement l'émetteur du message de routage. Cela permet à un routeur d'informer sur les meilleurs chemins d'un autre routeur).

. 0 = le prochain routeur est l’émetteur du message (RIP-1)

Le champ “metric“(32 bits) :

. distance en nombre de “hops” entre la destination spécifiée par “IP address” et le prochain routeur spécifié, soit par “next-hop address” (RIP-2), soit par l'adresse de l'émetteur du message (RIP-1).

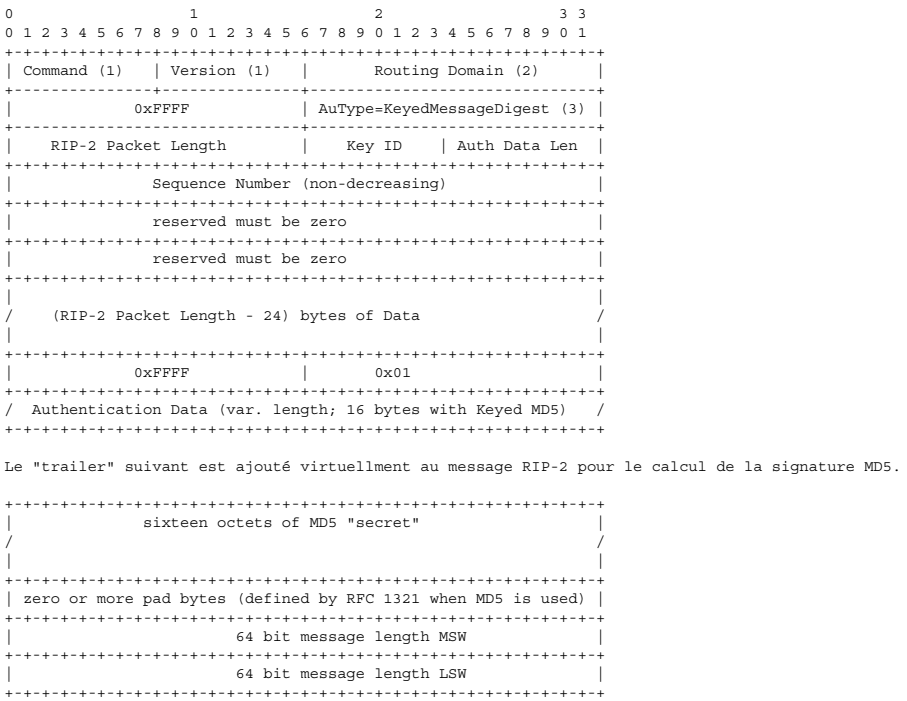
. [1-15] : distance normale

. 16 = distance infinie (destination inaccessible)

5.6. Authentification

Les routeurs sont des équipements sensibles
- il faut pouvoir authentifier les informations données par un routeur
- RIP authentification message :
  . address family = 0xfff
- types d'authentification utilisés :
  . Mot de passe : route tag = 2 (rfc 1723)
    - les 16 octets suivants contiennent un mot de passe (en clair !) --> rejeu
  . MD5 : route tag = 3 (rfc 2082)
    - les 16 octets suivants forment une signature numérique du MdR :
      hachage MD5 (fonction non reversible, avec un paramètre secret) du MdR
      (donc authentification du routeur émetteur et intégrité du MdR)

Format du message RIP-2 avec authentification



Le "trailer" suivant est ajouté virtuellment au message RIP-2 pour le calcul de la signature MD5.

5.7. Optimisation

5.7.1 Broadcast versus multicast address

RIP-1 utilise l'adresse de diffusion locale (255.255.255.255)

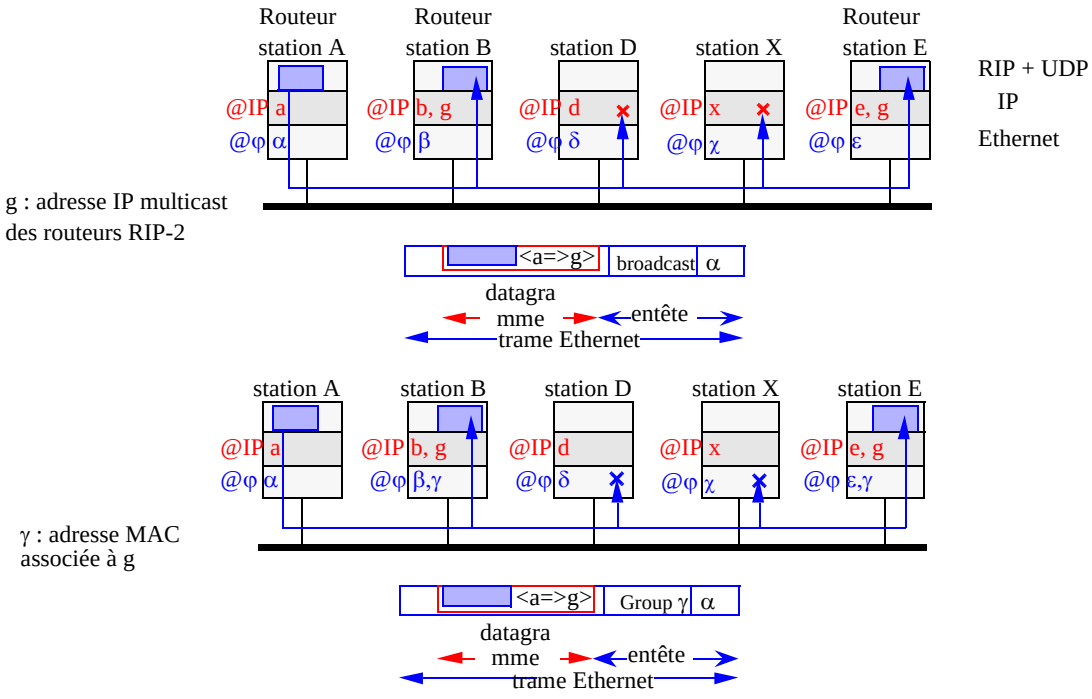
. Toutes les stations reçoivent une copie du message

RIP-2 utilise l'adresse multicast réservée (224.0.0.9 : le groupe des routeurs RIP-2)

. Seuls les routeurs RIP reçoivent une copie du message

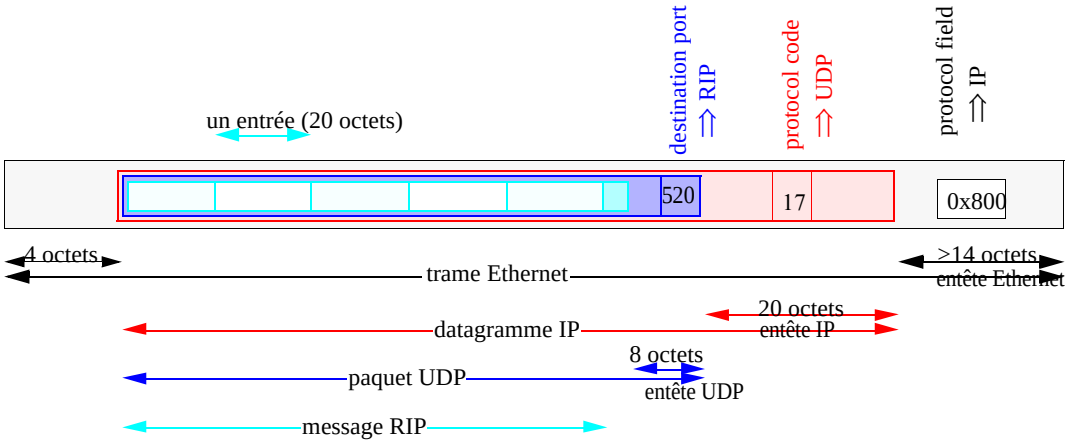
=> moins de surcharge pour les drivers IP et UDP des autres stations ou routeurs

5.7.2 Group MAC address versus Broadcast

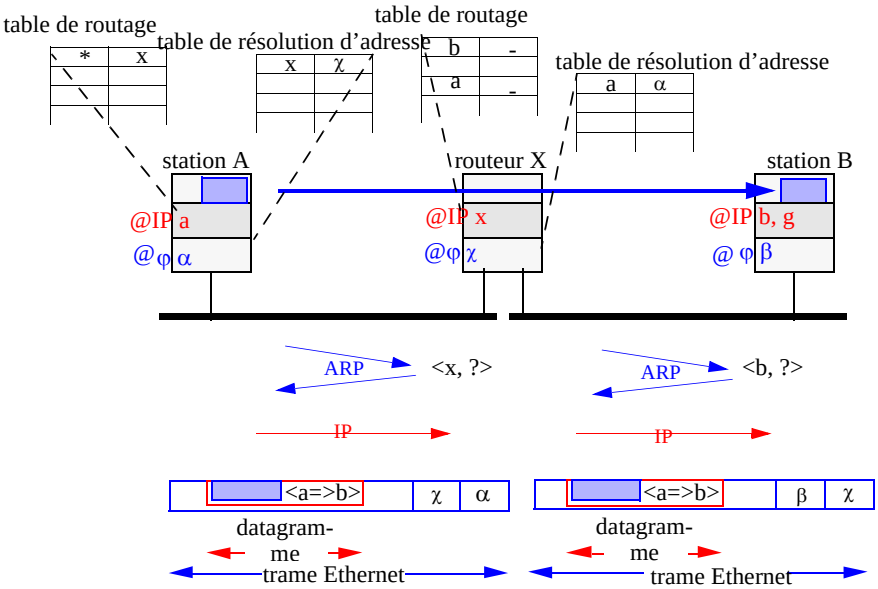


5.8. RIP et les autres protocoles

5.8.1 RIP + UDP + IP (+ Ethernet)



5.8.2 RIP + ARP



6. Conclusion

RIP
Simplicité
Nécessaire à IP

Vitesse de stabilisation faible
Pas de connaissance de l'adressage des sous-réseaux (sauf RIP-2)
Etendu limitée (heureusement) => IGP (Interior Gateway Protocol)
Mono métrique (hop!)
Métrique grossière (hop!)

Nombreux autres protocoles sous Internet :
. OSPF (Open Shortest Path First) : link-state protocol (= OSI IS-IS)
. GGP (Gateway to Gateway Protocol) : distance vector algorithm
. BGP (Border Gateway Protocol) (=+ OSI IDRP : Interdomain Routing Protocol)