



Routage

Master 2 Informatique - UFR S.A.T

Pr. Ousmane THIARE

`ousmane.thiare@ugb.edu.sn`
`http://www.ousmanethiare.com`

12 septembre 2025

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Chapitre 9 : Routage

Chapitre 9 : Routage

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

1 Introduction

2 L'algorithme "Distance Vector"

3 Quelques problèmes

4 Quelques solutions

5 Le protocole RIP

6 Conclusion



Routage

Introduction

L'algorithme "Distance Vector"

Quelques problèmes

Quelques solutions

Le protocole RIP

Conclusion

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



Routage

Introduction

L'algorithme "Distance Vector"

Quelques problèmes

Quelques solutions

Le protocole RIP

Conclusion

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.



Routage

L'algorithme "Distance Vector"

Présentation

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 (pour information) :

- 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 : Short Path First

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion



Routage

L'algorithme "Distance Vector"

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

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.



Routage

L'algorithme "Distance Vector"

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Principe

Chaque routeur maintient localement une liste (BdD) des meilleures routes $\Rightarrow$ table de routage $\langle @$ de destination, distance, $@$ du prochain routeur $\rangle$

Chaque routeur actif diffuse un extrait de sa table de routage (message de routage) :

- Périodiquement (30s)
- A tous leurs voisins immédiats
- Une liste de couple $\langle @$ de destination, distance $\rangle$



Routage

L'algorithme "Distance Vector"

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Principe

Tous les routeurs mettent à jour leur tables de routage en conséquence. L'adresse du prochain routeur est implicitement celui de l'émetteur du message de routage.

Etat des stations :

- Actif (les routeurs) diffusent leurs routes,
- Passif (les stations d'extrémité) écoutent.



Routage

L'algorithme "Distance Vector"

Algorithme de mise à jour

Chaque couple de la liste est comparé aux entrées de la table de routage :

- [1] l'entrée n'existe pas dans la table et la métrique reçue n'est pas infinie :
 - une nouvelle entrée est créée : prochain routeur = routeur d'où provient la liste ; distance = distance reçue + 1.
- [2] l'entrée existe et sa métrique est supérieure à celle reçue :
 - on met à jour l'entrée : prochain routeur = routeur d'où provient la liste ; distance = distance reçue + 1
- [3] l'entrée existe et son prochain routeur est celui d'où provient la liste :
 - distance = distance reçue + 1 (augmentation ou diminution de la distance).
- [4] sinon rien.



Routage

L'algorithme "Distance Vector"

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion



Algorithme de mise à jour

Etat initial :

Chaque routeur connaît son environnement immédiat :

- son adresse, ses interfaces,
- ses (sous-)réseaux directs : distance = 0.

Routage

L'algorithme "Distance Vector"

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

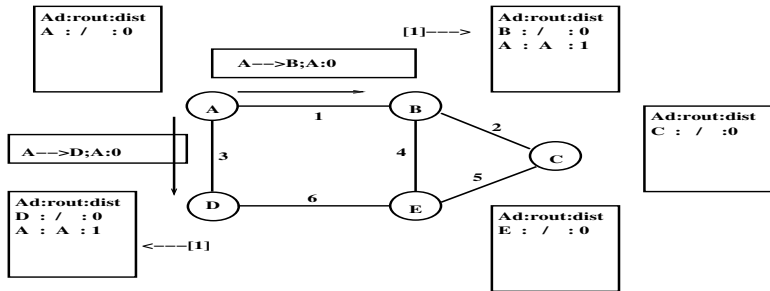
Quelques
solutions

Le protocole RIP

Conclusion

Illustration des différentes phases de l'algorithme

La première phase



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

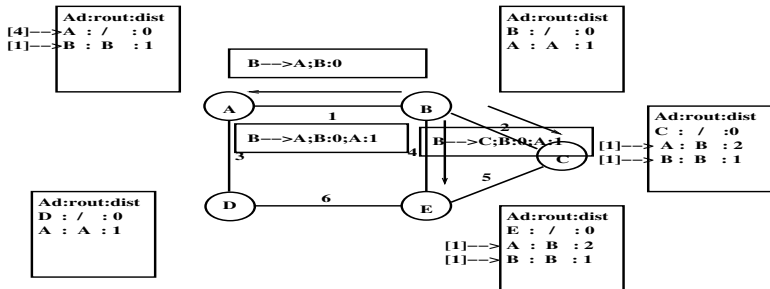


Routage

L'algorithme "Distance Vector"

Illustration des différentes phases de l'algorithme

Les phases suivantes



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



Routage

L'algorithme "Distance Vector"

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

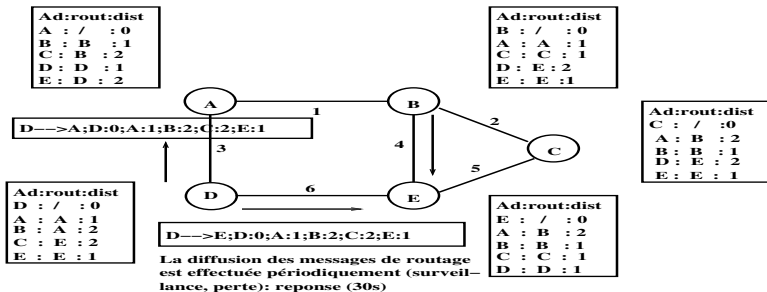
Quelques
solutions

Le protocole RIP

Conclusion

Illustration des différentes phases de l'algorithme

L'état stable de surveillance



Routage

Quelques problèmes

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

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



Routage

Quelques problèmes

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Présentation des problèmes

Slow convergence :

Le rebond

- des boucles sont créées : certains datagrammes y circulent sans fin (trous noirs)

Incrémentation infinie

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



Routage

Quelques problèmes

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Présentation des problèmes

Fiabilité

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



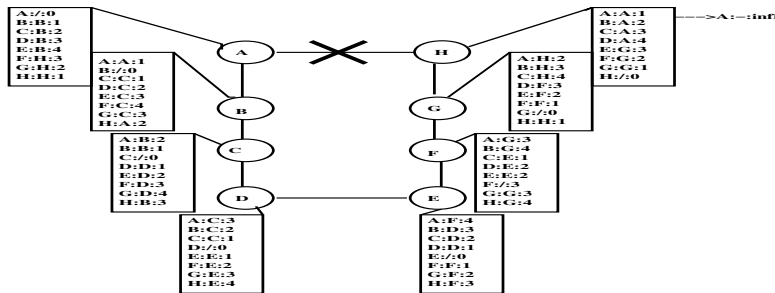
Routage

Quelques problèmes

Illustration de quelques problèmes

La panne Une topologie simple de 8 stations.

On ne s'intéresse qu'à l'accès à la station A à partir des autres stations.



(0) La liaison 1 tombe en panne.

(1) L'entrée correspondante dans la table de routage de H



Routage

Quelques problèmes

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Illustration de quelques problèmes

La panne

- 0) La liaison 1 tombe en panne.
- (1) L'entrée correspondante dans la table de routage de H est invalidée
- (2) Simultanément (périodiquement G diffuse sa table de routage



Routage

Quelques problèmes

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

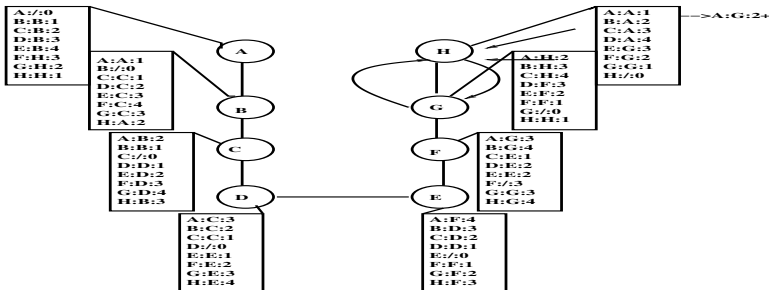
Quelques
solutions

Le protocole RIP

Conclusion

Illustration de quelques problèmes

Le rebond



Routage

Quelques problèmes

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion



Illustration de quelques problèmes

Le rebond

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

Création d'un circuit $G \Leftrightarrow H$: tous les paquets à destination de A passant par G ou H rebondiront entre G et H

mauvais routage, risque de congestion $\Rightarrow$ destruction de paquets (TTL)

Routage

Quelques problèmes

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

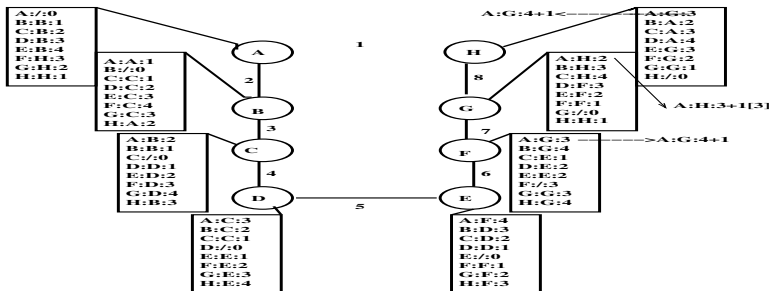
Quelques
solutions

Le protocole RIP

Conclusion

Illustration de quelques problèmes

La propagation



Routage

Quelques problèmes

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Illustration de quelques problèmes

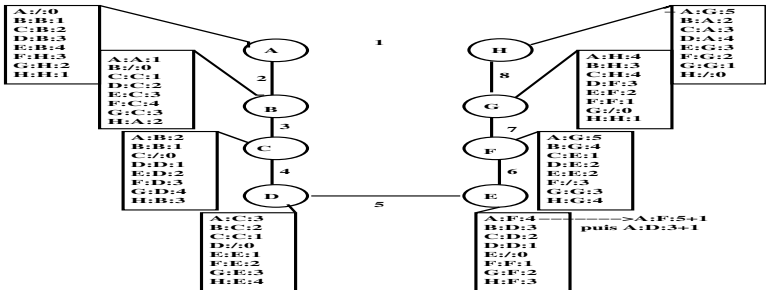
La propagation

- (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)



Quelques problèmes

Le basculement



Routage

Quelques problèmes

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion



Illustration de quelques problèmes

Le basculement

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

Routage

Quelques problèmes

Introduction

L'algorithme
"Distance
Vector"

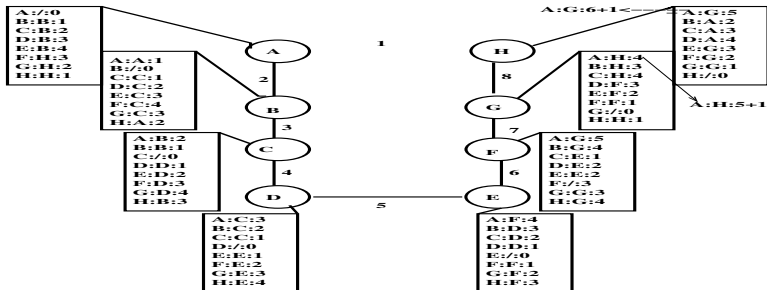
Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Illustration de quelques problèmes Pendant ce temps là : le comptage



Routage

Quelques problèmes

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Illustration de quelques problèmes

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 continue de la métrique : "count to infinity problem"



Routage

Quelques solutions

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

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
- $-\infty = 16!$





Solutions aux problèmes précédents

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 !

Routage

Quelques solutions

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion



Solutions à l'inaccessibilité

Route time-out

Détection des stations inaccessibles. Toute station dont on a plus de nouvelles devient inaccessible :

- durée limitée de validité des entrées de la table de routage

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

Solutions à l'inaccessibilité

Poison reverse

On diffuse les destinations qui deviennent inaccessibles aux voisins

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





Solutions à l'inaccessibilité

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

- 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 immédiate des modifications.

Routage

Le protocole RIP

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Présentation

Routing Information Protocol :

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

Triggered update :

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

- prise en compte immédiate des modifications.

routed : Unix RIP routing daemon

commande netstat -r : visualise la table de routage

commande route : modifie la table de routage

fichier : /etc/hosts : la table de routage initiale



Routage

Le protocole RIP

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Présentation

RIP + UDP + IP

- Port numéro 520 (service RIP)
- Infini = 16 hops ? étendue limitée
- Période de diffusion des message de routage [15-45s]
- Durée de validité d'un entrée (3 mn)
- Délai aléatoire de diffusion immédiate [0-5s]
- Split horizon + poison reverse + triggered update + hold down





Contraintes et avalanches

Contraintes

- Les messages de routage ont une longueur limitée : 512 octets $\Rightarrow$ 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"), ces messages ne sont pas liés (par ex. pas de numéro).

Avalanches

Pour limiter les risques de congestion
(avalanche/synchronisation)

les diffusions sont retardées aléatoirement [RFC 1056] :

- diffusion immédiate [0-5s]
- diffusion périodique [15-45s]

Routage

Le protocole RIP

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Le format général des messages RIP

0	7	8	15	16	31 bit
command		version		routing domain	
address family			route tag		
IP address					
subnet mask					
next-hop address					
metric					
address family			route tag		
IP address					
subnet mask					
next-hop mask					
metric					

en mots de 32 bits
longueur < 512 octets



Routage

Le protocole RIP

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

L'entête des messages RIP

0	7	8	15	16	31 bit
command		version		routing domain	
address family				route tag	
IP address					
subnet mask					
next-hop address					
metric					
address family				route tag	
IP address					
subnet mask					
next-hop mask					
metric					



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
 - suit à use demand, envoi périodique, envoi spontané



L'entête des messages RIP

Le champ "version" (8 bits) :

- 1=RIP-1 ($\Rightarrow$ les champs "routine romain", "route tag", "subnet mask", "next-hop address" sont inutilisés=0)
- 2=RIP-2

Le champ "routine domaine" (16 bits)

- RIP est générique :
 - plusieurs domaines peuvent être gérés simultanément par le même routeur
 - 0 par défaut et obligatoire pour RIP-1



Routage

Le protocole RIP

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Les entrées des messages RIP

0	7	8	15	16	31 bit
command		version		routing domain	
address family				route tag	
IP address					
subnet mask					
next-hop address					
metric					
address family				route tag	
IP address					
subnet mask					
next-hop mask					
metric					



Routage

Le protocole RIP

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

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 $\Rightarrow$ (32 bits)

Le champ "route tag" (16 bits) :

- transmet des informations utilisées par le routage interdomaine
- 0 pour RIP-1



Routage

Le protocole RIP

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Les entrées des messages RIP

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

- l'adresse d'un réseau IP ($\Rightarrow$ netid)
- l'adresse d'un sous-réseau IP ($\Rightarrow$ subnet mask : subnetid)
- l'adresse d'une station ($\Rightarrow$ @IP)
- l'adresse par défaut ($\Rightarrow$ n'importe quelle destination : 0 :0 :0 :0)

Le champ "subnet mask" (32 bits) :

- 0 pour RIP-1
- spécifie la taille du champs "subnetID" dans le champs "host" de l'adresse IP



Routage

Le protocole RIP

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion



Les entrées des messages RIP (suite)

0	7	8	15	16	31 bit
command		version		routing domain	
address family			route tag		
IP address					
subnet mask					
next-hop address					
metric					
address family			route tag		
IP address					
subnet mask					
next-hop mask					
metric					

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=RIP-1



Les entrées des messages RIP (suite)

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)



Routage

Le protocole RIP

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Améliorations

Authentification :

- les routeurs sont des équipements sensibles
- il faut pouvoir authentifier les informations données par un routeur
- RIP authentication message :
 - address family = 0xff
- type d'authentification :
 - route tag = 2
 - les 16 octets suivants contiennent une clef d'authentification.



Routage

Le protocole RIP

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

Optimisation

- 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)
 - Seuls les routeurs RIP reçoivent une copie du message ⇒ moins de surcharge pour les drivers IP des autres stations et autres routeurs.



Conclusion

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

RIP

- Simplicité
- Nécessaire à IP.
- Vitesse de stabilisation faible
- Pas de connaissance de l'adressage des sous-réseaux (sauf RIP-2)
- Etendue limitée (heureusement) $\Rightarrow$ IGP (Interior Gateway Protocol)
- Mono métrique



Conclusion

Introduction

L'algorithme
"Distance
Vector"

Quelques
problèmes

Quelques
solutions

Le protocole RIP

Conclusion

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 RP)

