

Chapitre 4: Les Protocoles de Routage

Fatma Louati Ben Mustapha
Routage dans les Réseaux
2^{ème} année Ingénieur Info
ENICarthage

Plan

1. Les protocoles de routage dynamique
2. Présentation des protocoles de routage vecteur distance
 1. Découverte du réseau
 2. Maintenance des tables de routage
 3. Boucles de routage
 4. Exemples de protocoles vecteur distance
3. Présentation des protocoles à états des liens
 1. Présentation de l'algorithme SPF
 2. Processus de routage d'états des liens
 1. Étude des réseaux directement connectés
 2. Envoi de paquets Hello
 3. Création du LSP
 4. Diffusion des LSP aux voisins
 5. Création de la base de données d'états des liaisons
 6. Arborescence SPF
 3. Mise en œuvre des protocoles à états des liens

1. Présentation des protocoles de routage dynamique

- Utilisés pour faciliter l'échange d'informations de routage entre routeurs (surtout en cas de modification de topologie)
 - Permettent aux routeurs de partager de manière dynamique des informations sur les réseaux distants et d'ajouter automatiquement ces informations à leurs propres tables de routage
 - découverte automatique de nouveaux réseaux ➡ autres chemins en cas d'échec d'une liaison vers un réseau actif

1.1. Comparaison avec routage statique

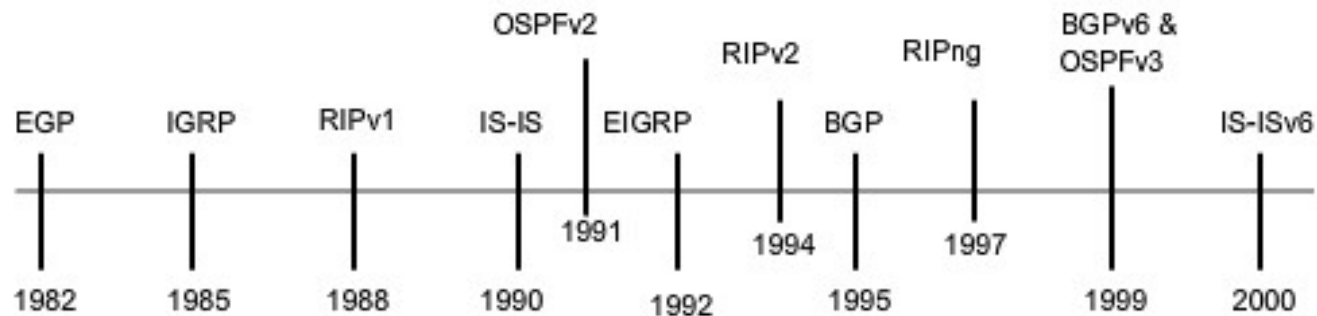
- Routage statique principalement utilisé pour:
 - Faciliter maintenance tables de routage dans réseaux de petite taille non amenés à se développer de manière significative
 - Effectuer routage depuis et vers réseaux d'extrémité
 - Utiliser une seule route par défaut, servant à représenter un chemin vers tout réseau ne présentant aucune correspondance plus spécifique avec une autre route indiquée dans la table de routage

1.1. Comparaison avec routage statique

	Routage dynamique	Routage statique
Complexité de la configuration	Généralement indépendant de la taille du réseau	Augmente avec la taille du réseau
Connaissances d'administrateur requises	Connaissances avancées requises	Aucune connaissance supplémentaire n'est requise
Modifications apportées à la topologie	S'adapte automatiquement aux modifications apportées à la topologie	Intervention de l'administrateur requise
Évolutivité	Idéal pour les topologies simples et complexes	Idéal pour les topologies simples
Sécurité	Moins sécurisé	Plus sécurisé
Utilisation de la ressource	Utilise l'UC, la mémoire, la bande passante de la liaison	Aucune ressource supplémentaire n'est requise
Prévisibilité	La route dépend de la topologie actuelle	La route menant à la destination est toujours la même

1.2. Evolution des protocoles dynamiques

- Utilisation depuis le début des années 80
 - Première version de RIP en 1982
- De nouveaux protocoles de routage à mesure de l'évolution et de la complexité des réseaux



1.3. Définitions et composants

- Protocole de routage = ensemble de processus, algorithmes et messages utilisés pour échanger des informations de routage et construire la table de routage (\subset meilleurs chemins)
 - Permet d'effectuer les opérations suivantes :
 - Découverte des réseaux distants
 - Actualisation des informations de routage
 - Choix du meilleur chemin vers des réseaux de destination
 - Capacité à trouver un nouveau meilleur chemin si le chemin actuel n'est plus disponible
- Composants protocole de routage:
 - Structures des données - tables et/ou BD conservées dans mémoire vive
 - Algorithme - pour faciliter échange info de routage et déterminer le meilleur chemin d'accès
 - Messages - pour découvrir routeurs voisins, échanger infos de routage et autres tâches pour découvrir et gérer des info précises sur le réseau.

1.4. Fonctionnement

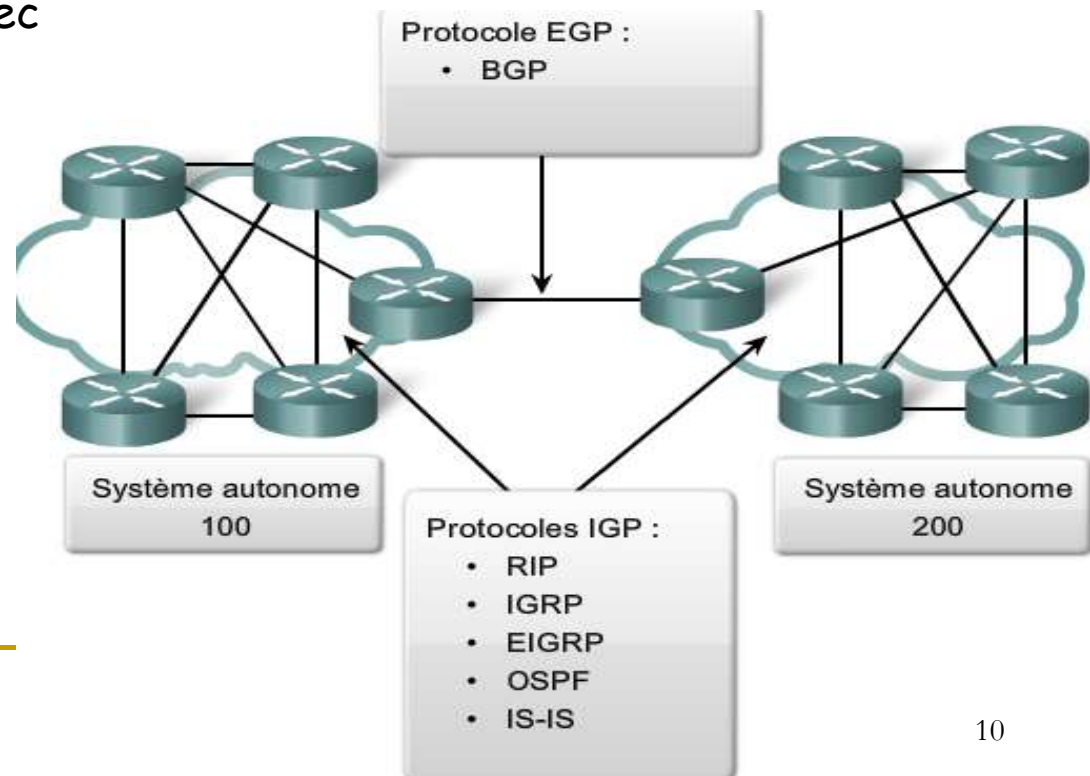
- Fonction = découvrir des réseaux distants et s'adapter rapidement en cas de modification de la topologie
- Méthode adoptée = $f(\text{algorithme}, \text{caractéristiques de fonctionnement})$
- D'une manière générale:
 - Routeur envoie et reçoit messages de routage sur ses interfaces
 - Routeur partage messages et infos de routage avec d'autres routeurs qui utilisent le même protocole de routage
 - Routeurs échangent infos de routage pour découvrir des réseaux distants
 - Lorsqu'un routeur détecte une modification topologique, le protocole de routage peut l'annoncer aux autres routeurs

1.5. Classification

- Trois classifications possibles:
 - Internes et externes
 - Sans classe ou avec classes
 - Vecteur distance ou états des liaisons

Protocoles IGP et EGP

- Système autonome (SA), domaine de routage ↔ ensemble de routeurs avec administration commune (nombreux réseaux individuels appartenant à des entreprises, des établissements scolaires et autres institutions)
 - Protocoles **IGP** (*Interior Gateway Protocol*) utilisés pour routage interne du système autonome → même algorithme
 - Protocoles **EGP** (*Exterior Gateway Protocol*) utilisés pour routage entre systèmes autonomes → avec administrations distinctes



Par classe ou sans classe

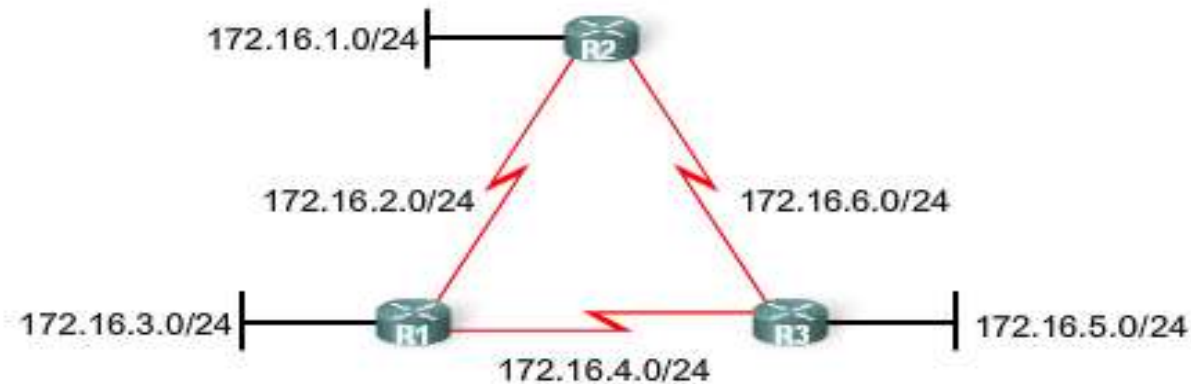
■ Protocole de routage par classe

- Pas d'envoi d'info sur masques de sous-réseau dans mises à jour de routage
 - masque déterminé en fonction du premier octet de l'adresse réseau
- Encore utilisés dans certains réseaux actuels qui ne prennent pas en charge les masques de sous-réseau de longueur variable (VLSM)
- Autre limite:
 - Incapacité à prendre en charge les réseaux discontinus.
- Ex: RIPv1 et IGRP.

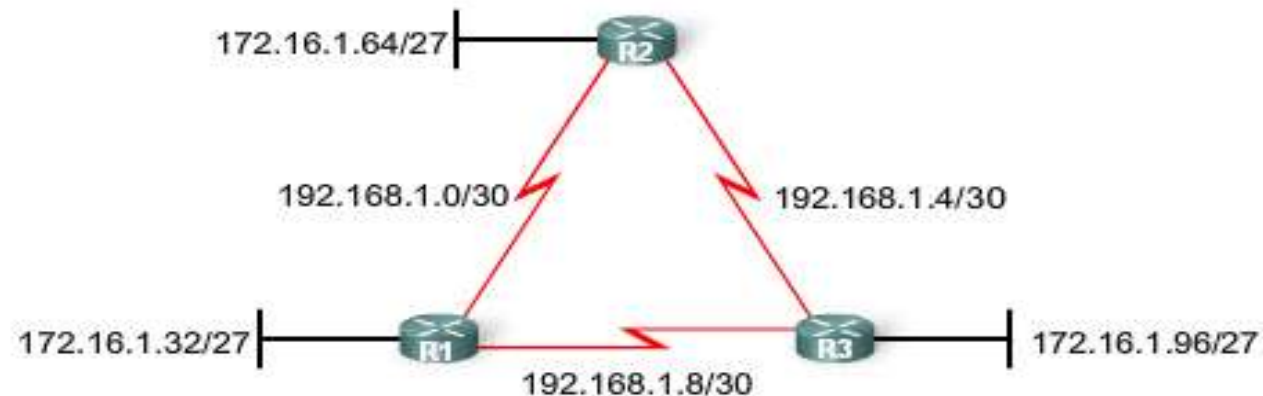
■ Protocole de routage sans classe

- Envoi d'info sur masques de sous-réseau avec adresses réseau dans mises à jour de routage
- Dans la plupart des réseaux actuels (VLSM)
- Ex:
 - RIPv2, EIGRP, OSPF, BGP

Par classe ou sans classe



Par classe : le masque de sous-réseau est identique dans toute la topologie



Sans classe : le masque de sous-réseau peut varier dans la topologie

Vecteur distance et états des liaisons

- **Protocole de routage vecteur distance**
 - Pas besoin de connaître tout le réseau avant d'échanger les informations de routage = vue partielle du réseau
 - Particulièrement adaptés si:
 - Réseau simple, linéaire
 - Administrateurs non suffisamment expérimentés pour configurer et dépanner les protocoles d'état des liaisons
 - Aucun problème même avec délais de convergence extrêmement longs

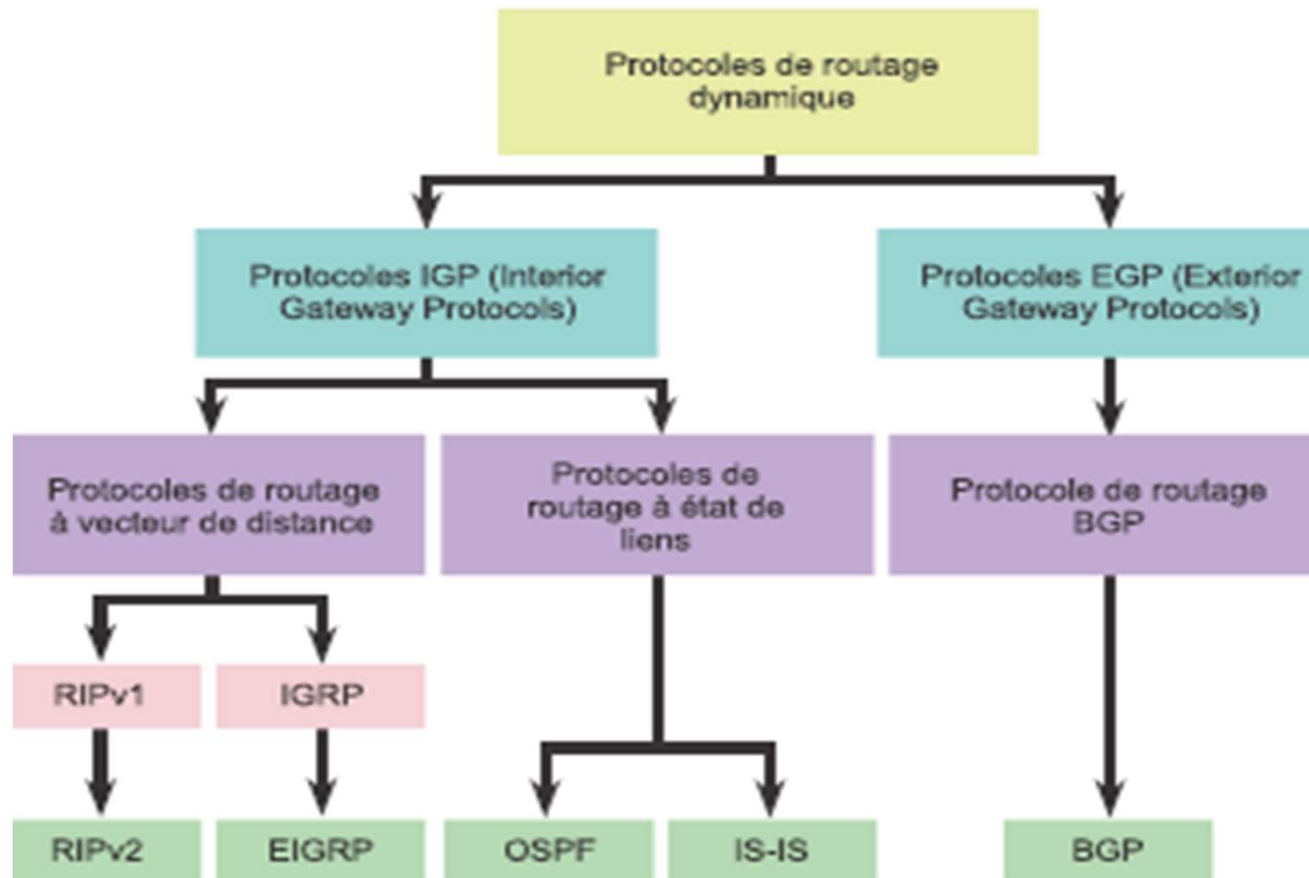
Vecteur distance et états des liaisons

- **Protocole de routage à états des liaisons**
 - Création de « vue complète » en récupérant des informations provenant de tous les autres routeurs → utilisation des informations d'état des liaisons pour créer une topologie et sélectionner le meilleur chemin vers tous les réseaux de destination de la topologie
 - Particulièrement adaptés si:
 - Réseau conçu de manière hiérarchique (grands réseaux)
 - Administrateurs avec bonne connaissance du protocole de routage d'état des liaisons
 - Réseaux pour lesquels une convergence rapide est primordiale

Classification des protocoles dynamiques

Protocoles IGP (Interior Gateway Protocols)				Protocoles EGP (Exterior Gateway Protocols)
	Protocoles de routage à vecteur de distance	Protocoles de routage d'état des liaisons		Protocole BGP
Par classe	RIP	IGRP		EGP
Sans classe	RIPv2	EIGRP	OSPFv2	IS-IS
IPv6	RIPng	EIGRP pour IPv6	OSPFv3	IS-IS pour IPv6
				BGPv4 pour IPv6

Classification des protocoles dynamiques

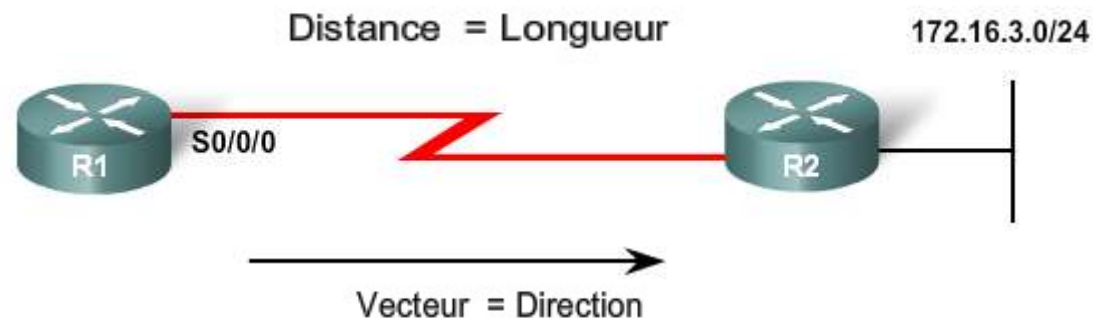


LES PROTOCOLES DE ROUTAGE VECTEUR DISTANCE

1. Technologie vecteur de distance



- Routes annoncées en tant que vecteurs de distance et de direction
 - Distance = mesure (nombre de sauts)
 - Direction = routeur de tronçon suivant ou l'interface de sortie
- Routeur sans connaissance du chemin complet vers un réseau de destination, ne connaît que:
 - direction ou interface où paquets doivent être transmis
 - distance le séparant du réseau de destination

■ Ex:





Pour R1, 172.16.3.0/24 est distant d'un saut (distance).
Il est accessible via R2 (vecteur).

1. Technologie vecteur de distance

- Caractéristiques en commun des protocoles à vecteur de distance:
 - **Mises à jour régulières** envoyées à intervalles fixes même si topologie inchangée
 - **Voisins** = routeurs partagent liaison et configurés avec le même protocole de routage
 - Routeur ne connaît que @ de ses propres interfaces et @ réseaux distants qu'il peut atteindre via ses voisins  **ne connaît pas la topologie du réseau**
 - **Mises à jour de diffusion** envoyées à 255.255.255.255 traitées par routeurs voisins configurés avec le même protocole de routage.
 - Possibilité d'utilisation d'@ de multidiffusion
 - **Mises à jour de toute la table de routage** envoyées régulièrement à tous les voisins (ou presque tous) et traitées par voisins  trouver informations pertinentes et supprimer le reste
 - Exception: EIGRP; pas d'envoi maj régulières de la table de routage

2. Algorithme protocoles de routage

- Algorithme Bellman-Ford  au centre du protocole 
 - Mécanisme d'envoi et de réception des informations de routage
 - Mécanisme de calcul des meilleurs chemins et d'installation de routes dans la table de routage
 - Mécanisme de détection des modifications topologiques et de réaction à celles-ci

3. Caractéristiques des protocoles de routage

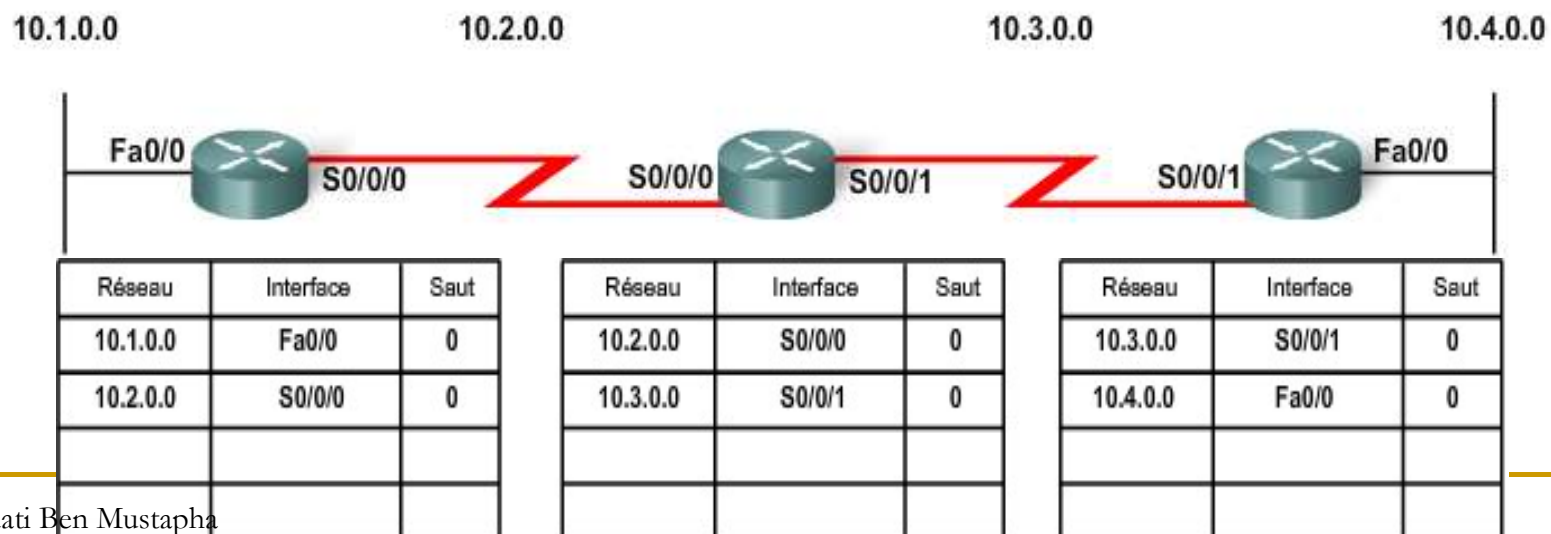
- **Temps de convergence** - rapidité de partage d'informations de routage et de disposition d'une base de connaissances cohérente
 - Table de routage incohérentes non mises à jour en raison de convergence plus lente dans un environnement réseau changeant → boucles de routage
- **Évolutivité** - taille maximale d'un réseau en fonction du protocole de routage déployé
- **Sans classe** (utilisation d'un masque VLSM) ou par classe
- **Utilisation ressources** - Exigences protocole telles que espace mémoire, utilisation processeur et utilisation bande passante de liaison
- **Implémentation et maintenance** - connaissances nécessaire à un administrateur réseau pour implémenter et gérer le réseau en fonction du protocole de routage déployé

3. Caractéristiques des protocoles vecteur distance

Avantages :	Inconvénients :
Implémentation et maintenance simples. Le niveau de connaissances requis pour déployer et effectuer la maintenance ultérieure d'un réseau avec un protocole à vecteur de distance n'est pas élevé.	Convergence lente. L'utilisation de mises à jour périodiques peut ralentir la convergence. Même si des techniques avancées sont utilisées, telles que les mises à jour déclenchées qui seront abordées ultérieurement, la convergence globale est toujours plus lente que celle constatée avec les protocoles de routage d'état des liaisons.
Faibles ressources requises. Les protocoles à vecteur de distance n'ont généralement pas besoin de grandes quantités de mémoire pour stocker les informations. Ils ne nécessitent pas non plus une UC puissante. Selon la taille du réseau et l'adressage IP implémentés, ils ne nécessitent généralement pas une bande passante importante pour envoyer les mises à jour de routage. Cependant, cela peut devenir un problème si vous déployez un protocole à vecteur de distance dans un réseau important.	Évolutivité limitée. La convergence lente peut limiter la taille du réseau car des réseaux plus importants nécessitent davantage de temps pour propager les informations de routage.
	Boucles de routage. Des boucles de routage peuvent survenir lorsque des tables de routage incohérentes ne sont pas mises à jour en raison d'une convergence lente dans un réseau changeant.

4. Découverte du réseau

- Démarrage à froid d'un routeur (mis sous tension) ➡ aucune information sur topologie réseau
 - Seules informations = fichier de configuration dans mémoire vive non volatile
 - Après succès d'amorçage, application configuration enregistrée
- ➡ Détection réseaux connectés directement si adressage IP correctement configuré
- **Exemple:** après démarrage à froid et avant échange info de routage:

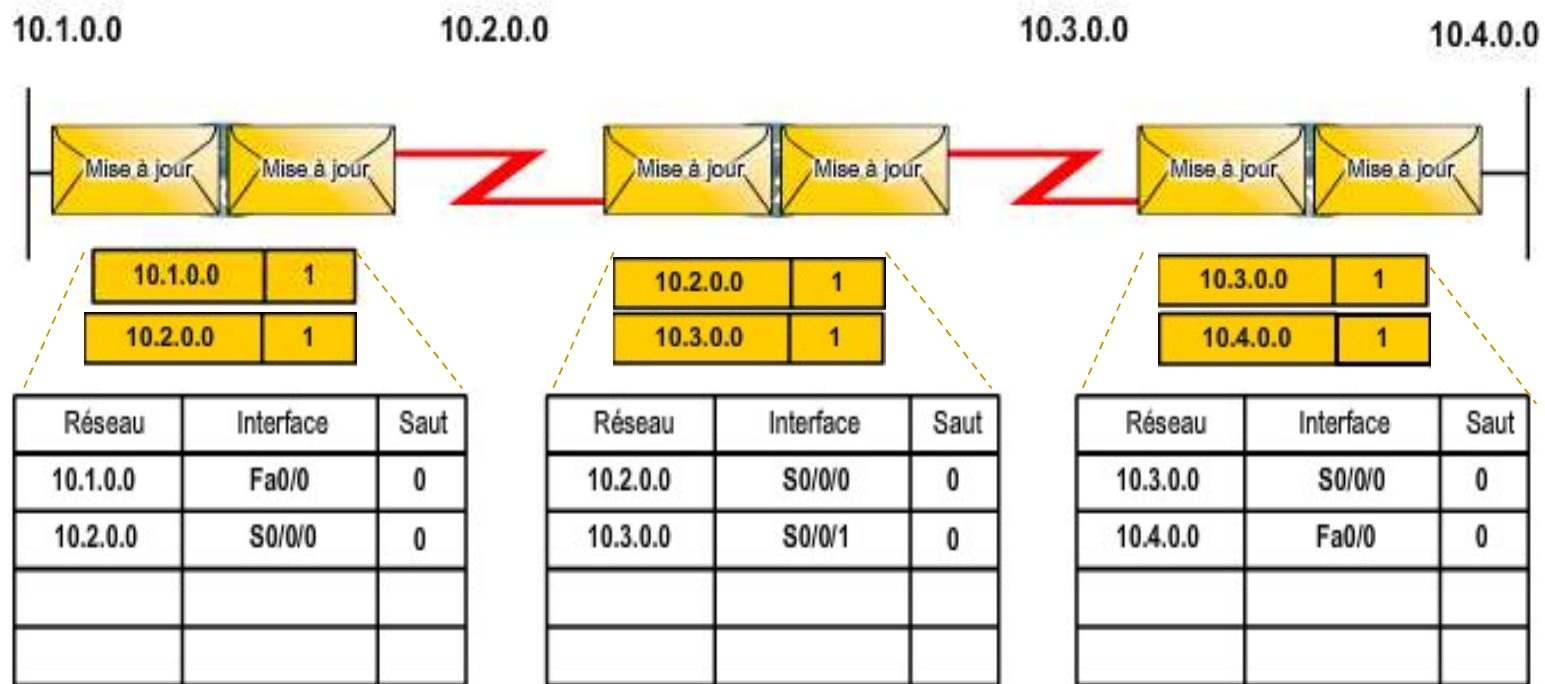


4. Découverte du réseau

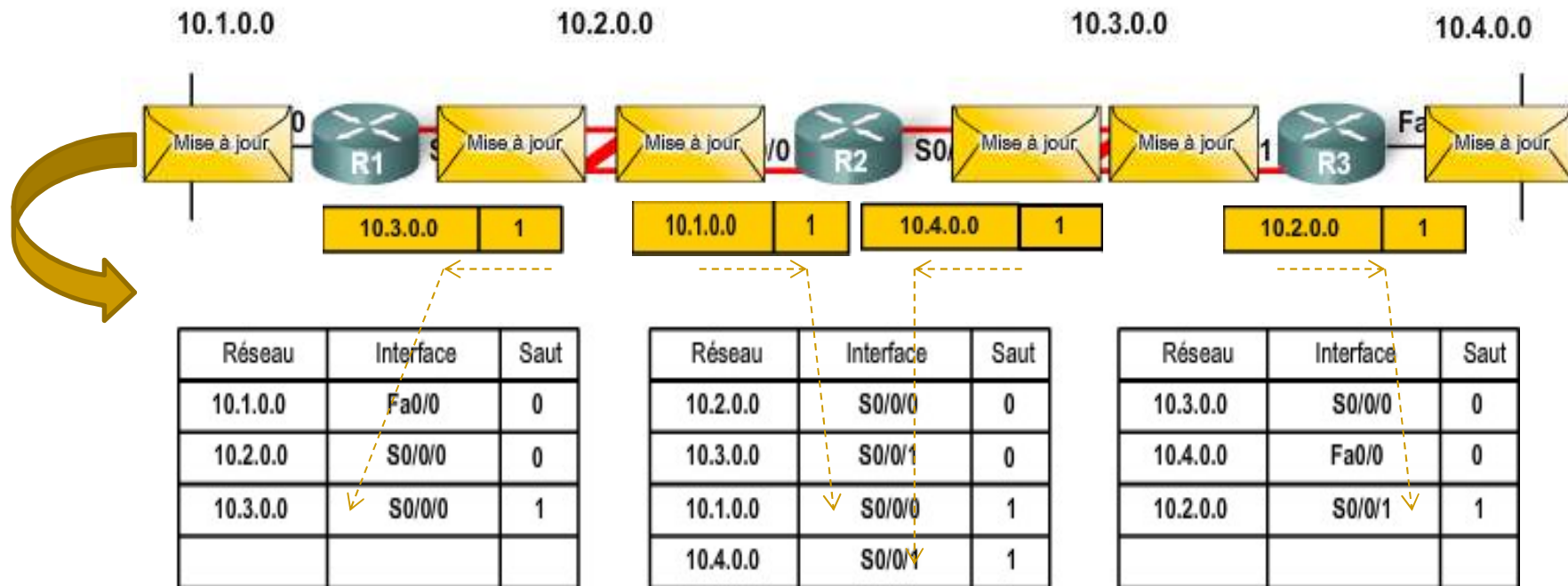
- Si protocole de routage configuré ➡ routeurs commencent à échanger des mises à jour de routage
- **Départ:** mises à jour avec informations concernant leurs réseaux directement connectés
- **Réception mise à jour:** recherche nouvelles informations ➡ ajout de toute route ne figurant pas actuellement dans la table de routage

4. Découverte du réseau

- Retour exemple:



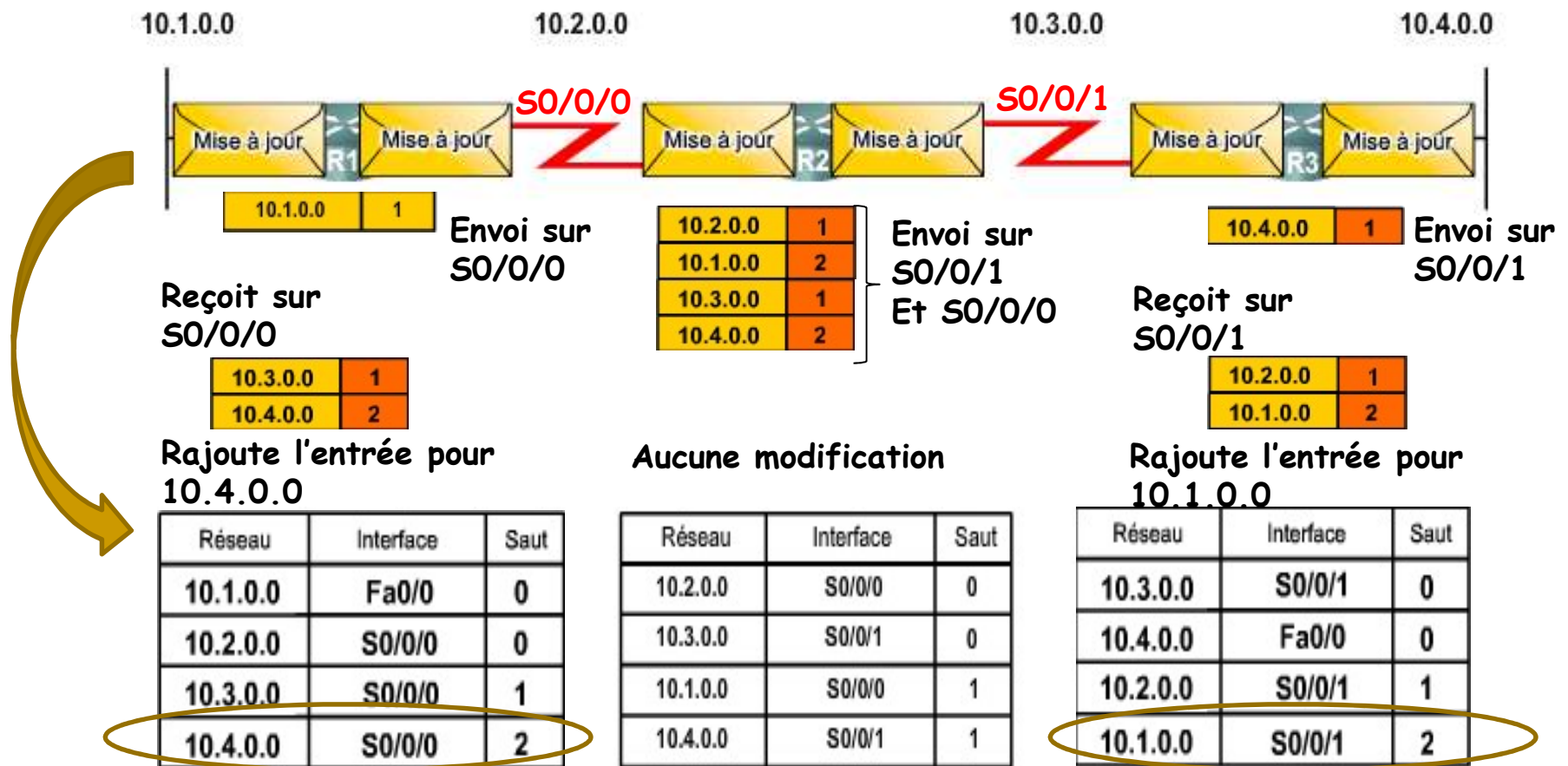
4. Découverte du réseau



- Après première série échanges de vecteur, chaque routeur connaît les réseaux connectés de ses voisins connectés directement
 - ❑ R1 ne connaît pas encore l'existence du réseau 10.4.0.0
 - ❑ R3 ne connaît pas encore l'existence du réseau 10.1.0.0
- Besoin d'un autre échange d'informations de routage ➡ connaissance complète du réseau et convergence

4. Découverte du réseau

- Deuxième série d'échange: Principe de l'horizon coupé: Un routeur ne peut pas envoyer d'informations via l'interface d'où il les a reçues



5. Convergence

- Temps nécessaire pour converger directement proportionnel à taille réseau
- Protocoles de routage évalués en fonction de la vitesse à laquelle ils peuvent propager ces informations ➡ vitesse de convergence ↔
 - Vitesse de la propagation d'une modification de la topologie lors d'une mise à jour de routage à ses voisins
 - Vitesse du calcul des meilleurs chemins à l'aide des nouvelles informations de routage collectées
- Un réseau n'est pas complètement opérationnel tant qu'il n'a pas convergé ➡ Prférence pour les protocoles de routage avec des temps de convergences courts

6. Mesures

- Découverte de plusieurs routes menant à la même destination ➡ sélection meilleur chemin ↔
 - Évaluation et différenciation des chemins disponibles ➡ utilisation de mesure
- Mesure = valeur utilisée par protocoles de routage pour affecter des coûts d'accès aux réseaux distants
 - Meilleur chemin en fonction de la route ayant la mesure la plus faible.
- Chaque protocole de routage ↔ sa propre mesure, non comparable à mesure utilisée par autre protocole
 - Deux protocoles de routage différents peuvent choisir des chemins différents vers une même destination en raison des mesures qu'ils utilisent
 - Nombre de sauts = mesure la plus facile à considérer

6. Mesures

- Exemple de mesures:
 - **Nombre de sauts** - Mesure simple qui compte le nombre de routeurs qu'un paquet doit traverser
 - **Bande passante** - Influence la sélection du chemin en préférant celui dont la bande passante est la plus élevée
 - **Charge** - Prend en considération l'utilisation d'une liaison spécifique en termes de trafic
 - **Délai** - Prend en considération le temps nécessaire à un paquet pour parcourir un chemin
 - **Fiabilité** - Évalue la probabilité d'échec d'une liaison, calculée à partir du nombre d'erreurs de l'interface ou des échecs précédents de la liaison
 - **Coût** - Valeur déterminée par l'IOS ou par l'administrateur réseau pour indiquer une route préférée. Le coût peut représenter une mesure, une combinaison de mesures ou une stratégie.

6. Mesures

- Protocole de routage RIP
- Mesure associée à une route particulière affichée avec `show ip route`
- Valeur mesure = seconde valeur entre crochets d'une entrée de la table de routage
 - R2 présente une route vers 192.168.8.0/24 duquel il est séparé par 2 sauts

```
R2#show ip route  
(**résultat omis**)
```

```
Gateway of last resort is not set
```

```
R   192.168.1.0/24 [120/1] via 192.168.2.1, 00:00:24, Serial0/0  
C   192.168.2.0/24 is directly connected, Serial0/0  
C   192.168.3.0/24 is directly connected, FastEthernet0/0  
C   192.168.4.0/24 is directly connected, Serial0/1  
R   192.168.5.0/24 [120/1] via 192.168.4.1, 00:00:26, Serial0/1  
R   192.168.6.0/24 [120/1] via 192.168.2.1, 00:00:24, Serial0/0  
                                [120/1] via 192.168.4.1, 00:00:26, Serial0/1  
R   192.168.7.0/24 [120/1] via 192.168.4.1, 00:00:26, Serial0/1  
R   192.168.8.0/24 [120/2] via 192.168.4.1, 00:00:26, Serial0/1
```

7. Maintenance tables de routage

- Différentes façons pour maintenir des tables de routage cohérentes avec les différentes mises à jour:
 1. Mises à jour régulières
 2. Mises à jour limitées
 3. Mises à jour déclenchées

7.1. Mises à jour régulières

- Envoi de la table de routage complète aux voisins à un intervalle prédéfini
 - RIP ↔ toutes les 30s sous forme de diffusion (255.255.255.255) que la topologie ait été ou non modifiée
 - « 30s » = minuteur de mise à jour des routes → connaître l'âge des informations de routage dans une table de routage
- Âge informations de routage actualisé à chaque réception d'une mise à jour → mise à jour informations de la table de routage dès la modification de la topologie telle que:
 - Défaillance d'une liaison
 - Introduction d'une nouvelle liaison
 - Défaillance d'un routeur
 - Modification des paramètres de liaison

7.1. Mises à jour régulières



10.1.0.0 10.2.0.0 10.3.0.0 10.4.0.0



Réseau	Interface	Saut
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Réseau	Interface	Saut
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Réseau	Interface	Saut
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

Réseau	Interface	Saut
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Réseau	Interface	Saut
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Réseau	Interface	Saut
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

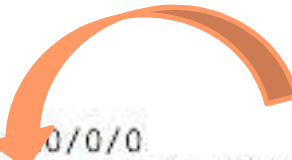
7.1. Mises à jour régulières

```
R1#show ip route  
<output omitted>
```

```
Gateway of last resort is not set
```

```
    10.0.0.0/16 is subnetted, 4 subnets  
C       10.2.0.0 is directly connected, Serial0/0/0  
R       10.3.0.0 [120/1] via 10.2.0.2, 00:00:04, Serial0/0/0  
C       10.1.0.0 is directly connected, FastEthernet0/0  
R       10.4.0.0 [120/2] via 10.2.0.2, 00:00:04, Serial0/0/0
```

Temps écoulé depuis dernière mise à jour (sec)



```
R1#show ip protocols
```

```
Routing Protocol is "rip"
```

```
Sending updates every 30 seconds, next due in 13 seconds  
Invalid after 180 seconds, hold down 180, flushed after 240
```

```
<output omitted>
```

```
Routing for Networks:
```

```
10.0.0.0
```

```
Routing Information Sources:
```

Gateway	Distance	Last Update
10.3.0.1	120	00:00:27

```
Distance: (default is 120)
```

7.2. Mises à jour déclenchées

- Objectif = accélérer convergence en cas de modification de la topologie
 - Mise à jour déclenchée ↔ mise à jour de la table de routage envoyée immédiatement en réponse à la modification d'un routage ↔ sans attendre expiration minuteurs
 - Détection modification → envoi immédiat message de mise à jour aux routeurs adjacents → routeurs récepteurs génèrent à leur tour des mises à jour déclenchées pour informer leurs voisins de la modification
- Mises à jour déclenchées envoyées si:
 - Changement état interface (activée ou désactivée)
 - Passage pour une route à l'état « inaccessible » (ou sort de cet état)
 - Installation route dans table de routage

7.2. Mises à jour déclenchées


- Problèmes associés aux mises à jour déclenchées :
 - Abandon ou endommagement paquets contenant message de mise à jour
 - Mises à jour déclenchées non produites instantanément → possibilité qu'un routeur qui n'a pas encore reçu la mise à jour déclenchée émette une mise à jour régulière au mauvais moment ↔ réinsertion de la route incorrecte dans un voisin ayant déjà reçu la mise à jour déclenchée

7.3. Mises à jour limitées


■ Exemple: EIGRP

- Pas d'envoi de mises à jour régulières, envoi de mises à jour limitées à propos d'une route en cas de modification d'un chemin ou de la mesure pour cette route
- Si nouvelle route disponible ou qu'une route doit être supprimée, EIGRP envoie mise à jour ne concernant que ce réseau et non la table entière.
- Informations envoyées uniquement aux routeurs qui en ont besoin

■ Caractéristiques des mises à jour de EIGRP:

- Non régulières car non envoyées périodiquement
- Mises à jour partielles envoyées uniquement en cas de modification topologique influençant les informations de routage
- Limitées  propagation des mises à jour partielles automatiquement limitée, mise à jour d'uniquement les routeurs ayant besoin de ces informations

8. Boucle de routage

- Boucle de routage  paquet transmis en continu entre une série de routeurs sans jamais atteindre le réseau de destination souhaité
 - Principal inconvénient des protocoles à vecteur distance, moins fréquentes avec les protocoles à états des liaisons
- Plusieurs raisons:
 - Routes statiques configurées incorrectement
 - Redistribution routes configurées incorrectement (transmission des informations de routage d'un protocole de routage à un autre)
 - Tables de routage incohérentes non mises à jour en raison d'une convergence lente dans un réseau changeant
 - Routes de suppression configurées ou installées incorrectement
 - Etc.

8. Boucle de routage

- Exemple de scénario menant à une boucle de routage: panne de liaison



Réseau	Interface	Saut
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Réseau	Interface	Saut
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

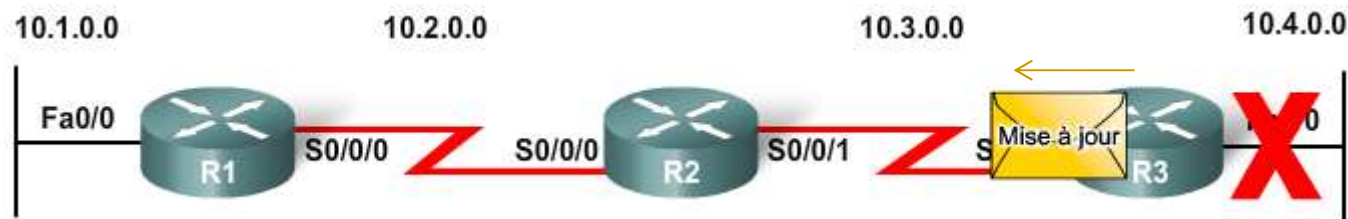
Réseau	Interface	Saut
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	0
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

Nouvelle table de routage de R3 donnant lieu à une boucle pour tout paquet à destination de 10.4.0.0

Réseau	Interface	Saut
10.3.0.0	S0/0/1	0
10.4.0.0	S0/0/1	2
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

8. Boucle de routage

- Plus, comptage à l'infini
 - À l'itération suivante, R3 envoie information de mise à jour à R2



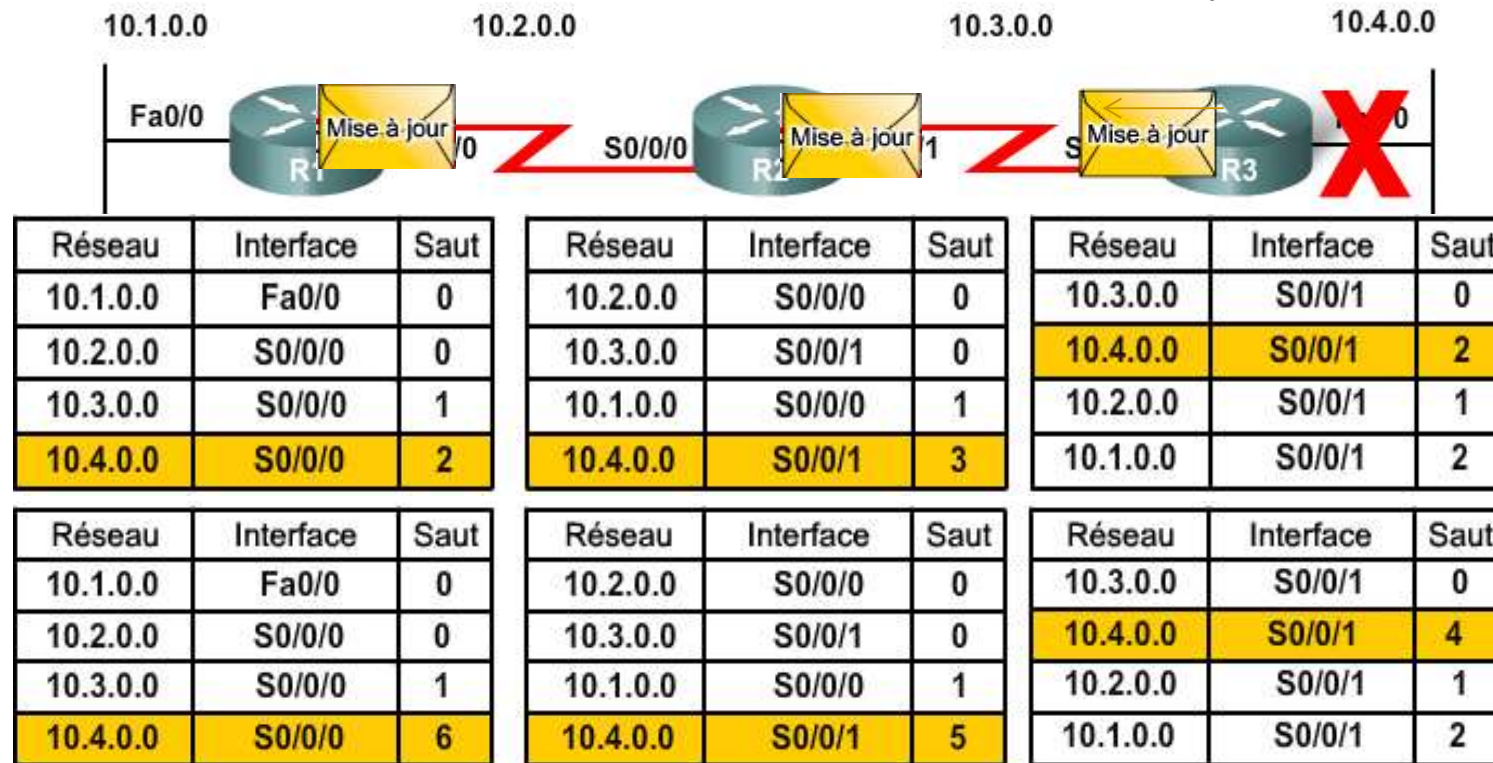
Réseau	Interface	Saut
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Réseau	Interface	Saut
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Réseau	Interface	Saut
10.3.0.0	S0/0/1	0
10.4.0.0	S0/0/1	2
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

8. Boucle de routage

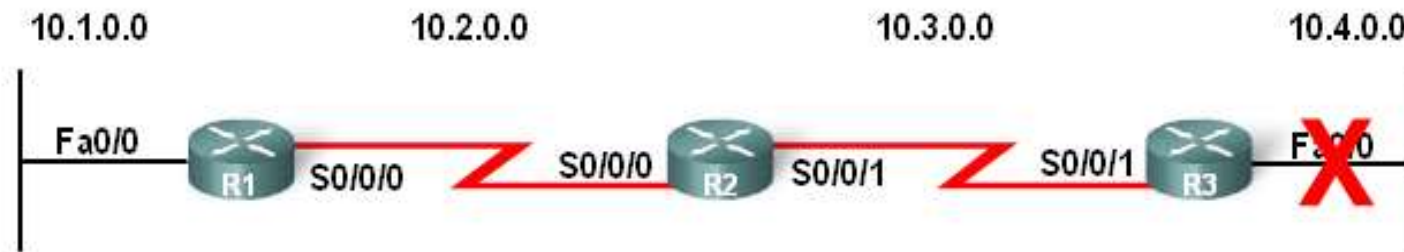
- Plus, comptage à l'infini
 - À l'itération suivante, R3 envoie information de mise à jour à R2



- Etc...

8. Boucle de routage

- Solution au problème de comptage à l'infini ➡ Attribution valeur maximale à la mesure ➡ arrêter incrémentation
 - RIP: « infini »=16 sauts ➡ mesure inaccessible ➡ route marquée inaccessible



Réseau	Interface	Saut
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	16

Réseau	Interface	Saut
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	16

Réseau	Interface	Saut
10.3.0.0	S0/0/1	0
10.4.0.0	S0/0/1	16
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

8.1. prévention de la boucle

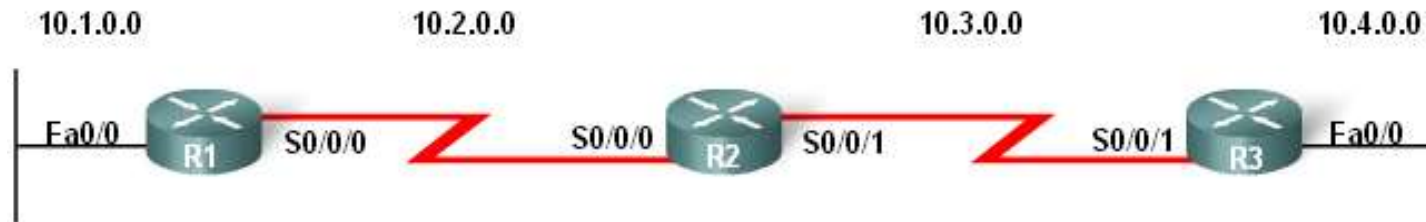
- Plusieurs techniques pour prévenir et éviter la boucle de routage
 1. Règle de découpage d'horizon
 2. Empoisonnement de routage
 3. Découpage d'horizon avec empoisonnement inverse
 4. Le champs TTL du protocole IP

8.1.1. Règle de découpage d'horizon

- Empêcher boucles de routage provoquées par la convergence lente d'un protocole de routage à vecteur de distance
- Règle de découpage d'horizon:
 - Un routeur ne doit pas annoncer de réseau par le biais de l'interface dont est issue la mise à jour
- Peut être désactivé par un administrateur

8.1.1. Règle de découpage d'horizon

- Ex: application de l'horizon coupé à la route 10.4.0.0

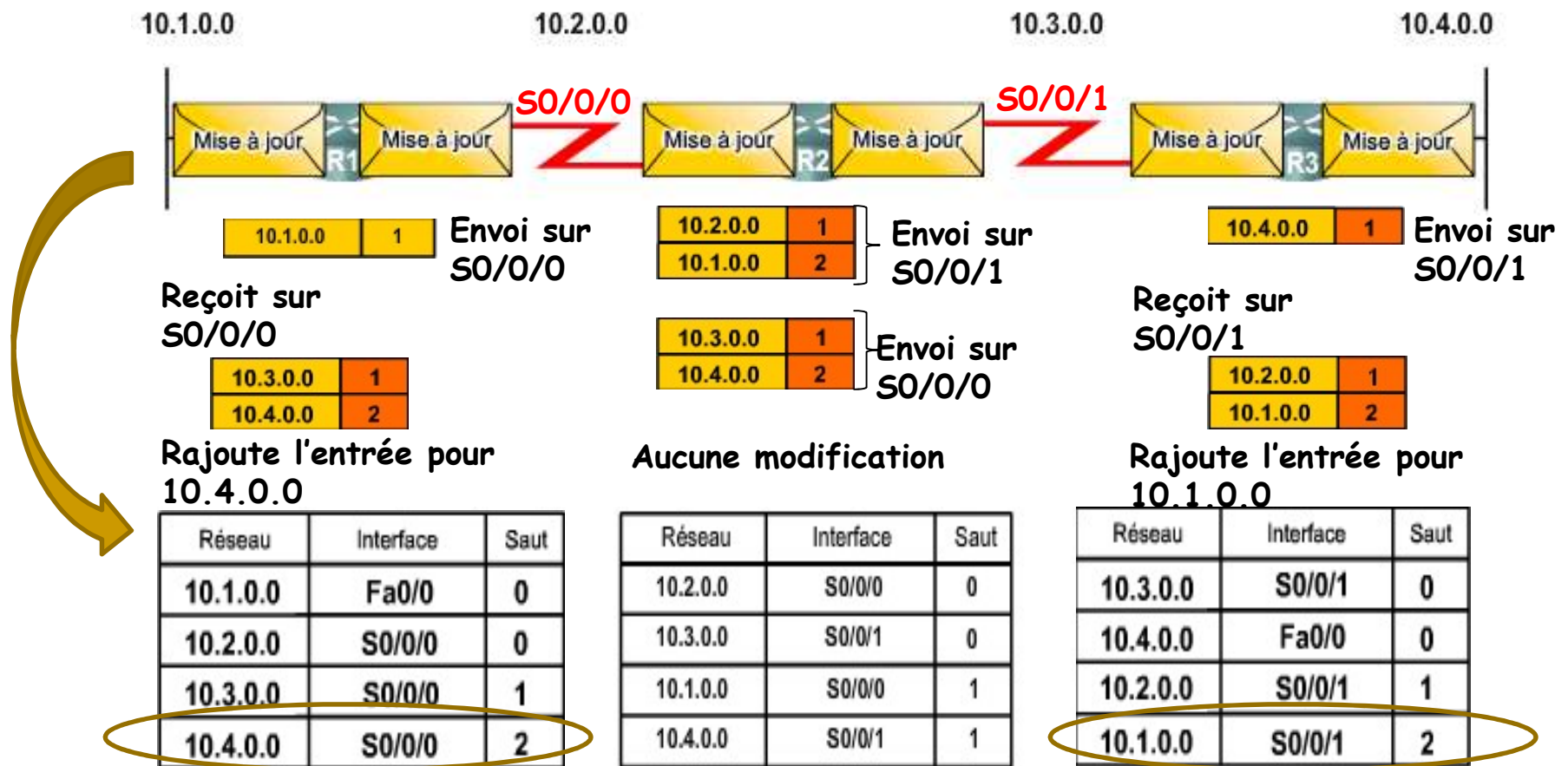


- R3 annonce le réseau 10.4.0.0 à R2
- R2 reçoit l'information et met à jour sa table de routage
- R2 annonce le réseau 10.4.0.0 à R1 via S0/0/0
 - R2 n'annonce pas 10.4.0.0 à R3 via S0/0/1 car route provient de S0/0/1
- R1 reçoit l'information et met à jour sa table de routage
- Découpage d'horizon ➡ R1 n'annonce pas infos sur le 10.4.0.0 à R2
- Échanges de mises à jour de routage complètes, sauf routes transgressant règle de découpage d'horizon
 - R2 annonce les réseaux 10.3.0.0 et 10.4.0.0 à R1
 - R2 annonce les réseaux 10.1.0.0 et 10.2.0.0 à R3

Mises à jour de routage différentes pour R1 et R3
- R1 annonce le réseau 10.1.0.0 à R2
- R3 annonce le réseau 10.4.0.0 à R2

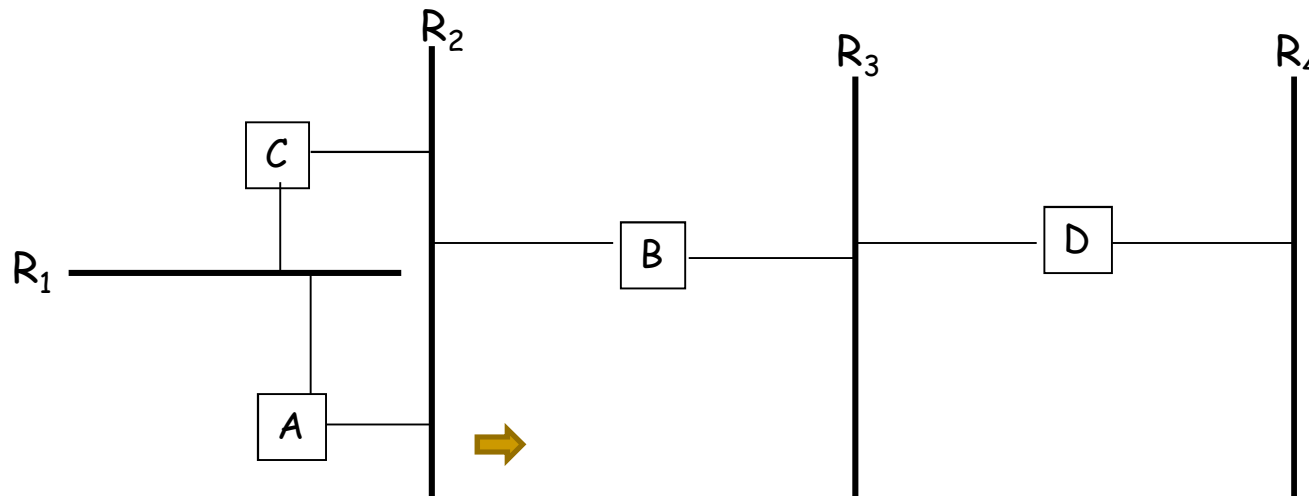
8.1.1. Règle de découpage d'horizon

- Principe de l'horizon coupé: Un routeur ne peut pas envoyer d'informations via l'interface d'où il les a reçues



8.1.1. Règle de découpage d'horizon

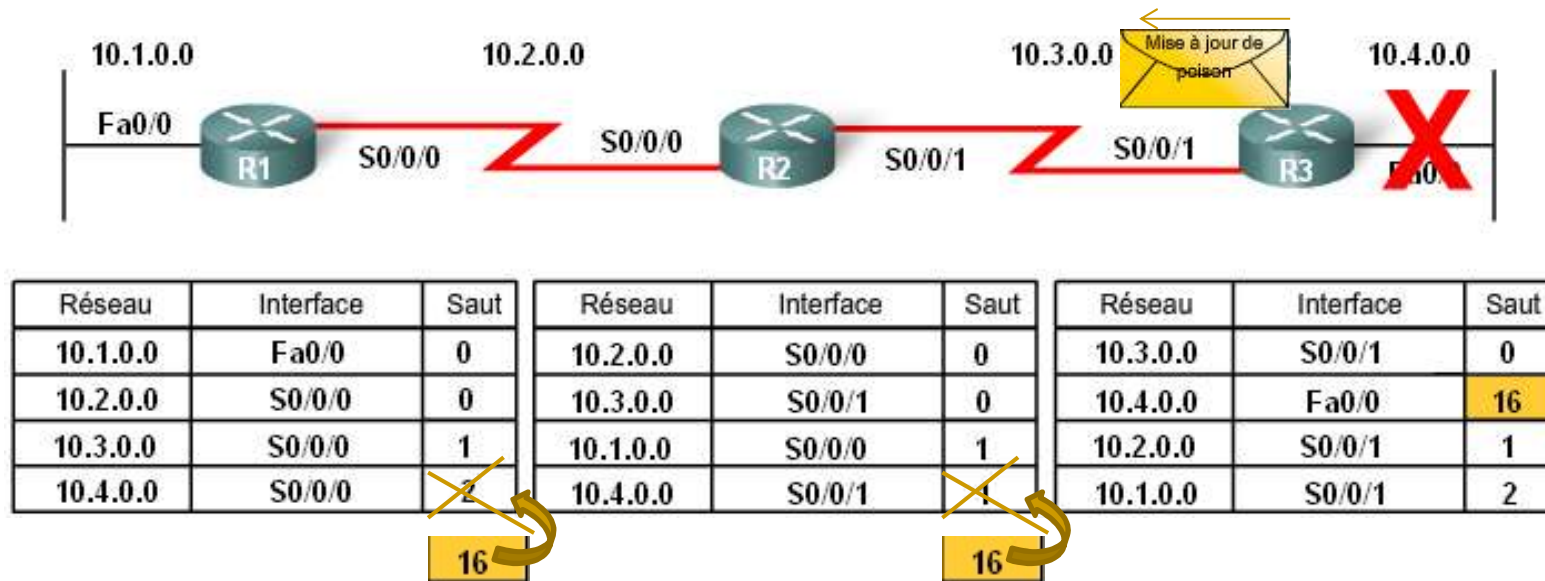
- Technique de l'horizon coupé limite les bouclages directs mais n'élimine pas complètement les boucles !!!!
- Ex:



- D devient inaccessible, B le détecte \Rightarrow met coût pour R_4 = infini
- Horizon coupé \Rightarrow A et C n'envoient pas d'info pour R_4 sur R_2
- C voit que A offre un coût de 3 pour joindre R_4 en comparaison avec B qui offre un coût infini \Rightarrow C modifie table et peut diffuser l'info sur R_2
- B croit que C peut joindre R_4 avec un coût de 4, etc.

8.1.2. Empoisonnement de routage

- Marquer la route comme étant inaccessible dans une mise à jour de routage envoyée à d'autres routeurs
 - Inaccessible ↔ mesure définie à sa valeur maximale
 - Pour RIP, route empoisonnée ↔ mesure = 16



Le réseau est convergé vers la route empoisonnée

- L'empoisonnement de routage accélère le processus de convergence

8.1.3. Découpage d'horizon avec empoisonnement inverse

- **Empoisonnement inverse + découpage d'horizon** ↔
 - Lors de l'envoi de mises à jour via une interface spécifique, tout réseau dont l'existence a été apprise sur cette interface est désigné comme étant inaccessible
- **Principe:** indiquer explicitement à un routeur d'ignorer une route et non lui cacher l'existence de la route
- L'empoisonnement inverse est une action spécifique qui annule le découpage d'horizon
- **Activé par défaut**
 - Peut ne pas être la valeur par défaut dans toutes les implémentations d'IOS

8.1.3. Découpage d'horizon avec empoisonnement inverse

- Exemple:

R3 envoie une mise à jour de poison déclenchée à R2.



Réseau	Interface	Saut
10.1.0.0	Fa0/0	0
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/0	1
10.4.0.0	S0/0/0	2

Réseau	Interface	Saut
10.2.0.0	S0/0/0	0
10.3.0.0	S0/0/1	0
10.1.0.0	S0/0/0	1
10.4.0.0	S0/0/1	1

Réseau	Interface	Saut
10.3.0.0	S0/0/1	0
10.4.0.0	Fa0/0	16
10.2.0.0	S0/0/1	1
10.1.0.0	S0/0/1	2

16

R2 « empoisonne » la route avec une mesure « infinie ».

R2 envoie un « empoisonnement inverse » à R3.

R3 non susceptible de recevoir des mises à jour incorrectes relatives au réseau 10.4.0.0



8.1.4. Le champs TTL de IP

- Durée de vie = champ de 8 bits dans l'en-tête IP ↔ limiter nombre de sauts qu'un paquet peut effectuer dans le réseau avant d'être supprimé
- **Objectif:**
 - Éviter circulation sans fin d'un paquet impossible à remettre
- Valeur définie par la source du paquet
 - Réduite de 1 par chaque routeur présent sur la route vers destination
 - Si TTL = zéro avant que arrivée paquet à destination ↔ paquet supprimé et envoi message d'erreur ICMP (*Internet Control Message Protocol*) à la source du paquet IP
- Même en présence d'une boucle de routage, les paquets ne circuleront pas dans une boucle sans fin sur le réseau



9. Exemples de protocoles vecteur distance

- Choix protocole à utiliser dans une situation donnée varie en fonction:
 - Taille du réseau
 - Compatibilité entre les modèles de routeurs
 - Connaissances administratives requises

	Ripv1	Ripv2	IGRP	EIGRP
Vitesse de convergence	Lent	Lent	Lent	Rapide
Évolutivité - Taille du réseau	Petit	Petit	Petit	Grand
Utilisation de VLSM	Non	Oui	Non	Oui
Utilisation de la ressource	Bas	Bas	Bas	Moyen
Implémentation et maintenance	Simple	Simple	Simple	Complexe

LES PROTOCOLES DE ROUTAGE A ETATS DES LIAISONS

1. Présentation

- Protocoles à vecteur de distance  utiliser panneaux de signalisation routière pour guider vers destination, en ne donnant que des informations sur distance et direction
 - Détermination chemins préférés en fonction de la distance ou mesure
 - Un routeur à vecteur de distance fait confiance à un autre routeur pour lui indiquer la distance réelle jusqu'au réseau de destination.
- Protocoles de routage d'état des liaisons  utiliser une carte routière pour voir toutes les routes potentielles et déterminer le chemin
 - Création carte topologique du réseau
 - Routeur utilise carte pour déterminer chemin le plus court vers chaque réseau
 - Envoi infos sur état liaisons vers d'autres routeurs du domaine de routage pour création propre carte topologique du réseau et calcul de façon indépendante le chemin vers chacun des réseaux

1. Présentation

- Protocoles de routage d'état de liaisons ont la réputation d'être beaucoup plus complexes que protocoles à vecteur de distance
- Protocoles de routage d'état de liaisons ont la réputation d'être beaucoup plus rapides que protocoles à vecteur de distance
- Protocoles de routage d'état de liaisons ont la réputation de ne pas causer:
 - De boucle de routage
 - Beaucoup de trafic de routage

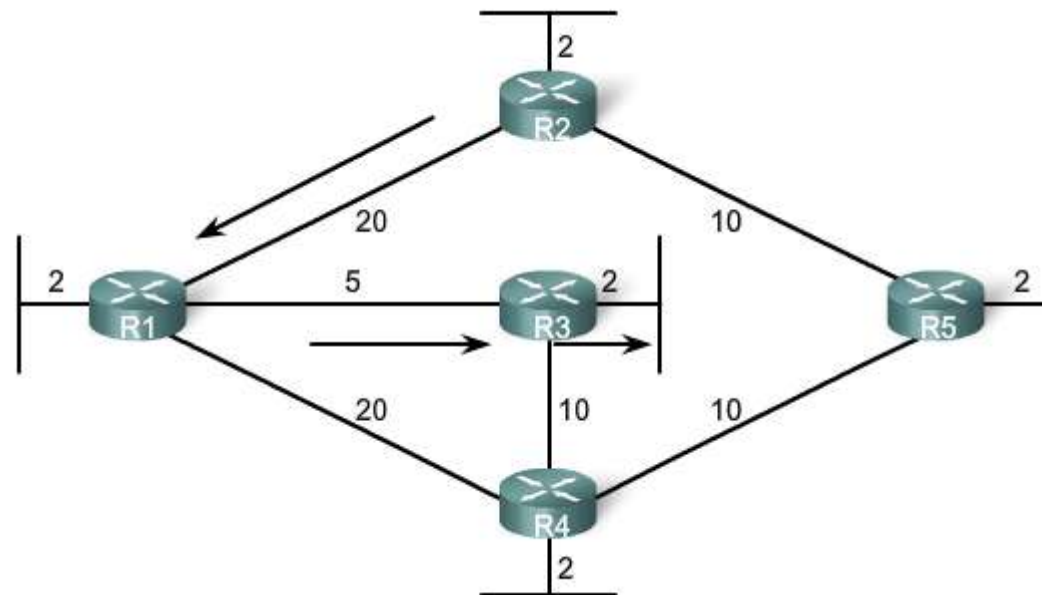
1. Présentation

	Protocoles IGP		Protocoles EGP
	Protocoles de routage à vecteur de distance	Protocoles de routage d'état des liaisons	Vecteur de chemin
Protocoles par classe	RIP IGRP		EGP
Protocoles sans classe	RIPv2 EIGRP	OSPFv2 IS-IS	BGPv4
Protocoles IPv6	RIPng EIGRP for IPv6	OSPFv3 IS-IS for IPv6	BGPv4 for IPv6

- Également appelés protocoles **SPF** (*Shortest Path First*), car conçus sur la base de l'algorithme SPF d'Edsger Dijkstra

2. Présentation de SPF

- SPF (shortest path first - chemin le plus court d'abord) = Dijkstra
 - Cumuler coûts de chaque chemin, depuis source jusqu'à leur destination



Chemin le plus court permettant à un hôte sur le réseau local (LAN) de R2 d'atteindre un hôte sur le réseau local de R3 :

$$\text{De R2 à R1 (20) + de R1 à R3 (5) + de R3 au LAN (2) = 27}$$

2. Présentation de SPF

- **Algorithme de Dijkstra (1959)**
 - Chaque nœud est étiqueté par sa distance au nœud source en suivant le meilleur chemin connu (à travers la base de données d'états des liaisons)
 - Validation du nœud source $I=S$
 - a/ calcul des valeurs de l'étiquette des nœuds adjacents à I non validés
 - b/ identification de J le nœud qui a la plus petite étiquette parmi tous les nœuds non validés, valider J
 - c/ si $J \neq D$ (destination), alors a/ avec $I=J$

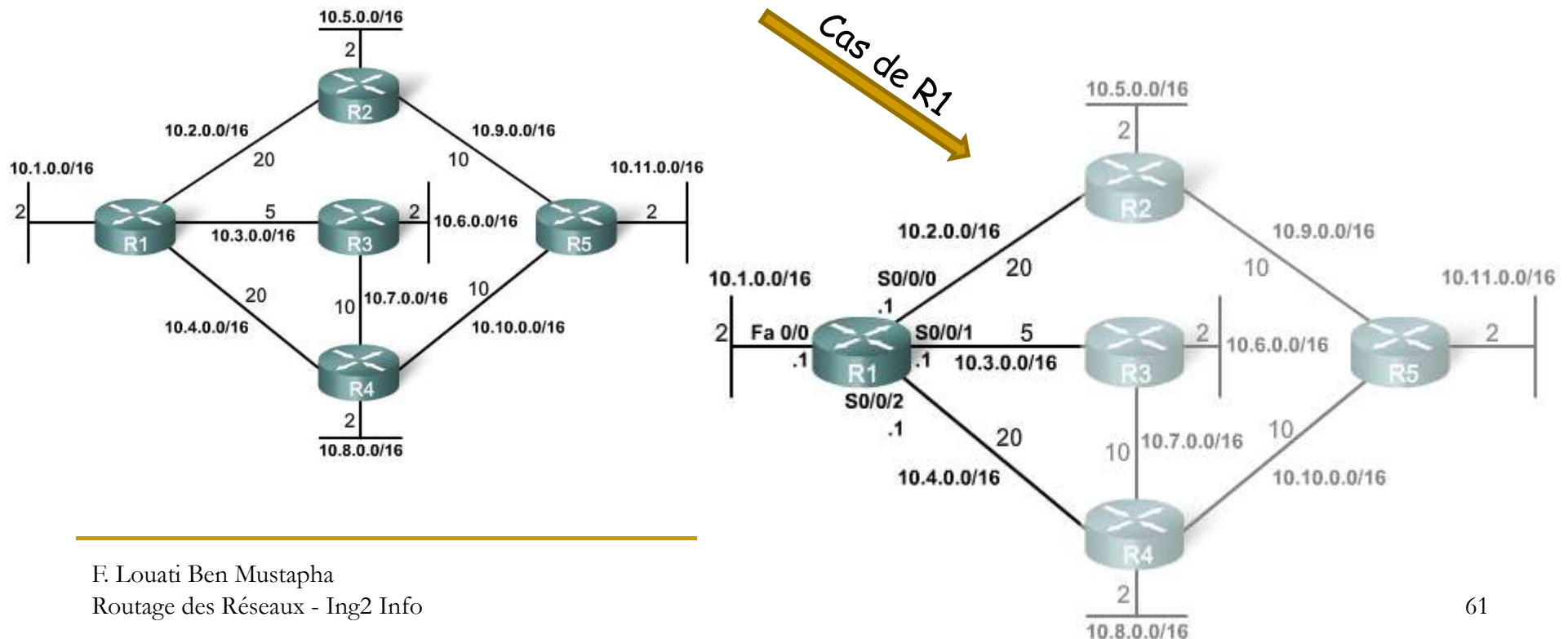
3. Présentation du processus de routage d'états des liens

■ Processus générique exécuté par routeurs:

1. Chaque routeur prend connaissance des réseaux directement connectés (interfaces à l'état actif)
2. Chaque routeur responsable de la détection de ses voisins sur les réseaux connectés directement ↔ échange paquets Hello avec routeurs d'état de liaisons situés sur des réseaux directement connectés
3. Chaque routeur crée un LSP (*Link-State Packet*) contenant état chaque liaison directement connectée ↔ enregistrement infos pertinentes sur chaque voisin (ID, type liaison, bande passante)
4. Chaque routeur diffuse LSP à ses voisins → stockage LSP dans base de données locale → diffusion par voisin LSP à leurs voisins, jusqu'à réception par tous les routeurs de la zone
5. Chaque routeur utilise base de données pour élaboration carte complète topologie et calcul meilleur chemin vers chaque réseau destination à l'aide de l'algorithme SPF

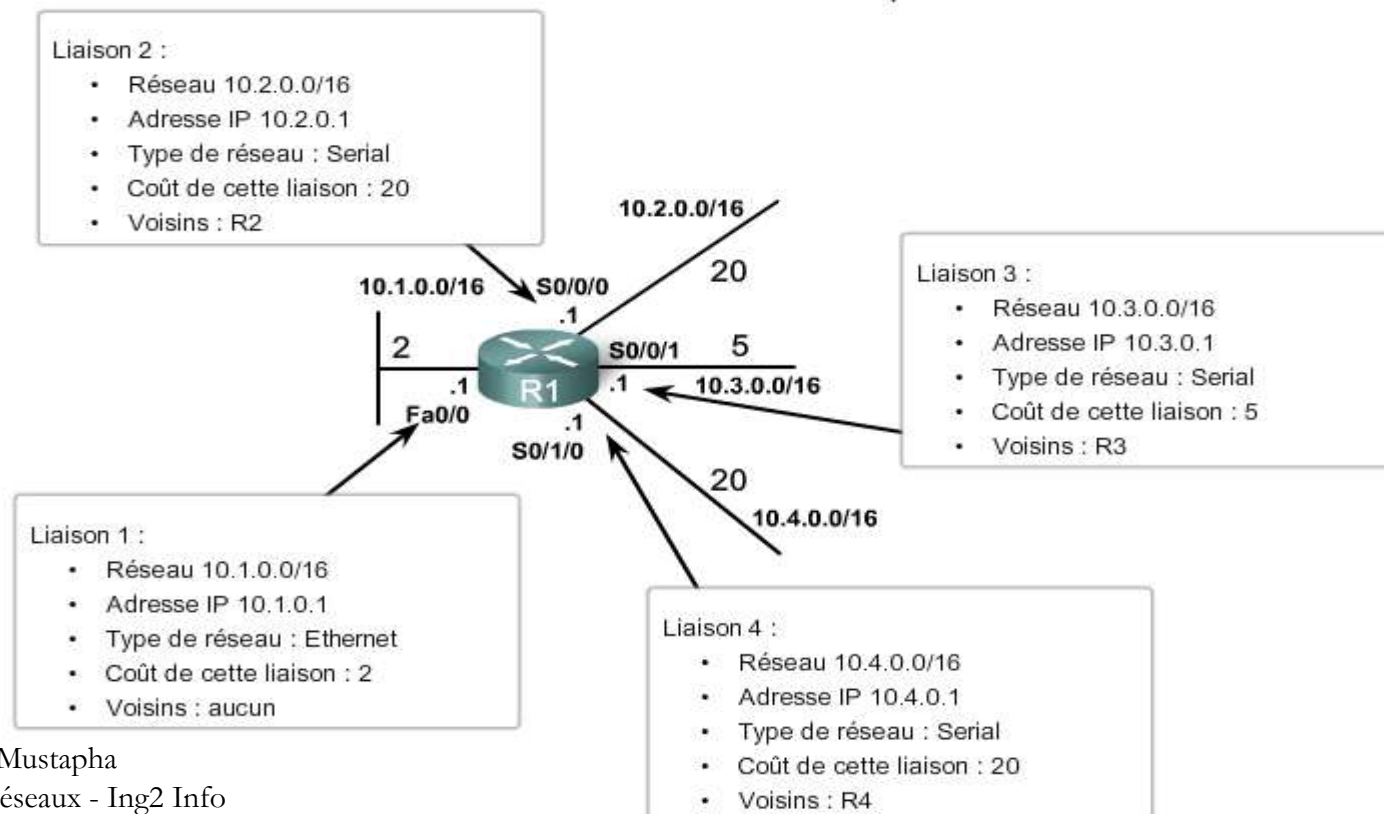
3.1. Etude des réseaux directement connectés

- Première étape du processus de routage d'état des liaisons
- Configuration interface de routeur avec @ IP et masque sous-réseau → interface ↔ élément du réseau, font partie de la table de routage



3.1. Etude des réseaux directement connectés

- Liaison ↔ interface de routeur incluse dans une des instructions `network` avant de pouvoir participer au processus d'état des liaisons
- État des liaisons ↔ informations relatives à l'état de ces liaisons

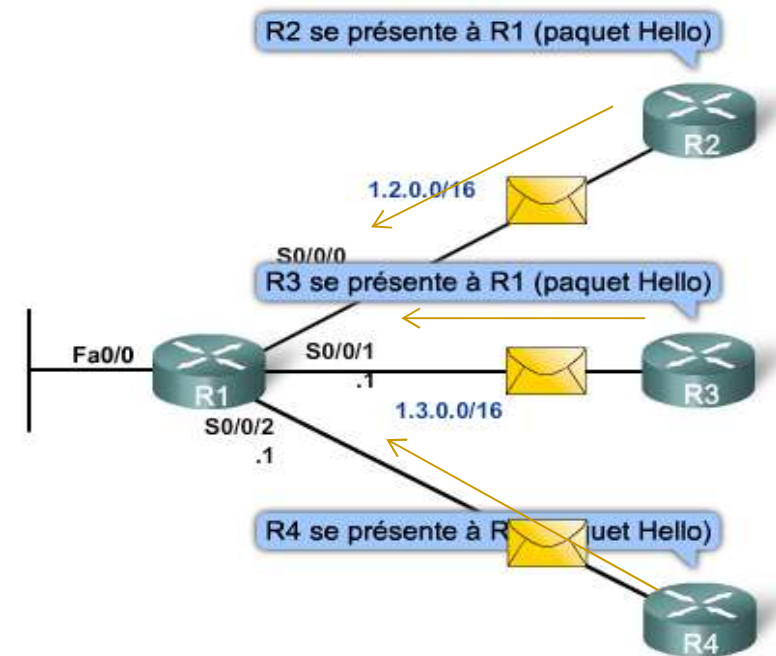
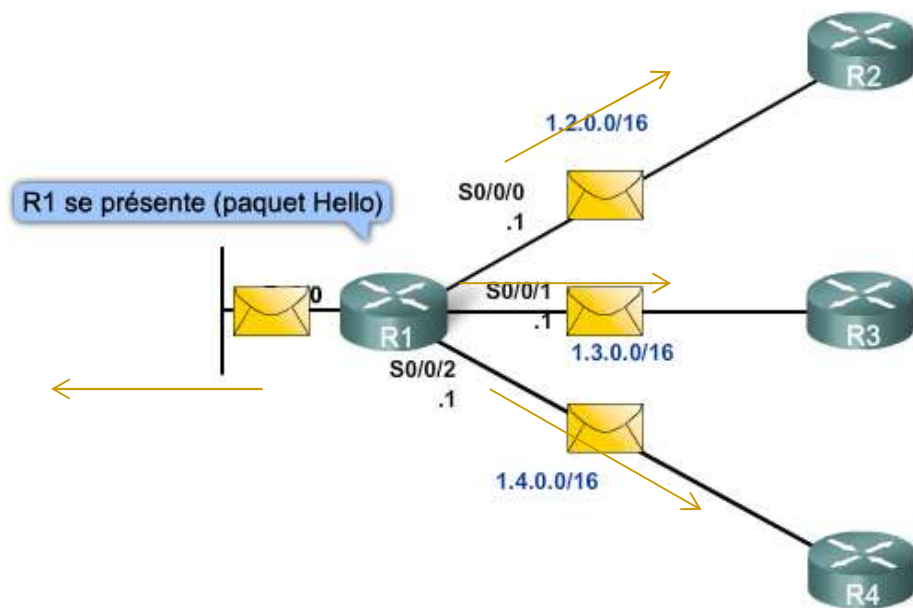


3.2. Envoi de paquets Hello

- Deuxième étape du processus de routage d'états des liaisons
- Envoi de paquet Hello aux voisins
 - Voisin = routeur prenant en charge le même protocole d'état des liaisons
- Deux routeurs d'états des liaisons apprennent qu'ils sont voisins ➡ formation de contiguïté (\cong EIGRP)
 - Continuation d'échange de paquets Hello entre deux voisins contigus pour assurer **fonction de « veille »** \longleftrightarrow surveiller état voisin:
 - Si cessation de réception de paquets Hello d'un voisin ➡ voisin considéré comme injoignable et interruption contiguïté



3.2. Envoi de paquets Hello

- Exemple:
 - R1 forme une contiguïté avec les trois routeurs



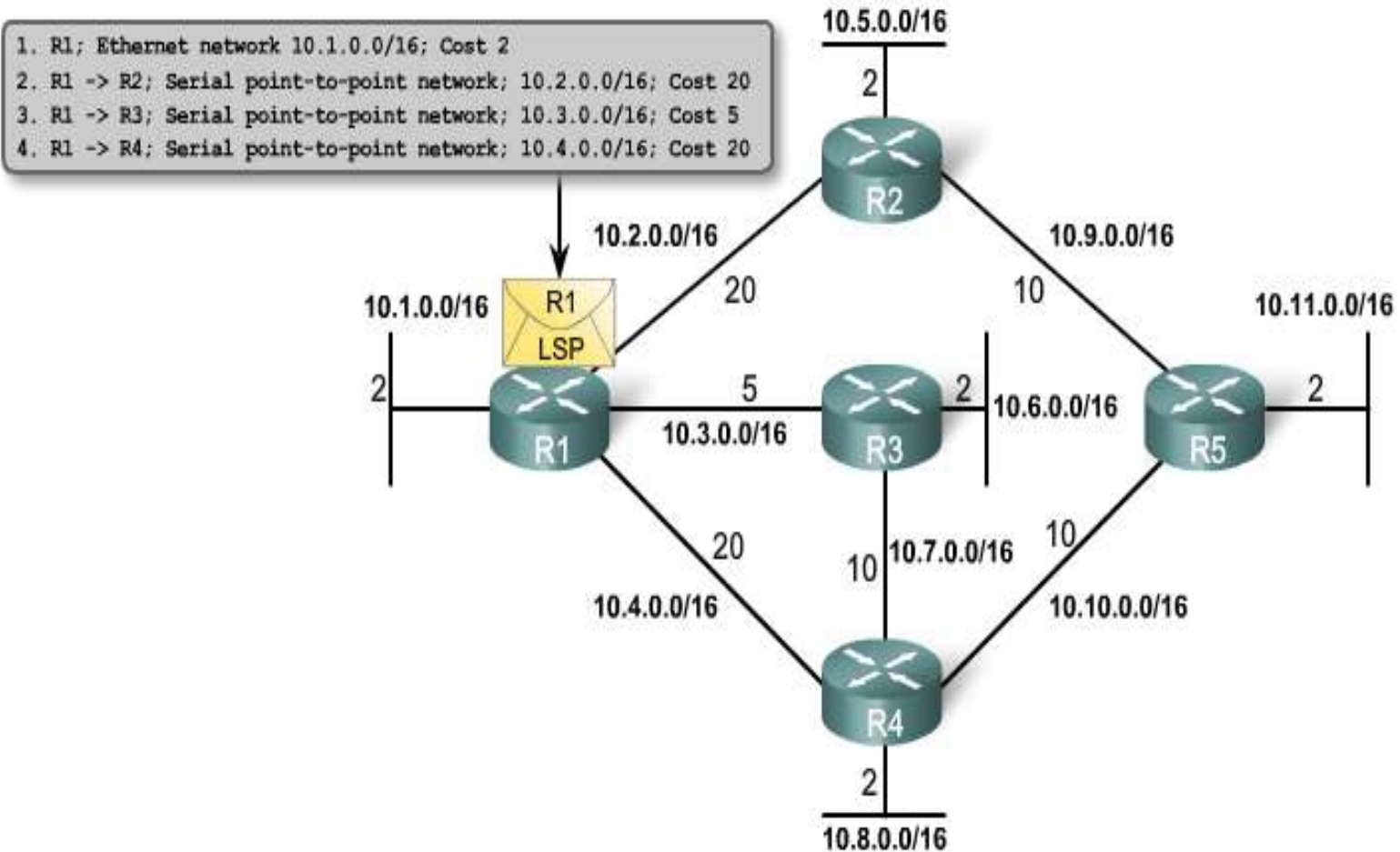
- Pas de voisin sur l'interface FastEthernet 0/0 ➡ pas de réception de paquet Hello sur cette interface ➡ R1 ne poursuit pas les étapes de processus de routage d'état des liaisons sur la liaison FastEthernet 0/0

3.3. Création du LSP

- **Troisième étape du processus de routage d'état des liaisons**
- Une fois qu'un routeur a établi des contiguités, il est en mesure de créer des paquets d'état des liaisons (LSP) relatifs à ses propres liaisons
- LSP (*Link-State Packet*)  contient état liaisons directement connectées
 - Enregistrement infos pertinentes sur chaque voisin (ID, type liaison, bande passante)
 - également d'autres informations, exemple les numéros de séquence et les données chronologiques  aide à la gestion du processus d'inondation
 - Informations utilisées par chaque routeur pour déterminer a déjà reçu LSP provenant d'un autre routeur ou si LSP contient de nouvelles informations, par rapport à la base de données d'état des liaisons
 - Conservation uniquement des informations à jour dans base de données d'état des liaisons

3.3. Création du LSP

- Version simplifiée des LSP de R1:



3.4. Diffusion des LSP aux voisins

- Quatrième étape du processus de routage
- Diffusion par chaque routeur des informations d'états des liaisons à l'ensemble des routeurs d'état des liaisons dans la zone de routage
- Réception LSP d'un routeur du voisinage ➡ envoi immédiat (sans calcul intermédiaire) par récepteur à toutes les autres interfaces, à part à celle par laquelle qui l'a reçu ➡ **diffusion de LSP** à partir des routeurs de la zone de routage
 - Protocoles de routage à vecteur de distance ↔ exécution algorithme Bellman-Ford pour traiter mises à jour de routage avant de les envoyer aux autres routeurs
 - Protocoles de routage d'état des liaisons ↔ calcul algorithme SPF après que la diffusion soit complète ➡ convergence plus rapide

3.4. Diffusion des LSP aux voisins

- LSP ↔ pas envoi de façon périodique
- Envoyé uniquement :
 - lors du démarrage initial du routeur ou du lancement du processus de protocole de routage sur ce routeur ;
 - Lors d'une modification de topologie, notamment en cas d'activation ou de désactivation d'une liaison ou d'établissement ou de rupture d'une contiguïté entre voisins

3.5. Création BD d'états des liaisons

- Étape finale du processus de routage d'état des liaisons
- Diffusion LSP via le processus d'inondation d'état de liaisons ↔
réception par chaque routeur de LSP provenant de tous les routeurs d'état de liaisons de la zone de routage
 - LSP stockés dans **base de données** d'état des liaisons
 - Utilisation de l'algorithme SPF pour construire arborescence SPF

3.5. Création BD d'états des liaisons

- Exemple: base de données d'état des liaisons de R1



Base de données d'état des liaisons de R1

Paquets LSP de R2 :

- Connecté au voisin R1 sur le réseau 10.2.0.0/16, coût : 20
- Connecté au voisin R5 sur le réseau 10.9.0.0/16, coût : 10
- Possède un réseau 10.5.0.0/16, coût : 2

Paquets LSP de R3 :

- Connecté au voisin R1 sur le réseau 10.3.0.0/16, coût : 5
- Connecté au voisin R4 sur le réseau 10.7.0.0/16, coût : 10
- Possède un réseau 10.6.0.0/16, coût : 2

Paquets LSP de R4 :

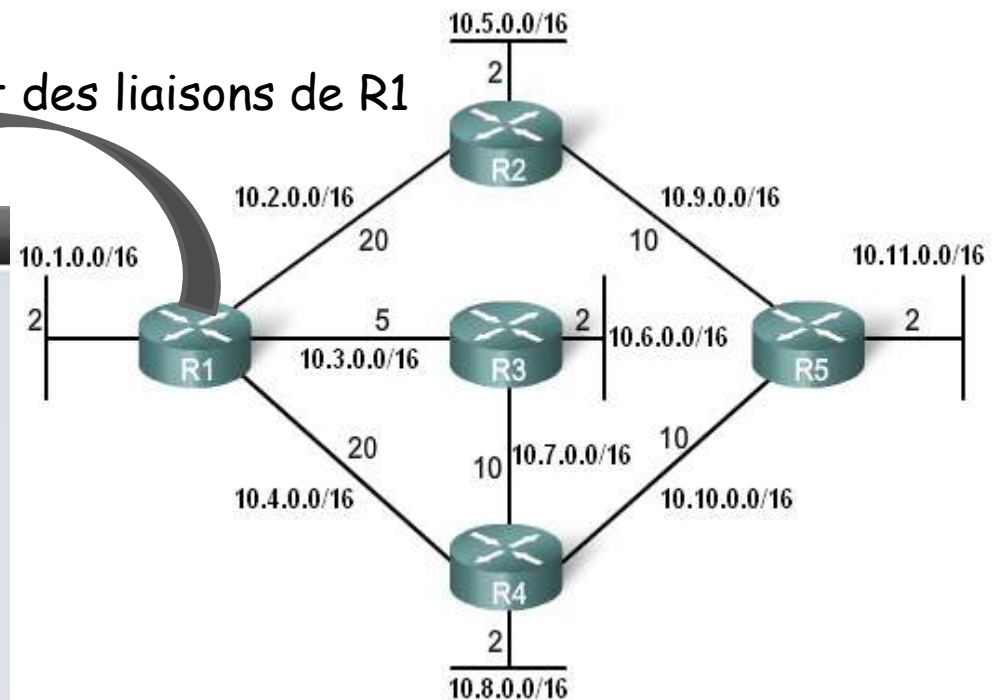
- Connecté au voisin R1 sur le réseau 10.4.0.0/16, coût : 20
- Connecté au voisin R3 sur le réseau 10.7.0.0/16, coût : 10
- Connecté au voisin R5 sur le réseau 10.10.0.0/16, coût : 10
- Possède un réseau 10.8.0.0/16, coût : 2

Paquets LSP de R5 :

- Connecté au voisin R2 sur le réseau 10.9.0.0/16, coût : 10
- Connecté au voisin R4 sur le réseau 10.10.0.0/16, coût : 10
- Possède un réseau 10.11.0.0/16, coût : 2

États des liaisons de R1 :

- Connecté au voisin R2 sur le réseau 10.2.0.0/16, coût : 20
- Connecté au voisin R3 sur le réseau 10.3.0.0/16, coût : 5
- Connecté au voisin R4 sur le réseau 10.4.0.0/16, coût : 20
- Possède un réseau 10.1.0.0/16, coût : 2

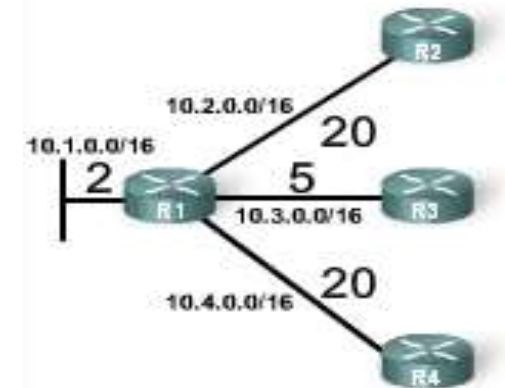


3.6. Arborescence SFP

- **Exemple:** élaboration arborescence SFP de R1
 - R1 se place à la racine de l'arborescence
 - Liaisons directement connectées déjà connues

États des liaisons de R1 :

- Connecté au voisin R2 sur le réseau 10.2.0.0/16, coût : 20
- Connecté au voisin R3 sur le réseau 10.3.0.0/16, coût : 5
- Connecté au voisin R4 sur le réseau 10.4.0.0/16, coût : 20
- Possède un réseau 10.1.0.0/16, coût : 2



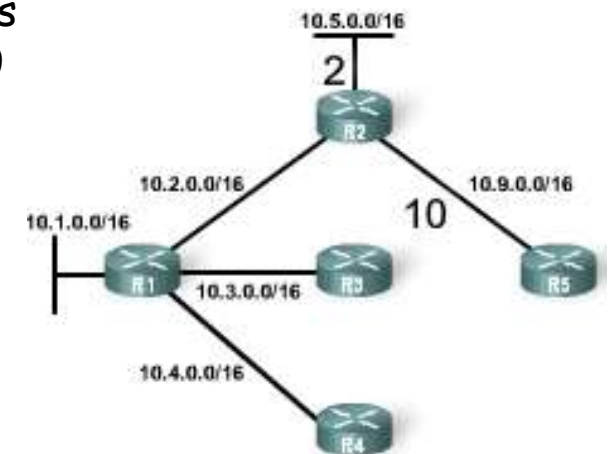
3.6. Arborescence SFP

- **Exemple:** élaboration arborescence SFP de R1
- Traitement des infos de liaisons suivantes de R2 ➡

Paquets LSP de R2 :

- Connecté au voisin R1 sur le réseau 10.2.0.0/16, coût : 20
- Connecté au voisin R5 sur le réseau 10.9.0.0/16, coût : 10
- Possède un réseau 10.5.0.0/16, coût : 2

- Utilisation 2nd LSP, création lien depuis R2 vers R5, avec le réseau 10.9.0.0/16 et un coût de 10
➡ info ajoutée l'arborescence SPF
- Utilisation 3^{ème} LSP, R1 apprend que R2 possède un réseau 10.5.0.0/16, avec coût de 2, sans voisin ➡ liaison ajoutée à l'arborescence SPF de R1



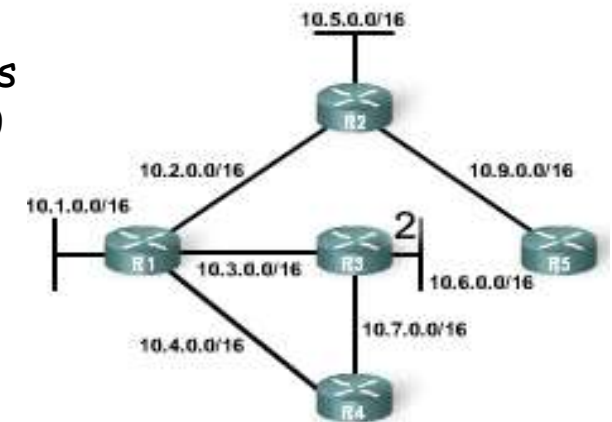
3.6. Arborescence SFP

- **Exemple:** élaboration arborescence SFP de R1
- Traitement des infos de liaisons suivantes de R3 ➡

Paquets LSP de R3 :

- Connecté au voisin R1 sur le réseau 10.3.0.0/16, coût : 5
- Connecté au voisin R4 sur le réseau 10.7.0.0/16, coût : 10
- Possède un réseau 10.6.0.0/16, coût : 2

- Utilisation 2nd LSP, création lien depuis R3 vers R4, avec le réseau 10.7.0.0/16 et un coût de 10
➡ info ajoutée l'arborescence SPF
- Utilisation 3^{ème} LSP, R1 apprend que R3 possède un réseau 10.6.0.0/16, avec coût de 2, sans voisin ➡ liaison ajoutée à l'arborescence SPF de R1



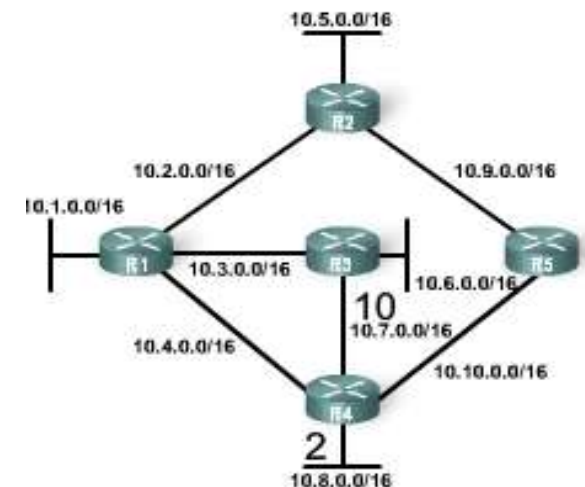
3.6. Arborescence SFP

- **Exemple:** élaboration arborescence SFP de R1
- Traitement des infos de liaisons suivantes de R4 ➡

Paquets LSP de R4 :

- Connecté au voisin R1 sur le réseau 10.4.0.0/16, coût : 20
- Connecté au voisin R3 sur le réseau 10.7.0.0/16, coût : 10
- Connecté au voisin R5 sur le réseau 10.10.0.0/16, coût : 10
- Possède un réseau 10.8.0.0/16, coût : 2

- Ignorer 1^{er} et 2nd LSP
- Utilisation 3^{ème} LSP, création lien depuis R4 vers R5, avec le réseau 10.10.0.0/16 et un coût de 10 ➡ info ajoutée l'arborescence SPF
- Utilisation 4^{ème} LSP, R1 apprend que R4 possède un réseau 10.8.0.0/16, avec coût de 2, sans voisin ➡ liaison ajoutée à l'arborescence SPF de R1



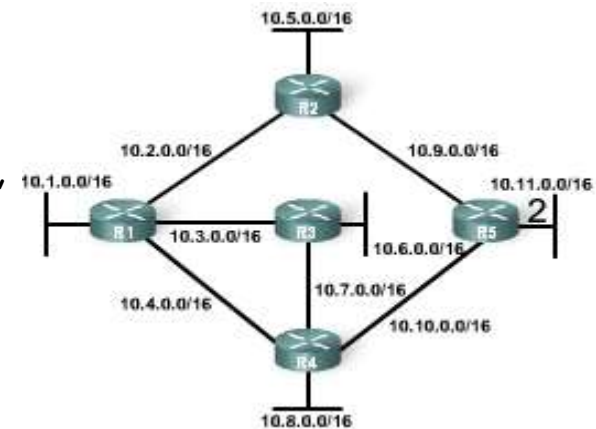
3.6. Arborescence SFP

- **Exemple:** élaboration arborescence SFP de R1
- Traitement des infos de liaisons suivantes de R5 ➡

Paquets LSP de R5 :

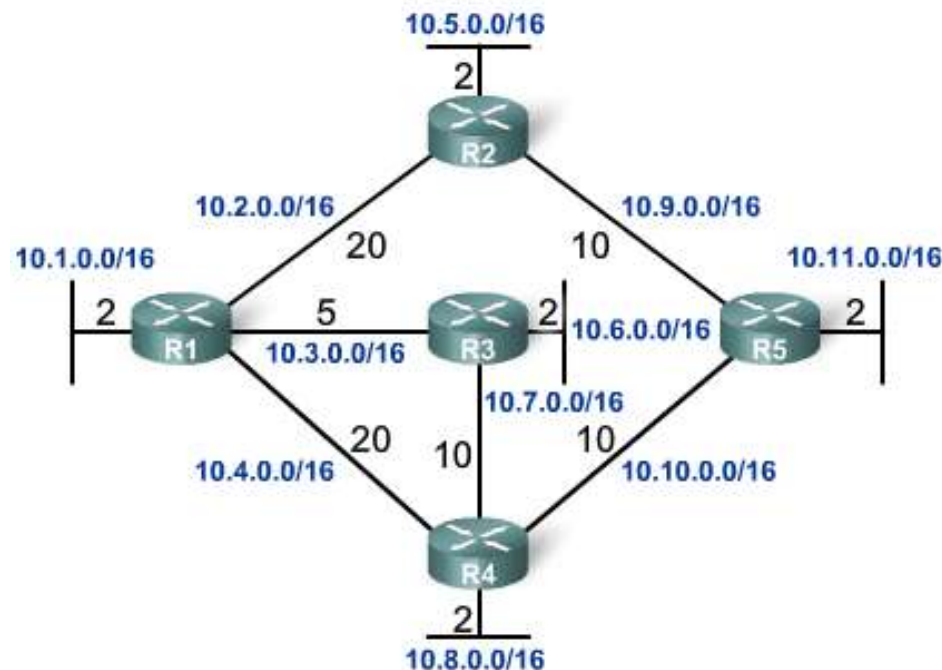
- Connecté au voisin R2 sur le réseau 10.9.0.0/16, coût : 10
- Connecté au voisin R4 sur le réseau 10.10.0.0/16, coût : 10
- Possède un réseau 10.11.0.0/16, coût : 2

- Ignorer 1^{er} et 2nd LSP
- Utilisation 3^{ème} LSP, R1 apprend que R5 possède un réseau 10.11.0.0/16, avec coût de 2, sans voisin ➡ liaison ajoutée à l'arborescence SPF de R1



3.6. Arborescence SFP

- **Exemple:** Arborescence SFP de R1 ➡



- Construction table de routage en utilisant l'algorithme **Shortest First Path** sur l'arborescence SFP pour trouver le plus court chemin de R1 vers tous les autres routeurs
 - Construction table en même temps que l'arborescence

3.6. Arborescence SFP

■ Exemple: Construction table de routage de R1 ➡

Table de routage pour R1

Réseaux directement connectés

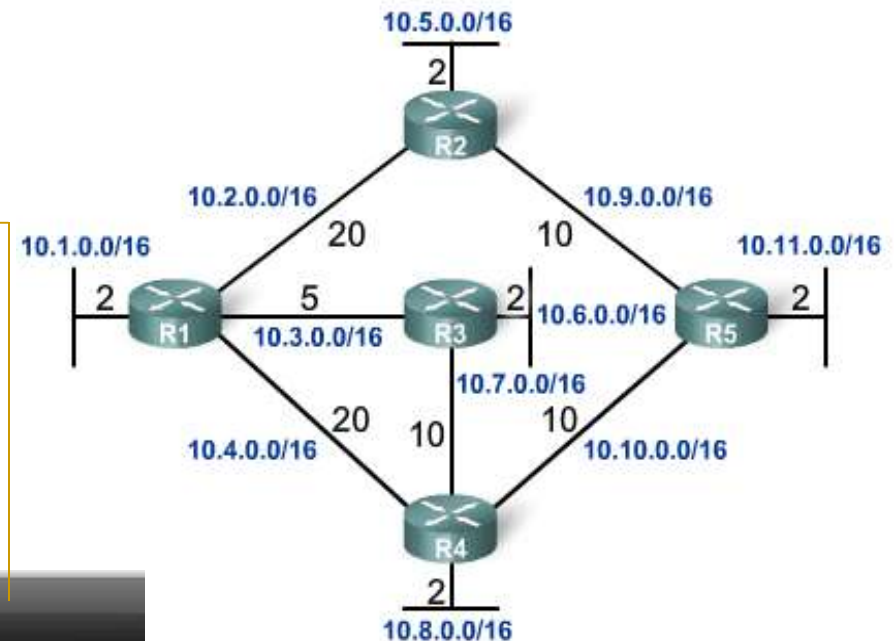
- Réseau directement connecté 10.1.0.0/16
- Réseau directement connecté 10.2.0.0/16
- Réseau directement connecté 10.3.0.0/16
- Réseau directement connecté 10.4.0.0/16

Réseaux distants

- 10.5.0.0/16 via interface série de R2 0/0/0, coût = 22
- 10.6.0.0/16 via interface série de R3 0/0/1, coût = 7
- 10.7.0.0/16 via interface série de R3 0/0/1, coût = 15
- 10.8.0.0/16 via interface série de R3 0/0/1, coût = 17
- 10.9.0.0/16 via interface série de R2 0/0/0, coût = 30
- 10.10.0.0/16 via interface série de R3 0/0/1, coût = 25
- 10.11.0.0/16 via interface série de R3 0/0/1, coût = 27

Informations de paquet SPF

- Réseau 10.5.0.0/16 via interface série de R2 0/0/0 à un coût de 22
- Réseau 10.6.0.0/16 via interface série de R3 0/0/1 à un coût de 7
- Réseau 10.7.0.0/16 via interface série de R3 0/0/1 à un coût de 15
- Réseau 10.8.0.0/16 via interface série de R3 0/0/1 à un coût de 17
- Réseau 10.9.0.0/16 via interface série de R2 0/0/0 à un coût de 30
- Réseau 10.10.0.0/16 via interface série de R3 0/0/1 à un coût de 25
- Réseau 10.11.0.0/16 via interface série de 0/0/1 à un coût de 27



4. Mise en œuvre

■ Avantages protocoles d'états des liaisons ➡

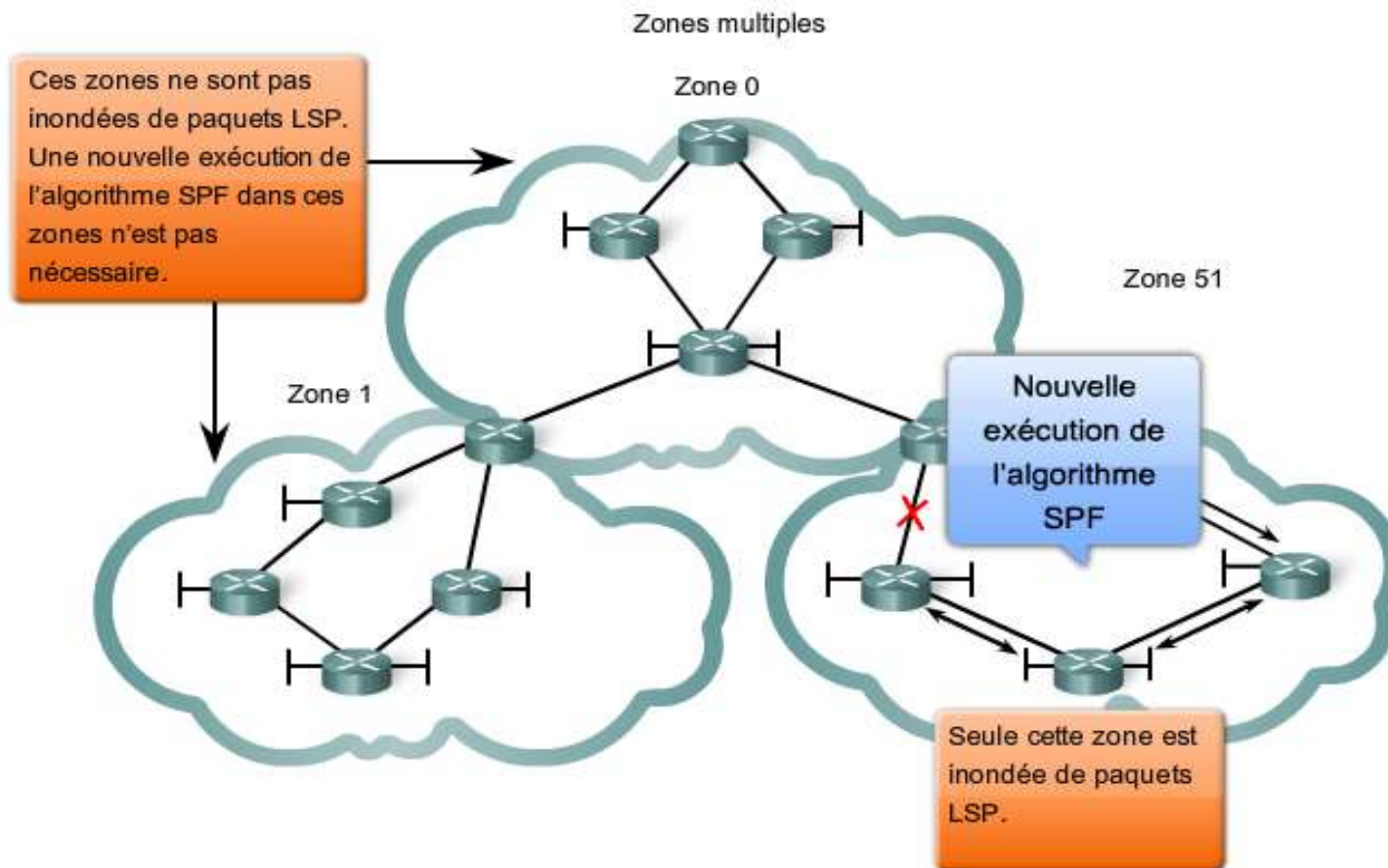
- Chaque routeur crée sa propre carte topologique du réseau pour déterminer le chemin le plus court.
- L'inondation immédiate de paquets LSP permet une convergence plus rapide.
- Les paquets LSP ne sont envoyés qu'en cas de modification de la topologie et contiennent uniquement les informations concernant cette modification.
- Une conception hiérarchique est utilisée lors de l'implémentation de plusieurs zones.

4.1. Définition de zones

- Objectif protocoles de routage d'état des liaisons modernes = minimiser effets sur mémoire, unité centrale et bande passante
- Utilisation et configuration de plusieurs zones ↔ réduction taille bases de données d'états des liaisons
 - Limiter quantité de données d'état des liaisons diffusées sur un domaine
 - Envoi LSP uniquement aux routeurs qui en ont besoin
- **Exemple:** modification topologie → réception LSP par uniquement les routeurs de la zone concernée et exécution algo SPF → isolement liaison instable dans zone spécifique du domaine de routage

4.1. Définition de zones

- Exemple:



4.2. éléments requis

- **Mémoire requise**

- Besoin de plus de mémoire, de temps processeur et parfois, davantage de bande passante que les protocoles de routage à vecteur de distance
 - Utilisation des bases de données d'état des liaisons et création de l'arborescence SPF

- **Temps processeur requis**

- Besoin de davantage de temps processeur que protocoles à vecteur de distance
 - Algorithme SPF, construction carte topologique complète

- **Bande passante requise**

- Diffusion paquets d'état liaisons ➡ effet sur bande passante disponible
 - Surtout pendant démarrage initial routeurs, problématique sur réseaux instables

4.3. Principaux protocoles

- **OSPF**

- Chapitre 5
- Conçu par le groupe de travail OSPF de l'IETF
- Le développement du protocole OSPF a commencé en 1987 ; il en existe aujourd'hui deux versions :
 - OSPFv2 : OSPF pour les réseaux IPv4 (RFC 1247 et RFC 2328)
 - OSPFv3 : OSPF pour les réseaux IPv6 (RFC 2740)

- **IS-IS**

- Conçu par ISO (International Organization for Standardization - Organisme international de normalisation)
- Décrit dans la norme ISO 10589
- Première application pratique développée chez DEC (Digital Equipment Corporation)
- Également connu sous le nom de DECnet Phase V
- Radia Perlman a été le principal concepteur du protocole de routage IS-IS.