Cours RSX103 - Chapitre 05

Routage BGP

Routage Inter-domaine vs Routage Intra-domaine

- **Routage IntraDomaine (IGP)**
  - **Objectif**
    - Choisir le **meilleur chemin** vers chaque destination, en respectant une **métrique donnée** (par exemple le retard, la bande passante) à l'intérieur des systèmes autonomes (AS)
  - **Problèmes à prendre en compte**
    - L'IGP devrait réagir **rapidement** aux **changements** de topologie
- **Routage Inter-domaine(EGP)**
  - **Objectif :**
    - Choisir un **meilleur chemin** vers chaque destination et qui est compatible avec **les politiques de routage des As de transit** sans connaître la topologie du passage ces AS
  - **Problèmes à prendre en compte :**
    - Chaque AS peut **définir ses propres politiques** de routage (⇔ permettre à chaque SA de choisir sa propre politique de routage), (Share Cost Peering, Customer-Provider)
    - L'EGP devrait **être scalable** (exemple : 13 000 AS, 120 000 chemins)

Pourquoi a-t-on besoin d'un EGP ?

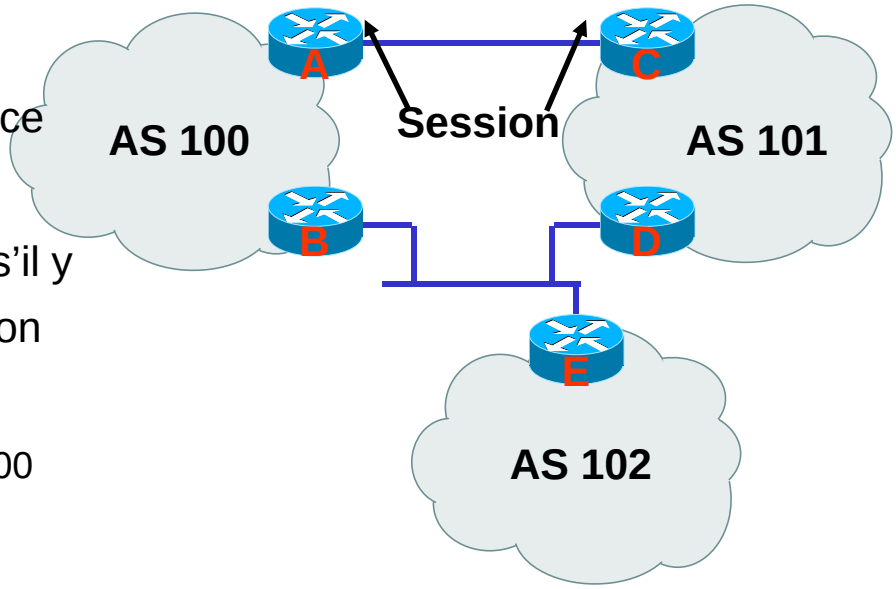
Pour quoi a-t-on besoin d'un EGP ?

- **S'adapter à un réseau de grande taille**
  - ✓ hiérarchie
  - ✓ limiter la portée des pannes
  - ✓ La table de routage doit avoir une entrée pour chaque adresse globale dans le monde (l'Internet a des centaines de milliers de routes : 50 000 routes à l'intérieur des routeur « au début de l'an 2000 »)
- **Définir des limites administratives**
- **Routage politique**
  - ✓ Les IGP n'ont pas de support pour ***des politiques de routage***
  - ✓ BGP permet de contrôler ***l'accessibilité des préfixes*** (routes)
    - ✓ Les routes sont identifiées avec *un préfixe/ longueur de préfixe* **CIDR**
      - ✓ Appelé **NRLI** (Network Layer Reachibility Information)

BGP - Principes de base

BGP : Principes de base -1-

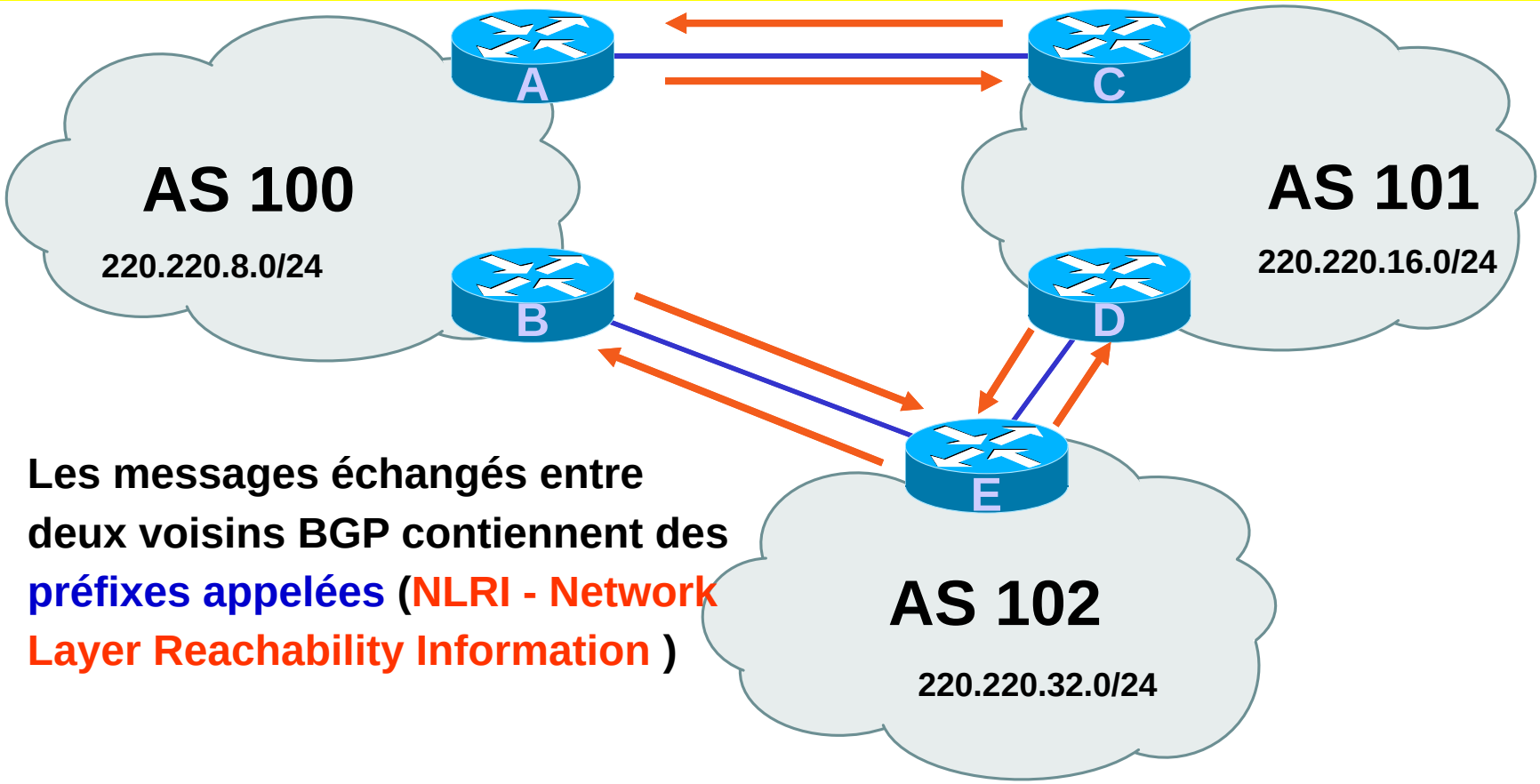
- **BGP est utilisé entre AS's (⇔ si vous n'êtes raccordé qu'à un seul AS vous n'avez pas besoin de BGP)**
- **L'échange des tables entre deux routeurs BGP voisins via une connexion peering (établie au-dessus de TCP (⇔ échange fiable des tables de routages)**
  - **(Une session BGP est établie au-dessus du protocole TCP (N° de port 179))**
- **Une fois la connexion est établie, chaque routeur va annoncer ses tables de façon particulière**
  - Dans un premier temps, chaque routeur annonce l'entièreté de **sa table de routage**
  - Ensuite, on ne va ré annoncer des routes que s'il y a des **routes qui ont changé** (annonce de façon incrémentale.
    - À l'opposé de RIP (⇔ La table de BGP # 120 000 routes)



BGP : Principes de base -2-

- **BGP permet de distribuer les routes** de façon scalable **tout en supportant des** politiques de routage, **pour cela**
  - protocole du **type vecteur de distance** (⇔ chaque routeur va devoir annoncer le contenu de sa table de routage ***en plus le chemin complet par lequel*** un paquet est passé)
  - **Deux Avantages à ça:**
    1. **permet d'éviter les problèmes du vecteur de distance** (possible de détecter les boucles lorsque l'on connaît le chemin complet)
    2. **permet de connaître la liste des** AS **par lesquels un paquet passera avant d'arriver à la destination** (nécessaire pour supporter ***du routage politique***)
- **Un autre élément, c'est que un routeur BGP lorsqu'il a reçu des routes, il ne va pas les accepter comme telle quelles).**
  - Utilisation **des filtres** pour décider qu'elles sont les routes qui **sont acceptable**, et quelles sont les routes qui sont considérées comme **meilleurs** parmi les routes qu'il reçoit.
  - Un routeur BGP, ne va pas annoncer toutes les routes qu'il connaît, il pourra annoncer à son voisin un sous-ensemble des routes qu'il connaît de façon à implémenter des **politiques de routage « via les attributs des routes »**

BGP : Principes de base -3-



- **Les messages échangés entre deux voisins BGP contiennent des préfixes appelées (NLRI - Network Layer Reachability Information)**

- **Chaque AS est le point de départ d'un ensemble de préfixes (NLRI)**

  - **NLRI = Network Layer Reachability Information** = Préfixes
  - Permet d'annoncer l'accessibilité d'une route
  - Composé des informations suivantes :
    - Préfixe réseau
    - Longueur du masque

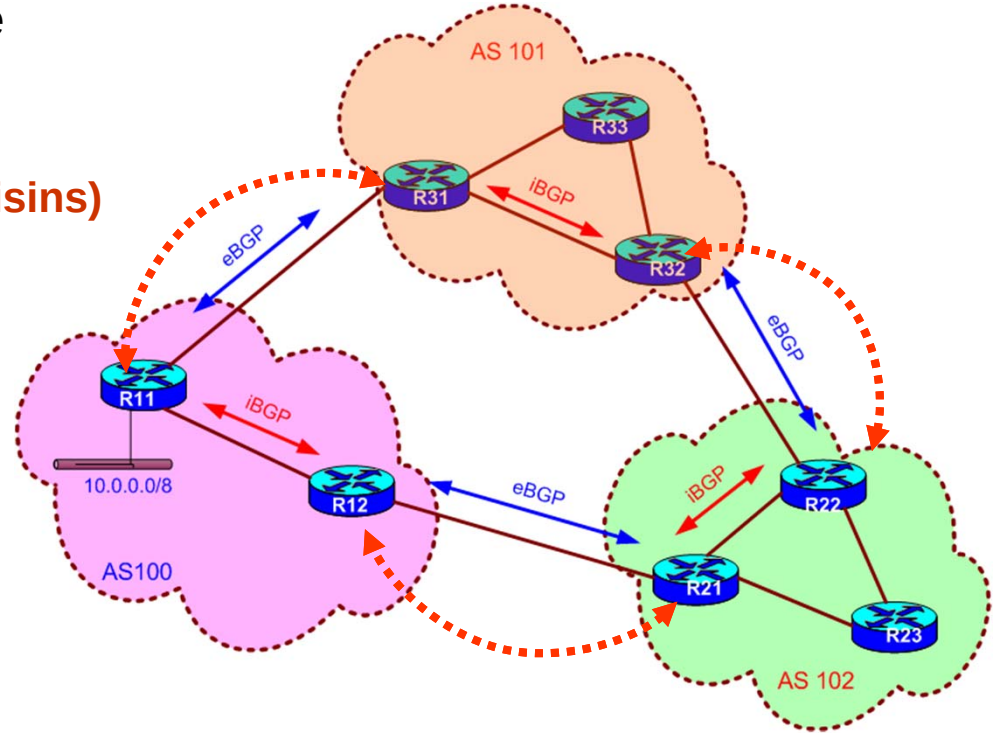
Les deux variantes de BGP

✪ **Au niveau de BGP, il existe deux variantes de BGP:**

1. **Session eBGP :utilisé entre deux routeurs BGP qui font parti de deux SA différents**
   - **Exemple : R11** et **R31** vont être reliés par une session BGP au-dessus de la liaison peering entre les routeurs **R11** et **R31**
   - Cette session servira pour annoncer les routes qui sont connus de AS100 vers AS 101 par exemple

Routeurs BGP appelés peers (voisins)

Session entre 2 AS différents = External BGP

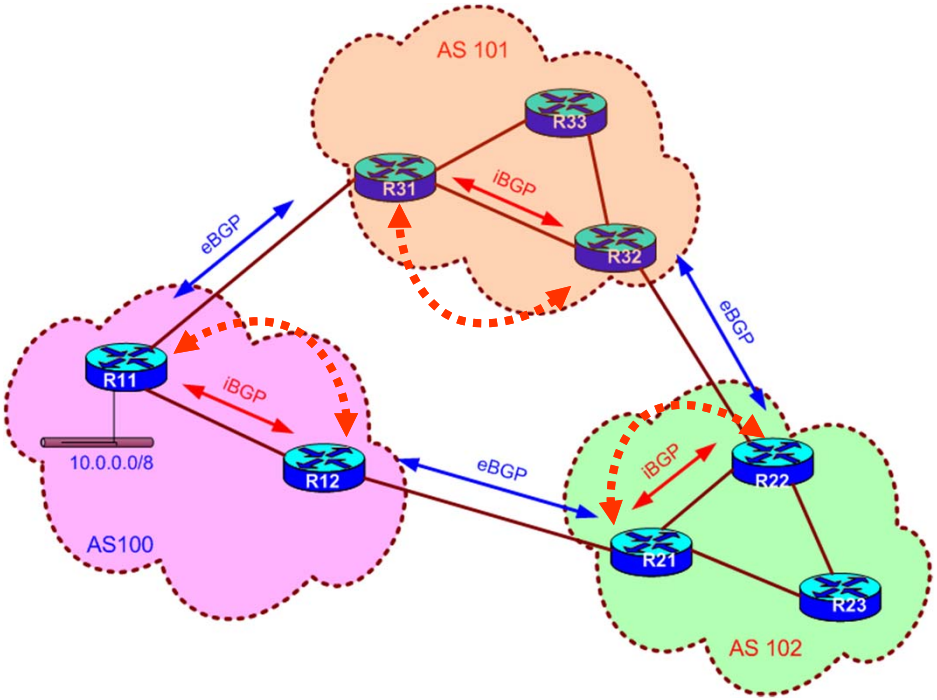


Connexion TCP/IP
eBG-

2. Sessions iBGP : qui vont être activées à l'intérieur d'un SA :

- Tous les routeurs à l'intérieur d'un SA qui implémentent BGP vont être interconnectés eux via des sessions **iBGP.**
- un **full-mesh BGP** entre tous les routeurs BGP à l'intérieur d'un même SA pour que tout routeur BGP soit au courant de **toutes les routes annoncées** par les autres routeurs BGP.

Les voisins d'un même AS sont appelés des voisins internes (internal peers)



Connexion TCP/IP
iBGP

Interactions BGP-OSPF

Interactions BGP-OSPF-1-

- Le routage est organisé de façon **hiérarchique** entre le routage intra-domaine contrôlé par **OSPF** et le routage inter-domaine contrôlé par **BGP.**

- Pour que le routage fonctionne correctement, il faut qu'il y ait des interactions entre ces deux protocoles :
  - **BGP** se charge de distribuer la topologie globale à l'échelle de l'Internet
  - **OSPF** se charge de distribuer la topologie interne du réseau

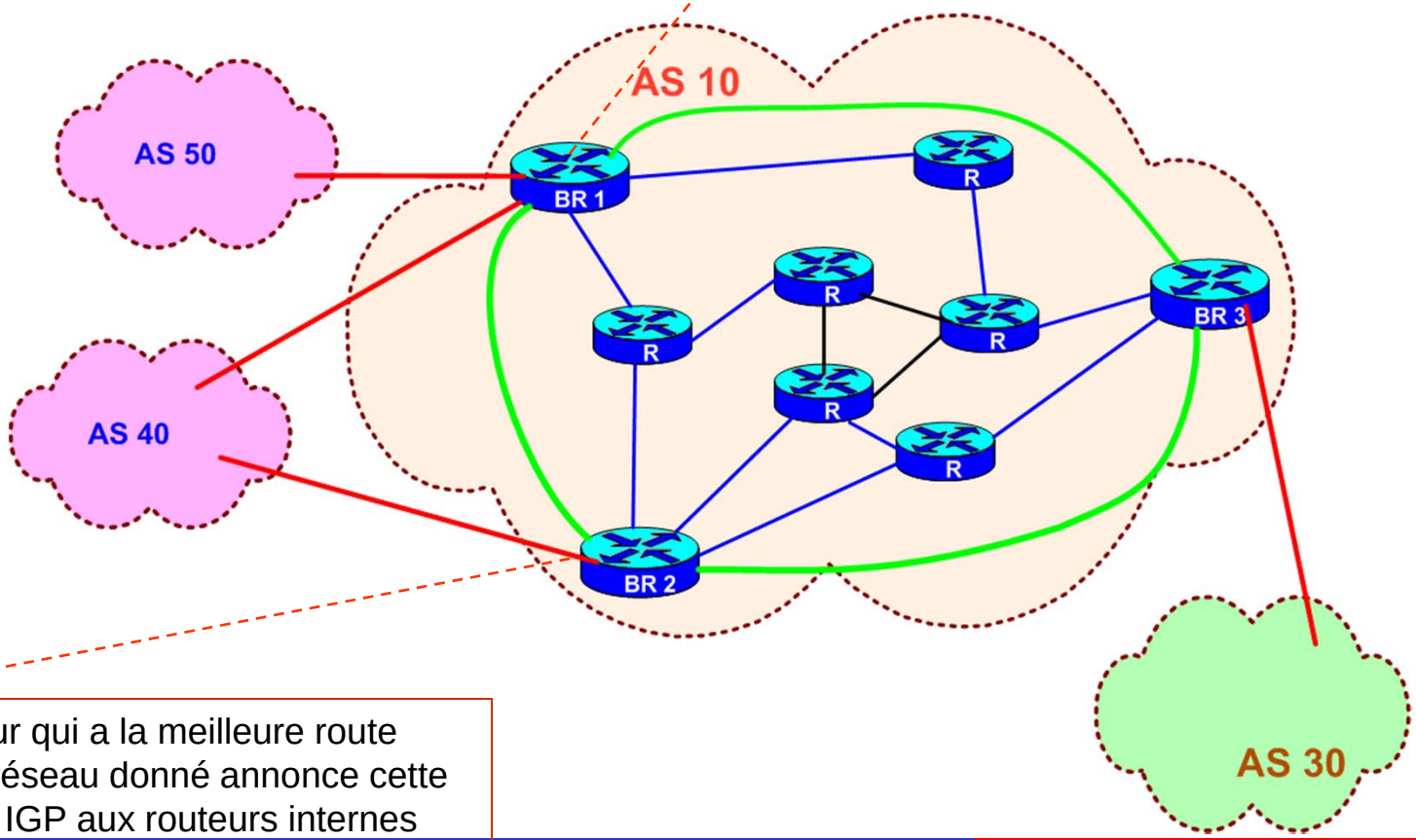
- Si on est en face à un **SA** qui dispose de **plusieurs connexions vers l'extérieur**, il faudra qu'on puisse choisir entre les différents routeurs qui se trouvent à la frontière du SA
  - ce choix va se faire à l'intérieur du **SA** lui-même

- Pour ça il faut qu'on ait deux mécanismes :
  1. Un qui permet aux **routeurs qui se trouvent à la frontière du SA** de **décider** entre eux de quel est le **meilleur routeur** pour joindre telle ou telle destination
  2. Suite à ça, il faut qu'ils **informent l'ensemble des routeurs** du **SA** de cette route.
     - ça va se faire en redistribuant les informations apprises de l'Internet via **BGP** à l'intérieur du de protocole **OSPF**

Interactions BGP-OSPF-2-

- **Rôle de l'IGP dans AS10**
  - distribuer la topologie interne du réseau
- **Rôle de l'EGP dans AS10**
  - Annoncer les routes internes à l'extérieur (via eBGP)
  - Apprendre les routes externes (via eBGP)
  - Distribuer (avec iBGP) les routes externes dans le réseau et éventuellement ré annoncer ces routes aux AS voisins

Full mesh iBGP entre les routeurs frontières de l'AS
permet aux routeurs frontières de déterminer quel routeur est le meilleur pour une route donnée



Le routeur qui a la meilleure route pour un réseau donné annonce cette route via IGP aux routeurs internes

Interactions OSPF-BGP –2-Commentaires-

- **Exemple de AS 10 qui possède deux connexions avec AS 40 via les routeurs BR1 et BR2**
  - Ces deux connexions n'ont pas les **mêmes performances**.
  - Il faudra, donc que les routeurs se mettent d'accord pour savoir si la meilleure route vers **AS40** passe par **BR1** ou **BR2**.
- **Solution :**
  - on va permettre aux différents routeurs qui se trouvent à la frontière du AS de s'échanger les tables de routages `inter domaines` via `une liaison BGP.` Ceci se fera via iBGP
    - tout routeur se trouve à la frontière de l'AS connaîtra l'ensemble des tables de routages des autres routeurs frontière. Ils vont pouvoir, chacun, sélectionner **la meilleure route pour joindre chaque destination**.
    - Le **routeur** qui est **responsable de cette route** va **injecter** l'information associée à cette route à l'intérieur du SA de telle façon que chaque routeur se trouve à l'intérieur du **SA** sache que pour joindre les réseaux de **l'AS 40** il faut passer par **BR1** plus tôt que par **BR2** (par exemple)

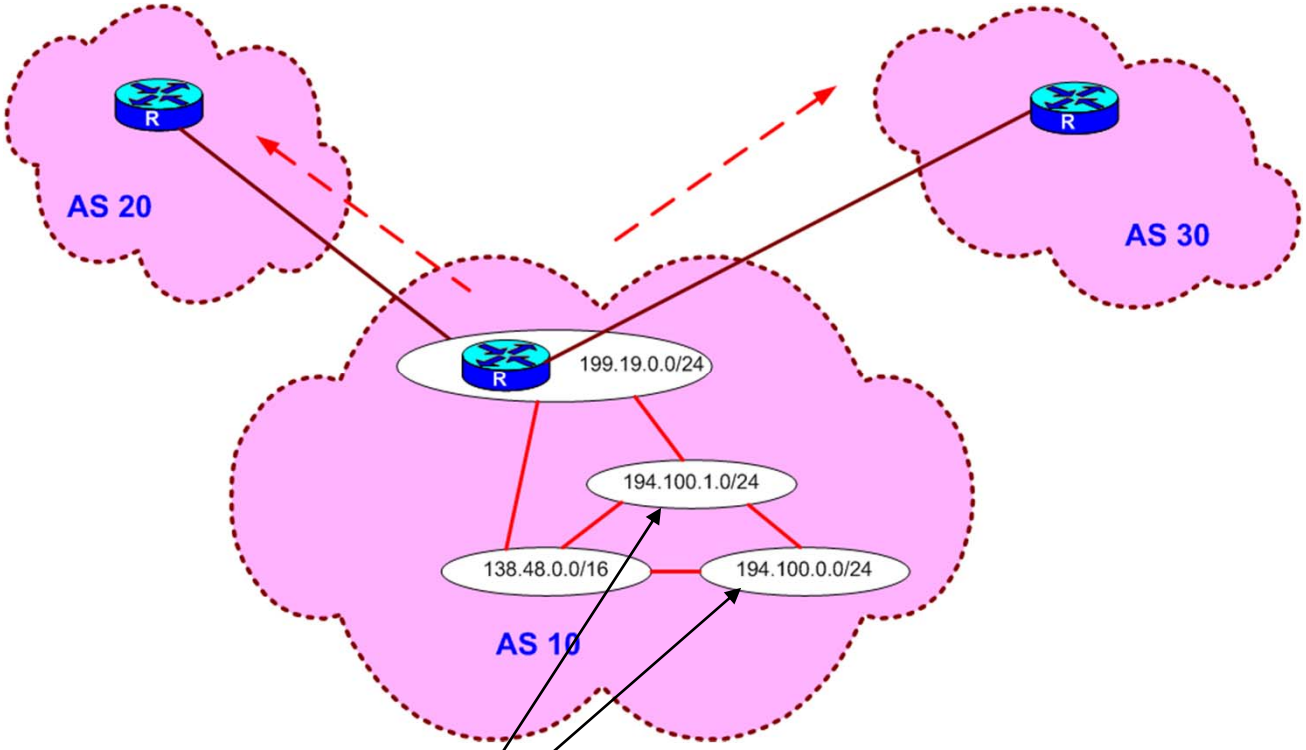
BGP – Quelques Exemples

BGP : exemple 1 -

- Au niveau d l'AS10, on a un certain nombre de sous réseaux qui existent :

  199.19.0.0/24 , 138.48.0.0/16, 194.100.0.0/24, 194.100.1.0/24

- Pour que ces réseaux ***soient accessibles de l'extérieur*** il faut que l'AS10 annonce les vecteurs de distance aux routeurs BGP de l'AS20 et de l'AS30.



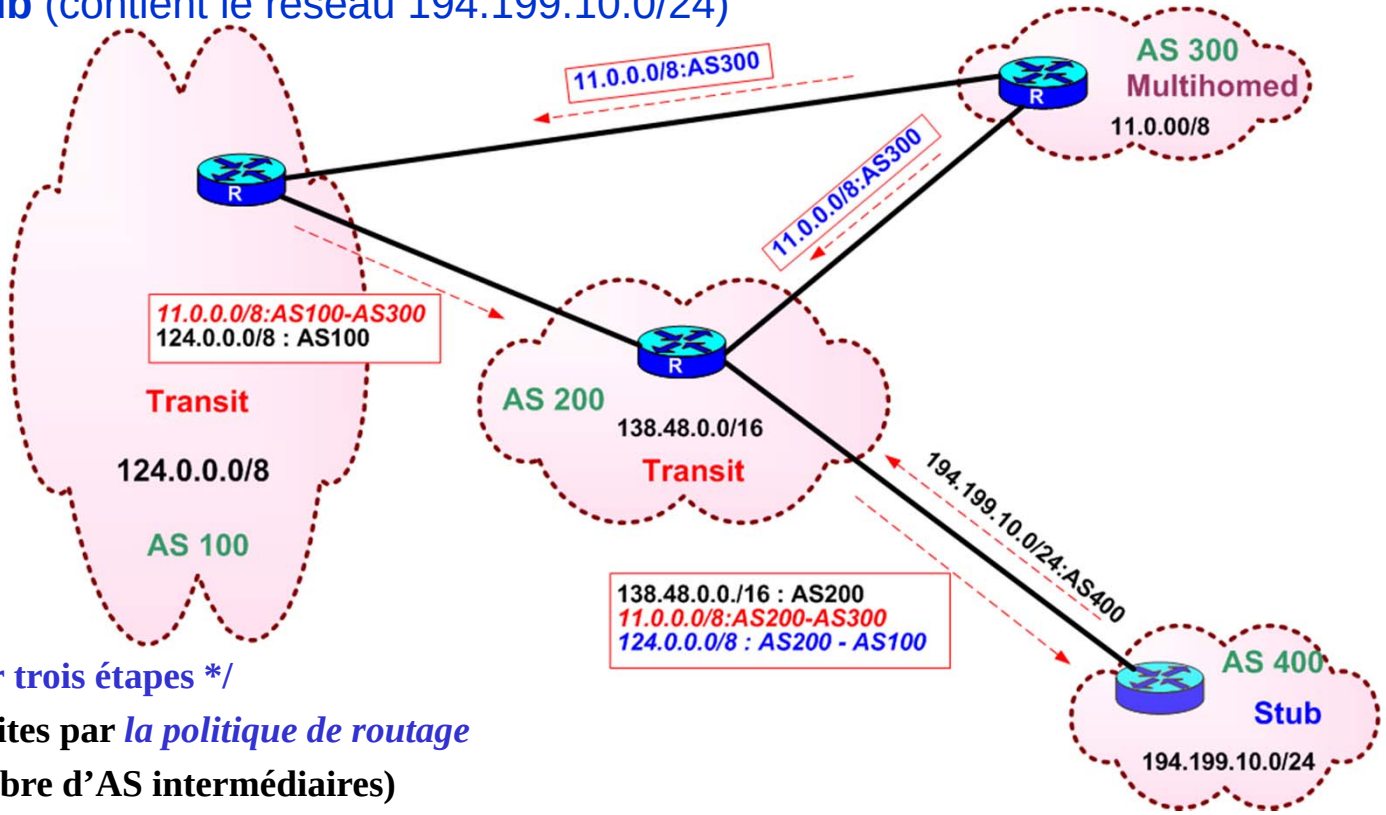
Un routeur BGP annonce aux routeurs extérieurs les sous réseaux qui peuvent être atteints via lui (199.19.0.0/24, 138.48.0.0/16, 194.100.0.0/23)

BGP : exemple 1 -suite-

- **Donc l'AS10 va construire un vecteur de distance qui indique pour chaque réseau le chemin pour joindre ce réseau**
  - On va essayer d'annoncer les routes de la façon la plus agrégé possible de façon à réduire la quantité d'information => même s'il y a quatre réseaux on va annoncer trois routes différentes.
    - **Une route vers 138.48.0.0/16**
    - **Une route vers 199.19.0.0/24**
    - **Une route vers les deux sous réseaux :**
      - **194.100.0.0/24**
      - **194.100.1.0/24**

        Fusionnés 194.100.0.0/23

- **Un exemple un peu plus compliqué :**
  - **AS300** est un SA multi-homed, donc ne fait pas de transit à l'intérieur de l'Internet (il contient le réseau 11/8)
  - L'**AS100** (contient le réseau **124/8**) et **AS200** (contient le réseau **138.48/16**) étant des SA de transit, il vont répercuter cette route vers tous leurs voisins.
  - **l'AS400** un **SA Stub** (contient le réseau 194.199.10.0/24)



Choix de la meilleure route /*par trois étapes */

1. **éliminer les routes interdites par *la politique de routage***
2. **route la plus courte (nombre d'AS intermédiaires)**
3. **si plusieurs routes de même longueur, choix d'une d'entre elles suivant un critère quelconque**

BGP : exemple-2-suite-

- **AS300 est un réseau multi-homed**
  - ✓ AS300 on va annoncer uniquement des routes qui permettent de joindre des endroits à **l'intérieur de l'AS300**.
  - ✓ Donc, l'AS300 va annoncer seulement comment joindre le réseau 11/8

- **L'AS100 et AS200 étant des réseaux de transit, il vont répercuter cette route vers tous leurs voisins.**

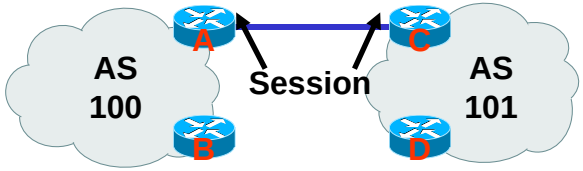
➢ **l'AS200 va annoncer une route directe AS200-AS300 pour atteindre le réseau 11/8.**

- Donc la route vers le réseau 11/8 va être répercutée par l'AS200 ensuite il va indiquer que le chemin vers le réseau 11/8 choisi passe par AS200. Il aurait pu annoncer une route AS200-AS100

LES MESSAGES BGP

Les messages BGP- comment construire une session BGP

- **Deux routeurs BGP deviennent des voisins « ou paire »**
  - Sur des connexions point à point fiables (TCP / port 179)



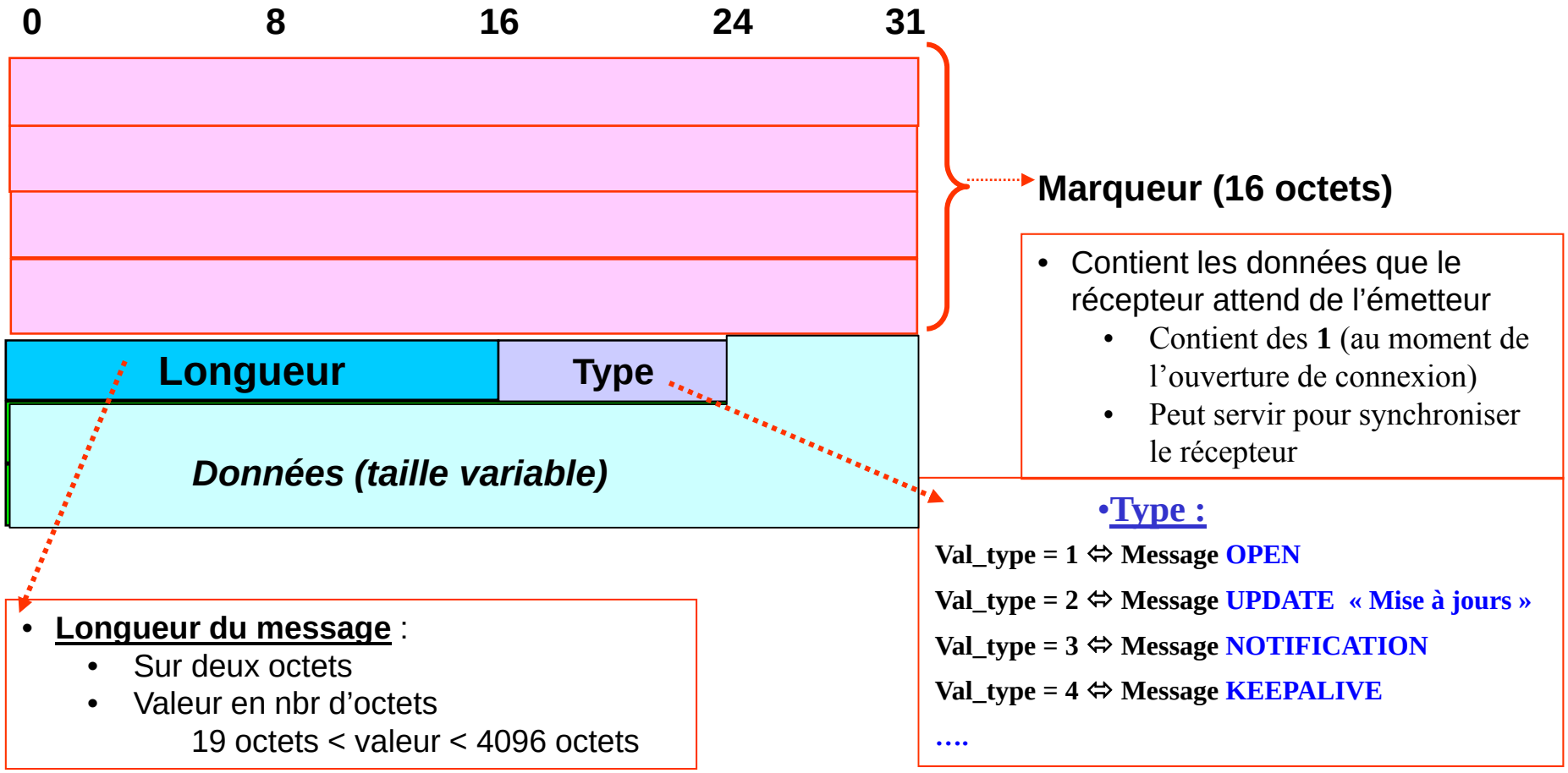
- **Négociation des paramètres avec le message `open`**
  - Voisin de BGP, **temps de maintien de l'information**, **attributs** et authentification (BGP peut utiliser MD5 dans son en-tête)
  - Un message de **`notification`** est utilisé pour informer le voisin des erreurs

- **Après établissement de la connexion, les voisins échangent entièrement les tables de routage en utilisant des message `update`**
  - Chaque message liste les attributs et le chemin des **NRLI (préfixe/longueur)**

- **BGP utilise des mise à jours périodiques pour signaler les changements dans le réseau**
  - Message **`update`** pour informer les voisions des **suppressions** et des **injections de routes**
  - Un débit faible est suffisant pour maintenir l'environnement de routage

Les messages utilisés par BGP

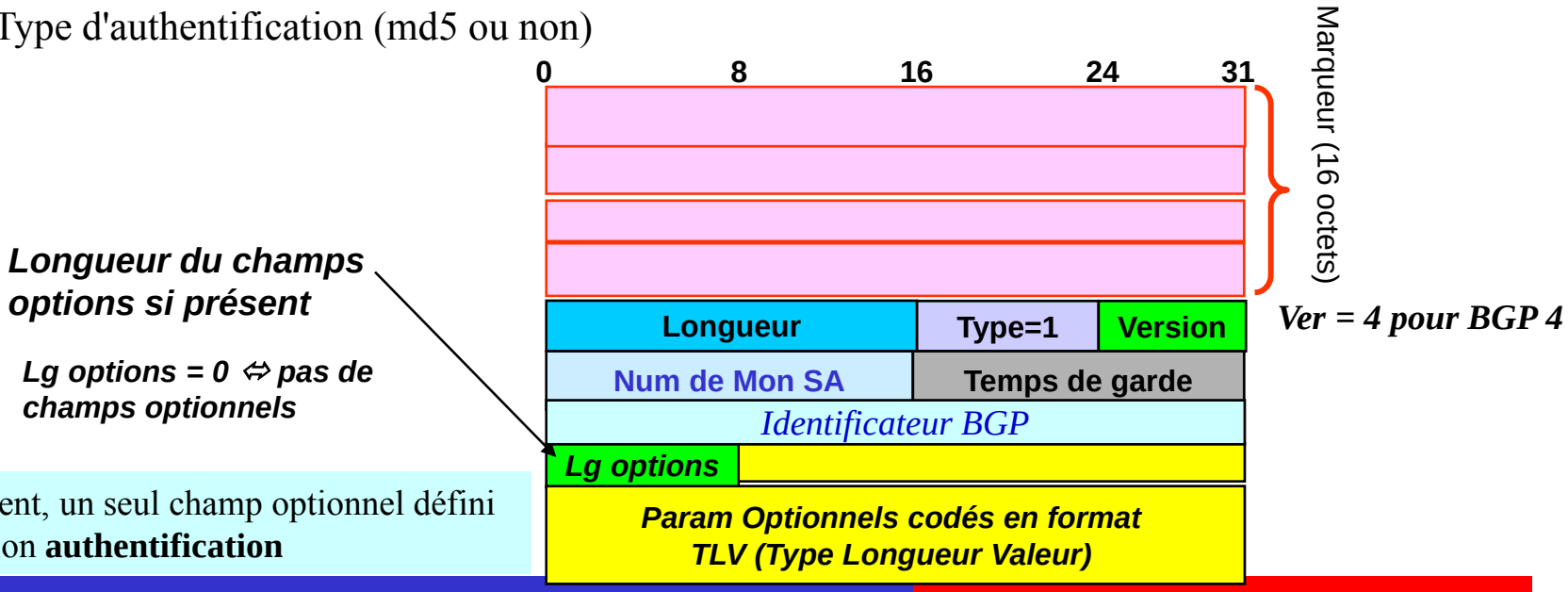
- **Messages BGP sont des messages de longueur variable (** Avec une en-tête de taille fixe qui est commun à tous les messages BGP)
- ***Rôle de l'entête :*** Comme TCP est un protocole qui transmet un flux de caractères, la notion de message disparaît après transmission. L'en-tête permet **au récepteur de ré découper le flux en message**



Le message OPEN

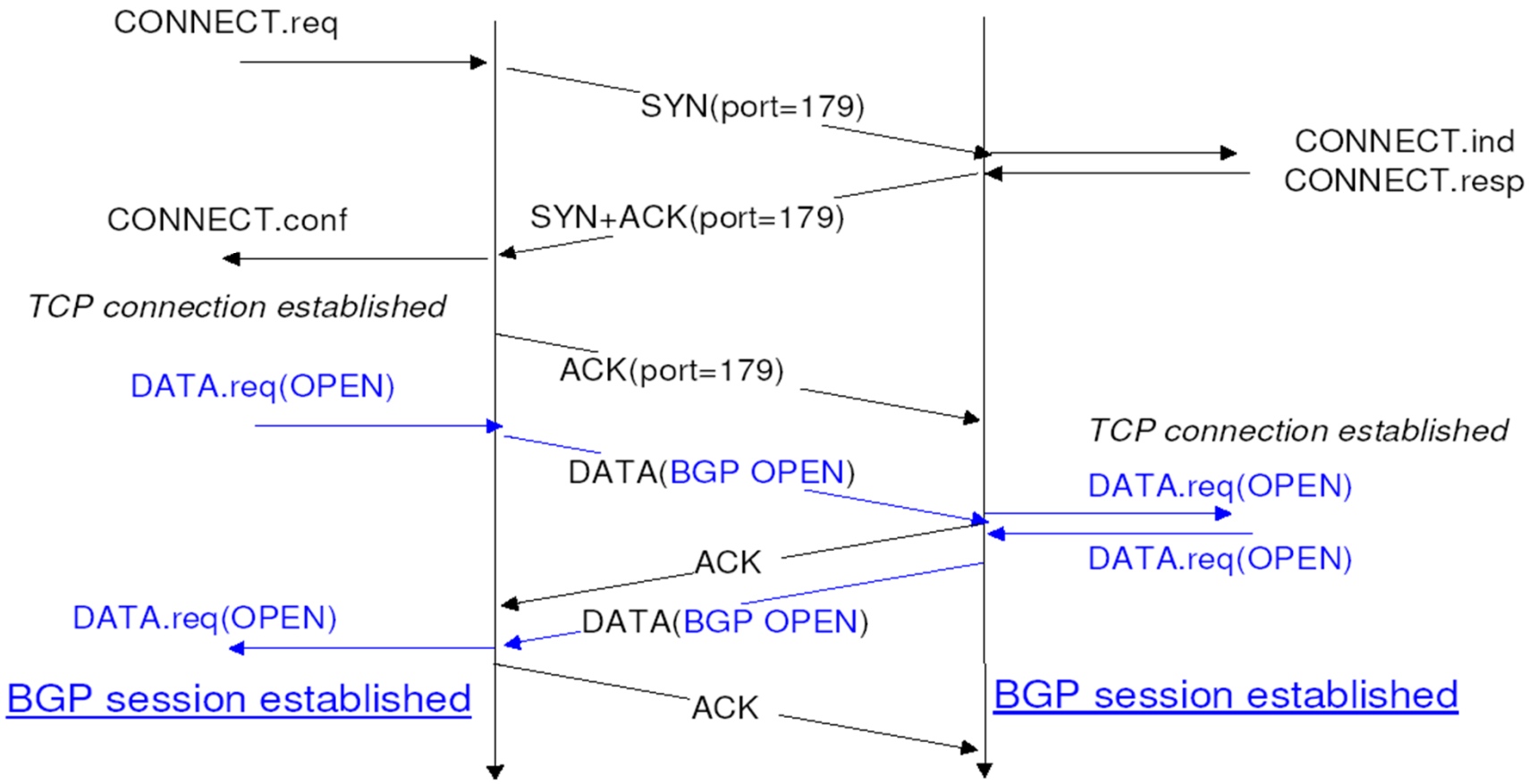
- **Message OPEN :** Pour établir la session de BGP entre deux routeurs paires
  - avec **ACK** de l'ouverture de sessions

- **Format du message :**
  - Définit et échange de nombreux paramètres
  - **Numéro d'AS** auquel appartient l'expéditeur (codé sur 16 bits)
  - Temps de Garde « Hold-timer » : temps maximum (sec) entre deux messages KEEPALIVE ou UPDATE
    - Hold-timer = soit 0s soit 3s au minimum (si sa valeur est différente de 0)
  - Le champ Identificateur BGP contient une des adresses IP du routeur émetteur du message (Ce champs doit être le même sur toutes les interfaces ⇔ le ID du routeur)
  - Type d'authentification (md5 ou non)



Actuellement, un seul champ optionnel défini : information **authentification**

Etablissement de la connexion



Message OPEN envoyé juste après l'ouverture de la connexion TCP par les deux voisins

- **Message NOTIFICATION**

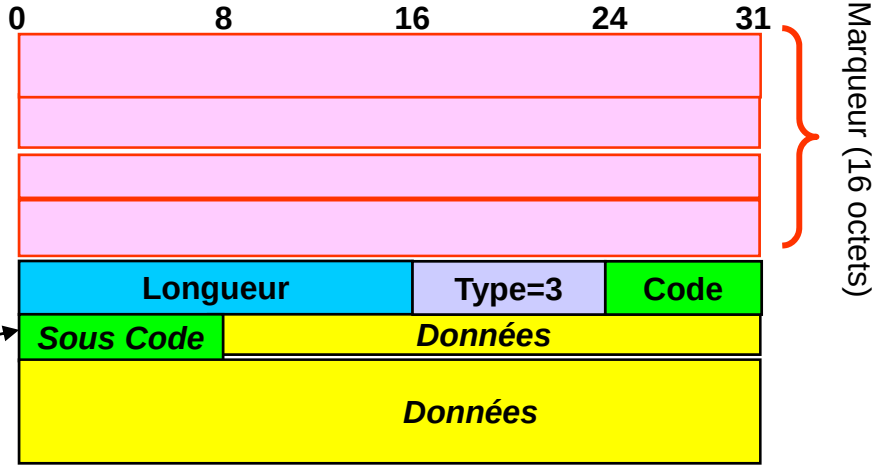
Err Code
SubCode

Exemple d'erreurs :

2. **Erreur Message OPEN**
   - Version non supportée
   - paramètre optionnel non supporté , ...
3. **Erreur Message UPDATE** ..
4. **Expiration du Timer : Hold Timer Finite**
5. **Erreur d'exécution de la machin d'état**

Données additionnelles (longueur variable)

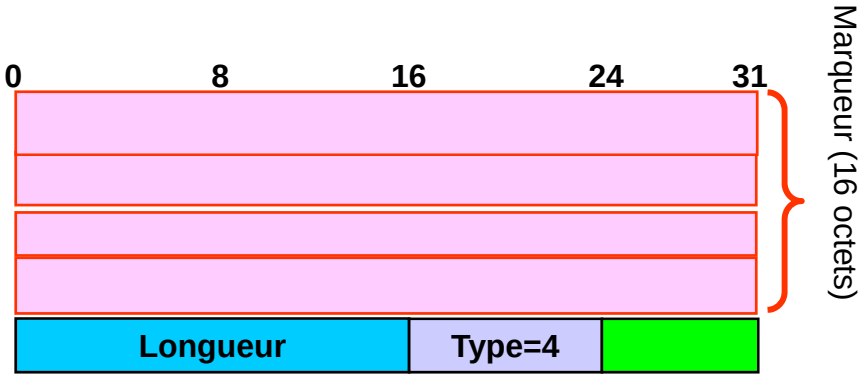
- utilisé en **cas d'erreurs** pour informer le voisin du problème avant de clôturer la session de BGP (normalement on doit avoir rarement voir passer ces messages)



Fournit des informations spécifiques détaillées sur la nature d'erreur

Les messages KEEPALIVE

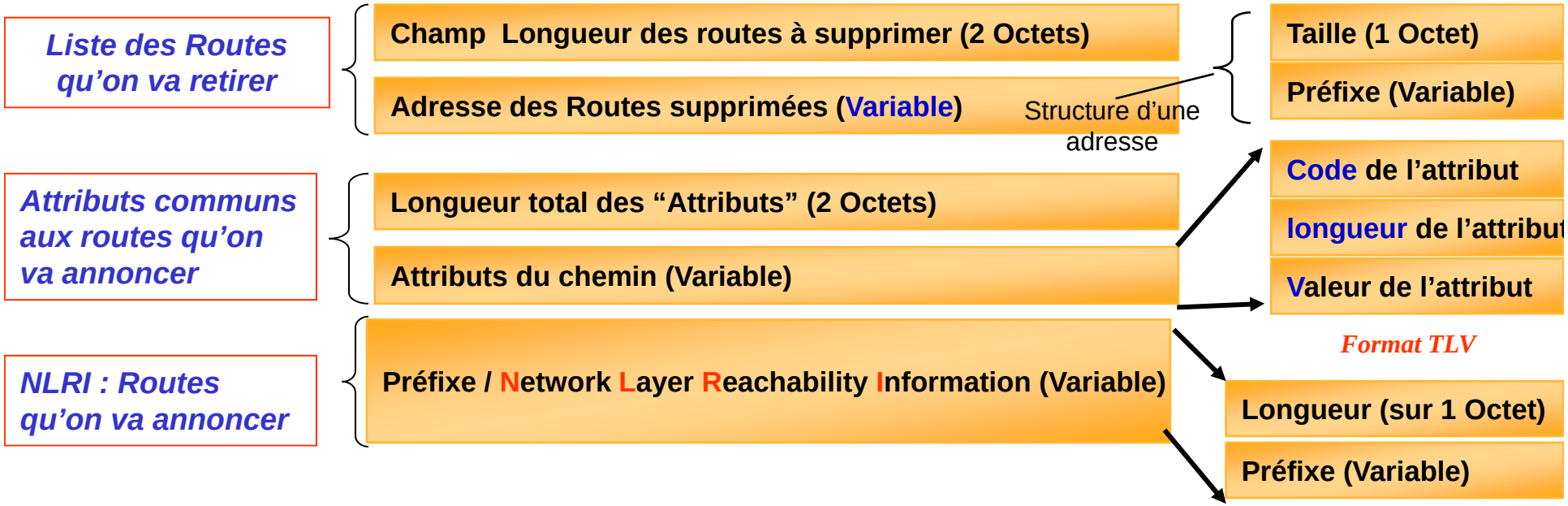
- **Messages KEEPALIVE**
  - Contient seulement l'en-tête par défaut
  - transmis régulièrement (toutes les **HoldTime/3**) s' il n'y a aucun autre message (pour savoir si le voisin est toujours actifs)



Message UPDATE

- **Message UPDATE « Mise à jours » :** Message utilisé pour **distribuer les informations de routage** « **pour contrôler le trafic** » : Il sert à deux choses :
  1. **Annoncer des préfixes d'IP (de nouvelles routes) qui n'étaient pas connues**
  2. **retrait des préfixes d'IP des routes (n'existe plus)**
  - **Découpé en deux parties.**

Exemple *: <18, 192.213.128.0> indique une route à retirer qui est (représentation CIDR) : 192.213.128.0/18*



- **Une mise à jour BGP permet d'annoncer une route (et une seule) à un voisin, ou bien de supprimer plusieurs routes qui ne sont plus accessibles**
- **Chaque message contient des attributs comme : Origine, chemin d'AS, Next-Hop, …**

- Les attributs trouvés dans des messages UPDATE :
- ORIGINE : **obligatoire dans le message update**
  - comment la route a été apprise (IGP, EGP, …etc.)
- AS-PATH : **obligatoire dans le message update**
  - la liste d'AS via lesquels, l'annonce est passé
- NEXT- HOP : **obligatoire dans le message update**
  - L'adresse IP du routeur qui a annoncé la route
- MULTI-EXIT-DISCRIMINATOR : **optionnel dans le message update**
  - peut être utilisé pour le contrôle de trafic
- LOCAL-PREF : **optionnel dans le message update**
  - peut être utilisé pour le contrôle de trafic
- ATOMIC AGGREGATE et AGGREGATOIR : **optionnel dans le message update**
  - qui peuvent être utilisés pour faire l'agrégation (pas utilisés dans le contrôle de trafic)
- COMMUNITY : **optionnel dans le message update**
  - Groupement logique des routes avec les même politiques de routage. Il permet de marquer les routes ayant des points communs avec une valeur, ce qui permet de simplifier le traitement des autres systèmes autonomes

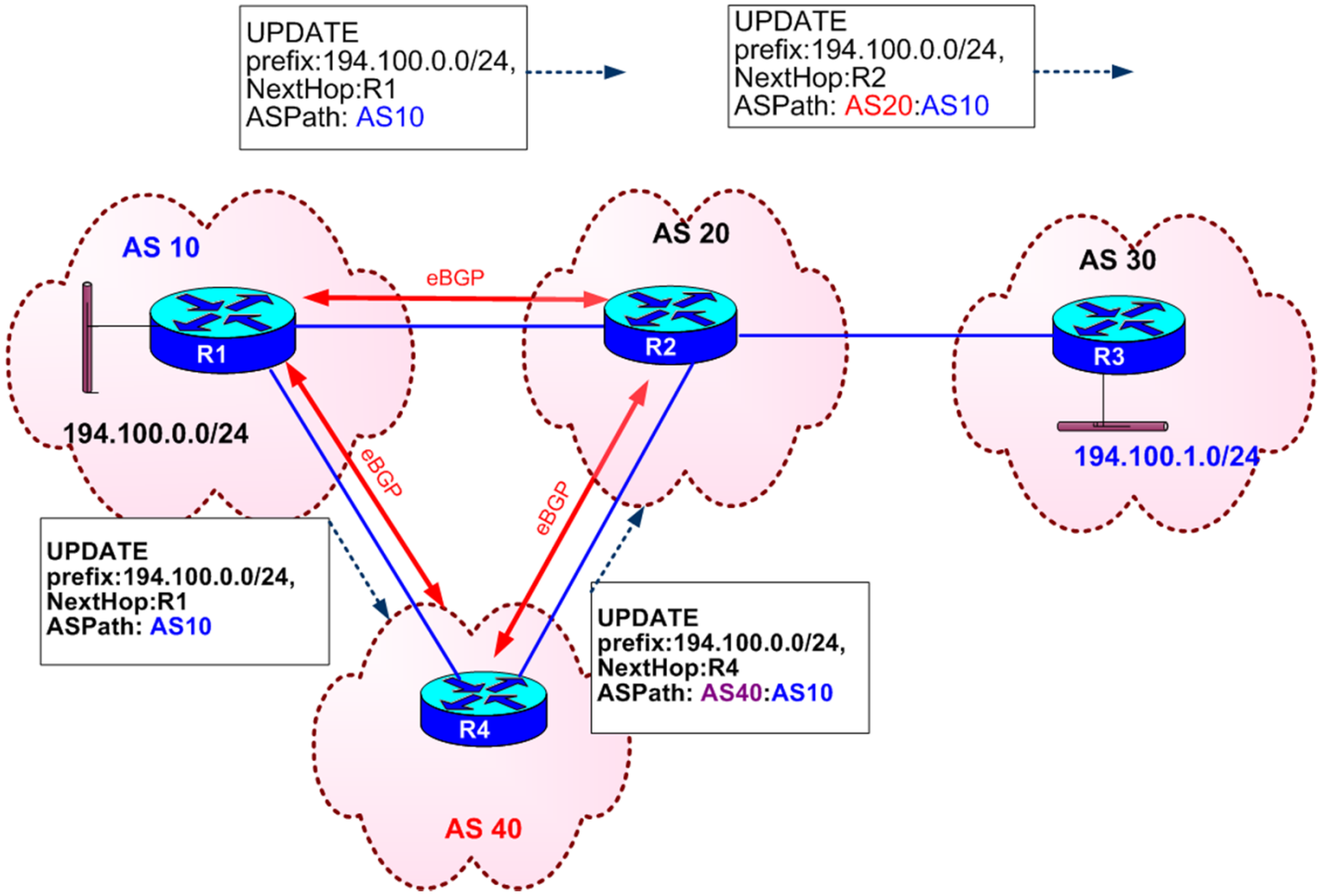
Attributs et contrôle de trafic BGP

- **<u>Si on veut contrôler le trafic, on va devoir intervenir sur les attributs des messages</u> UPDATEs.**
  - Ces attributs représentent les caractéristiques qui vont préciser les routes qu'on va annoncer dans BGP
- ***<u>Sur quelles caractéristiques ?</u>***

1. **ORIGINE**
   - La façon dont la route a été connue. Est-ce qu'elle provient d'un protocole de routage inter-domaine ou intra-domaine
2. **AS-PATH**
   - Lorsqu'on annonce une route, on va connaître la liste des SA intermédiaire par lesquelles on est passé depuis la source jusqu'au routeur qui annonce la route
3. **NEXT- HOP**
   - L'adresse IP du routeur qui nous permet de rejoindre la destination
4. **MULTI-EXIT-DISCRIMINATOR**
5. **LOCAL-PREF**

Quelques exemples de manipulation du message UPDATE

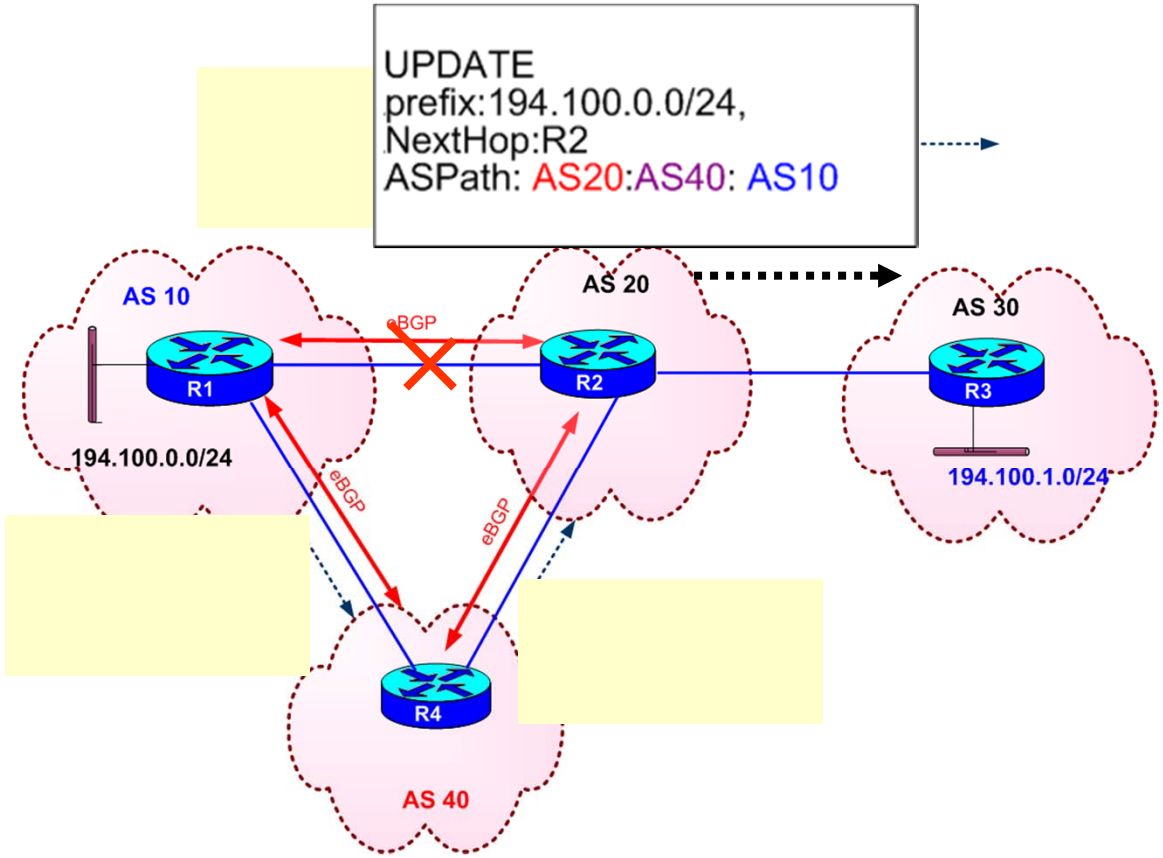
R1 initie l'échange



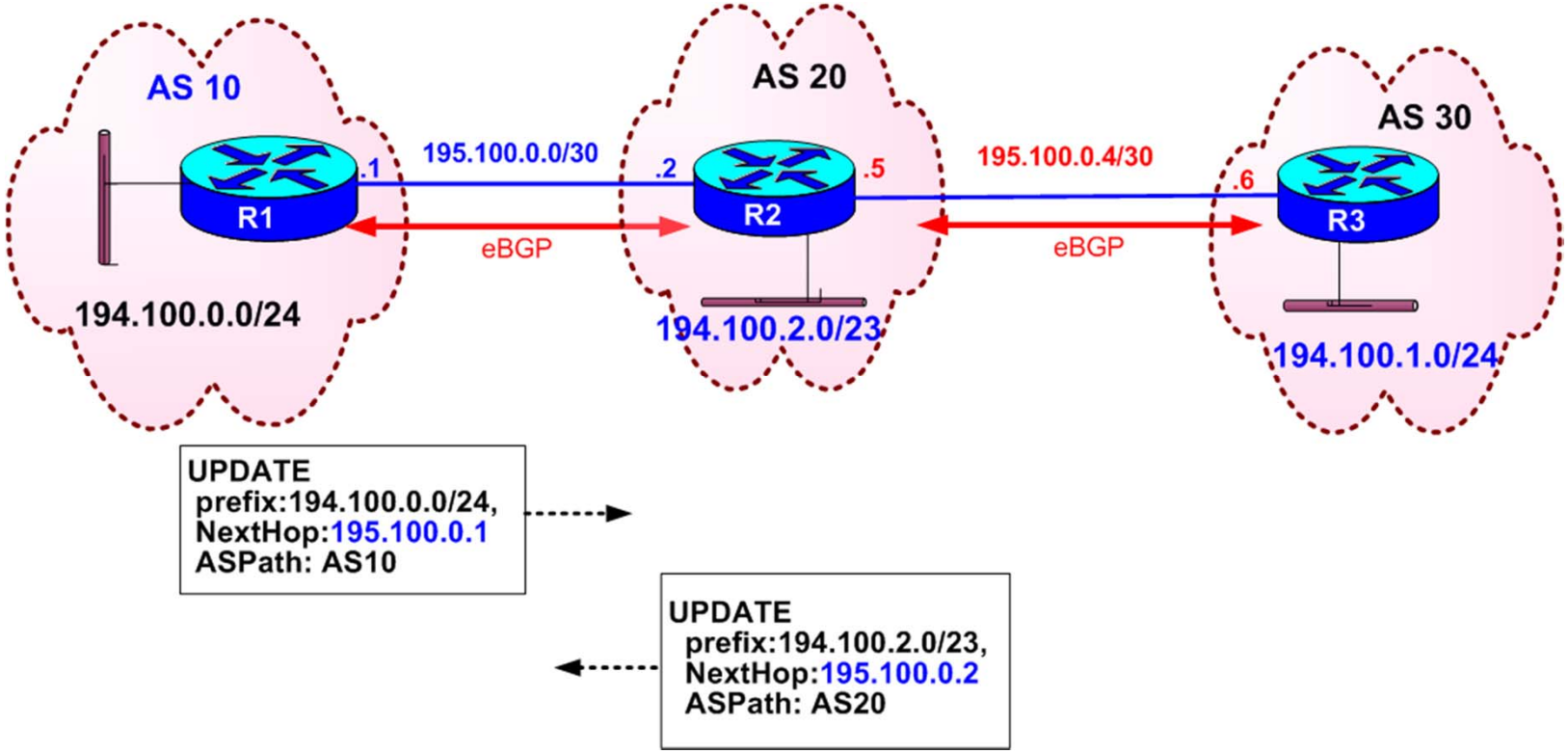
Qu'est ce qui se passe si la liaison AS10-AS20 tombe en panne ?

BGP et IP – exemple 1.2

- Si la liaison **AS10-AS20** tombe en panne , **AS20** ne peut plus considérer le chemin appris par **AS10.**
- Donc, AS20 va retirer ce chemin de sa table de routage et il va sélectionner à la place le chemin appris par **AS40.**
  - Ceci va forcer AS20 à envoyer le message **UPDATE** suivant à **AS30 :**

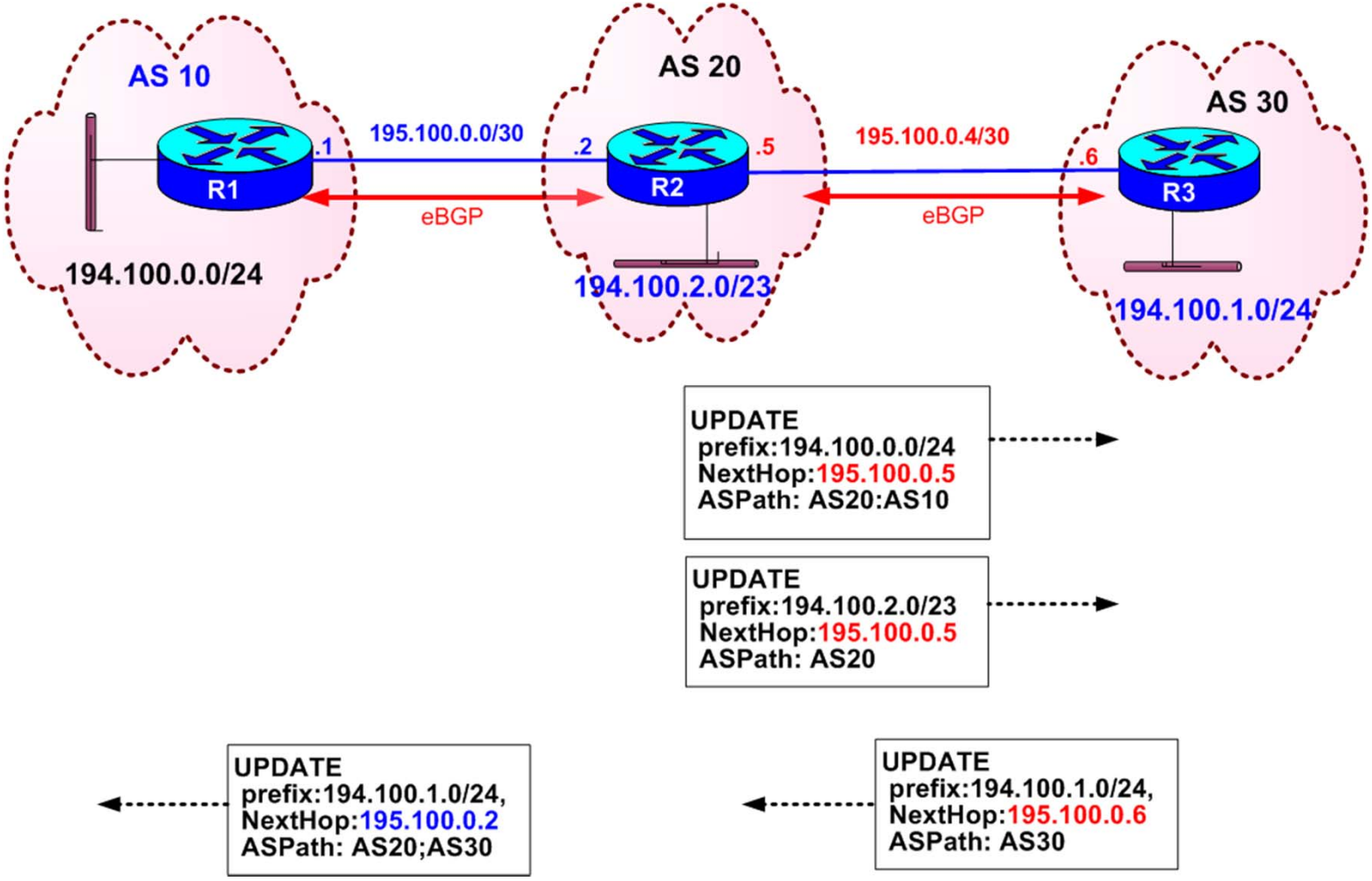


- **Les principaux attributs de la route dans le message UPDATE**
  - **NextHop** :Adresse IP du routeur utilisé pour atteindre la destination
  - **ASPath** : Le chemin suivi par l'annonce de la route

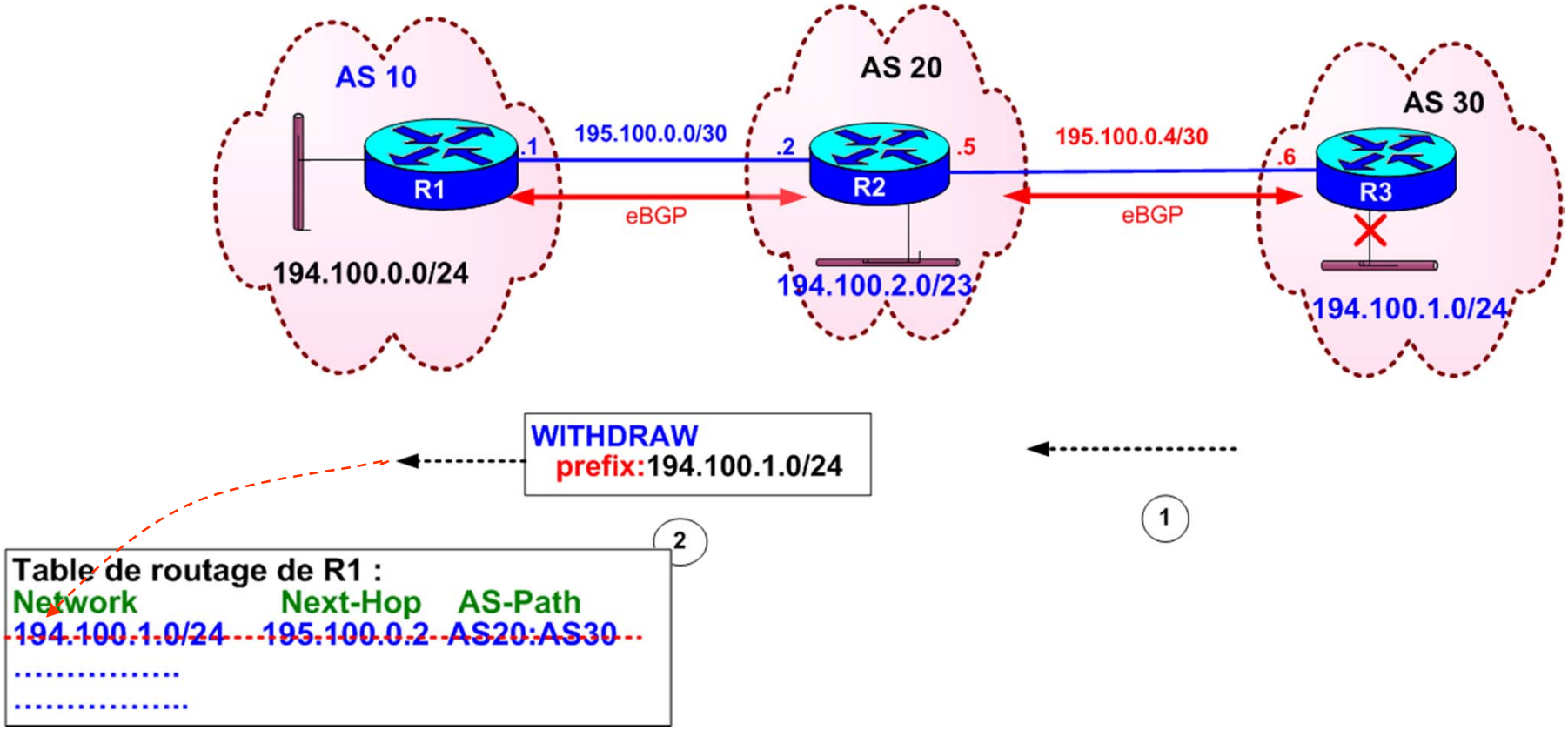


N.B : Dans cet exemple, on considère les messages BGP concernant les réseau suivants : :194.100.0.0/24, 194.100.1.0/24 et 194.100.2.0/23.

BGP et IP – exemple 2.2



BGP et IP – exemple 3 – suppression des routes

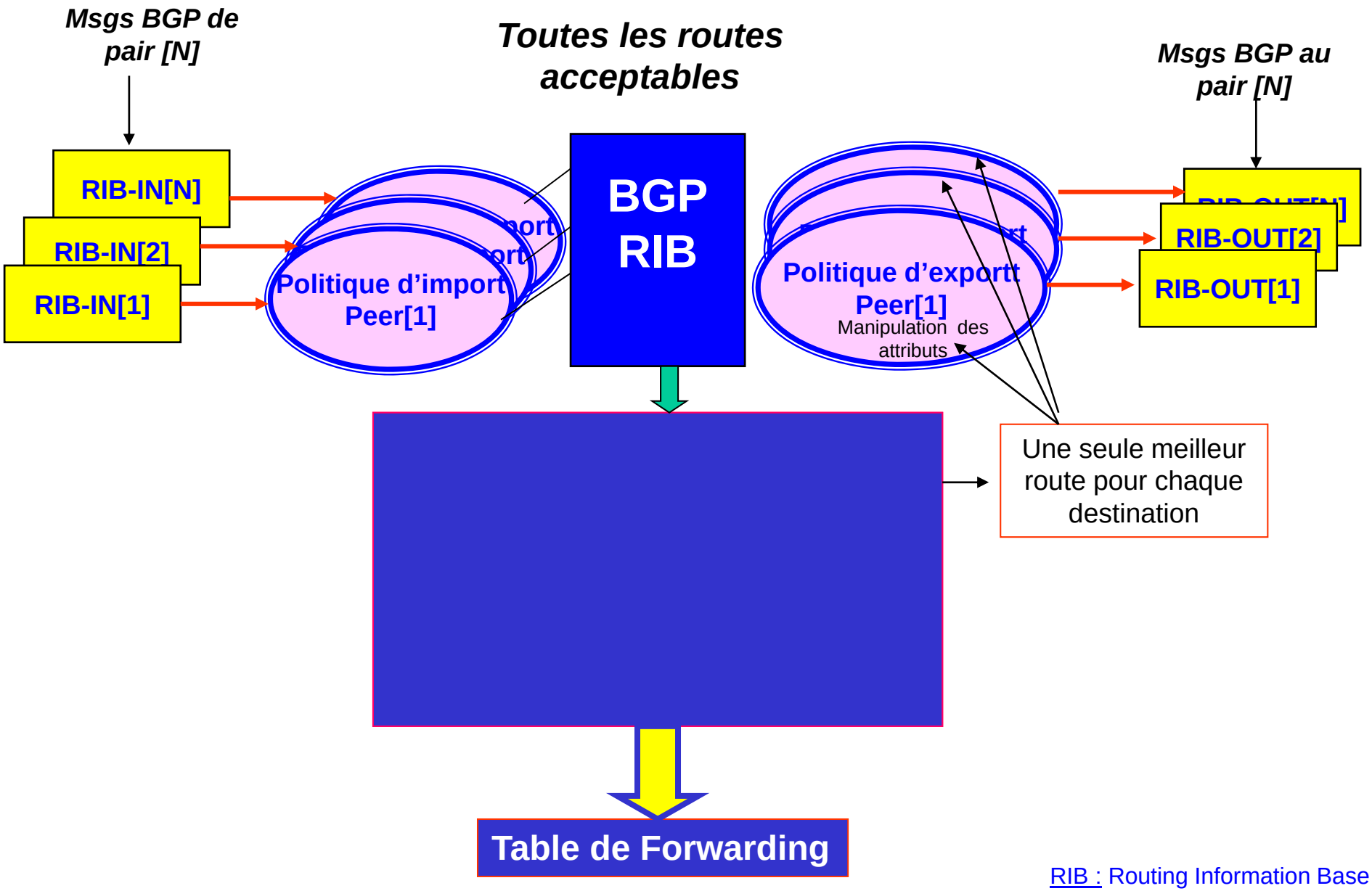


Processus de routage dans les routeurs BGP

Organisation du routeur BGP-

- *La clé de contrôle de trafic au niveau de BGP, c'est **l'organisation** du routeur et le **processus** qui caractérise le comportement du routeur lorsqu'il reçoit les messages **UPDATE.***

Organisation du routeur BGP-



RIB : Routing Information Base

Organisation du routeur BGP-

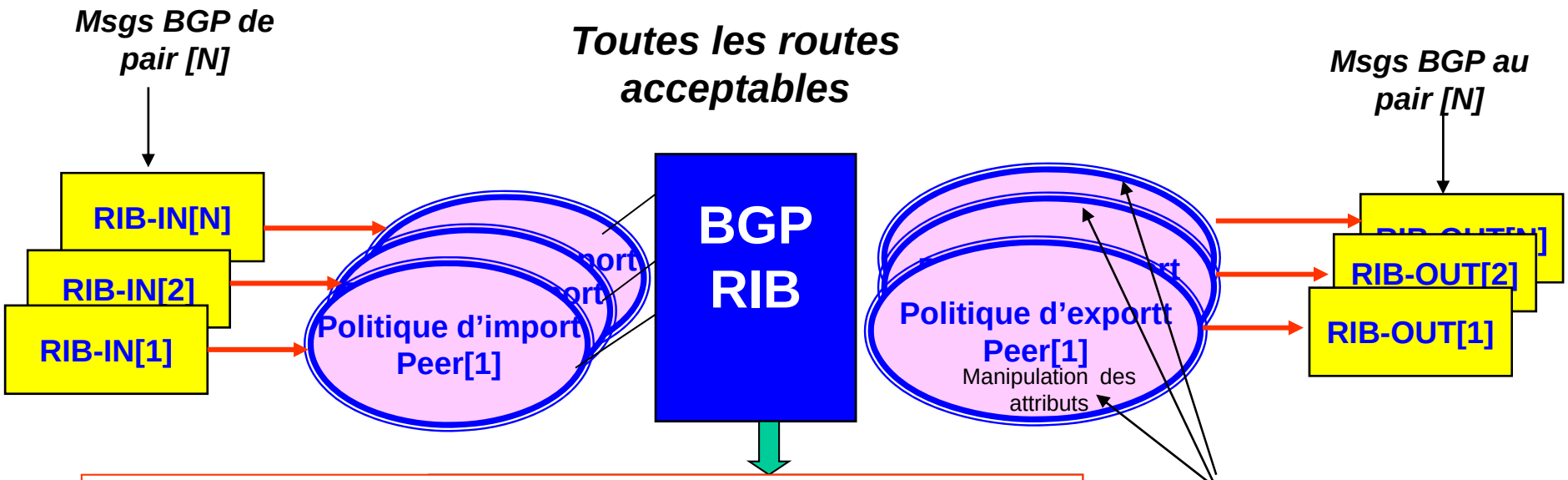
- ***Un routeur BGP va maintenir, en général, plusieurs sessions BGP avec ses voisins.***
  - *Pour chaque voisin avec qui il maintient une session BGP, le routeur va conserver une table qui sera la liste des messages **UPDATE** reçus de son voisin (qui est la **RIP-IN**)*
  - *On aura une table équivalent qui contiendra la liste des messages **UPDATE** qui ont été envoyées au voisins (qui sera la **RIP-OUT[i]** pour le voisin n° i)*
- ***NB :***
  - Lorsqu'on reçoit un message **UPDATE** d'un voisin, le routeur ne va pas l'accepter comme tel.
  - Avant de l'installer dans la table de routage **RIB** (Routing Information Base), il va être **soumis à un filtrage.**
  - Au niveau de **BGP**, on peut filtrer les routes qui sont annoncées entre voisins en se basant sur n'importe quel critère (Sur n'importe quelle caractéristique de la route, on va décider si la route est acceptable ou non).

- ***Grâce au filtre Import, on va sélectionner les routes qui sont acceptables et acceptées au niveau de la RIB***

Organisation du routeur BGP-

- ***Dans la base RIB, on va rassembler toutes les routes qui sont reçues des sessions BGP externes et qui sont considérées comme acceptables.***
  - *En générale, on aura **plusieurs routes** vers **chaque destination**.*
    - *Dans ce cas, on devra faire un choix pour déterminer **qu'elle est la meilleure** route qui nous permettra de joindre **chaque destination**.*
  - *Pour ça, on va utiliser **le processus de décision de BGP** : est un algo qui va pour chaque destination connue dans la table RIB déterminer parmi toutes les routes qui sont connues pour cette destination, **quelle est la meilleure route pour ce routeur BGP**.*
  - ***Le choix de la meilleure route va se faire en utilisant un certain nombre de critères qui vont être utilisés en cascade à la suite les uns derrière les autres***

Processus de décision BGP



BGP Decision Process :

1. Préférer les routes avec le **poids le plus élevée**
2. Préférer les routes avec le plus grand **local-pref**
3. Préférer les routes avec l'**AS-Path** le plus court
4. Préférer les routes avec le **MED** le plus petit
5. Préférer les routes apprises par **eBGP** aux routes **iBGP**
6. Préférer les routes avec les **next-hop les plus proche**
7. Tie **breaking rules**

   Préférer les routes apprises du **router-id** le plus faible

Une seule meilleur route pour chaque destination

Table de Forwarding

RIB : Routing Information Base

Processus de décision BGP

- ***Le choix de la meilleure route va se faire en utilisant un certain nombre de critères qui vont être utilisés en cascade à la suite les uns derrière les autres***

1. ***Sélectionner la route avec le poids le plus élevé : si suite à cette sélection, on n'a plus qu'une seule destination vers la destination, on a fini***
2. ***Si on reste avec plusieurs routes vers une destination, on va considérer l'attribut local_pref (c'est un attribut que l'on peut insérer à l'intérieur d'un message UPDATE).***
   - ***Si suite à ça, on n'a qu'une, on a fini.***
   - ***S'il nous reste plusieurs, on passe au suivant***

3. ***Regarder l'attribut AS_PATH : Sélectionner les routes qui ont l'AS_PATH la plus court de façon à favoriser les autres qui passent par un petit nombre d'un SA (⇔ en principe, une route qui passe par un petit nombre d'AS doit être de meilleure qualité d'une route qui passe par un plus grand nombre). Si suite à ça, on n'a qu'une, on a fini. S'il nous reste plusieurs, on passe au suivant***

4. ***Le MED le plus petit : «Multi-Exit-Discriminator » : On sélectionnera la route qui a le MED le plus petit. S'il nous reste plusieurs, on va continuer à regarder d'abord en préférant une route qui nous a été annoncée par une session eBGP par routeur externe***

5. ***En fin, si ça ne suffit pas, on regarder la route qui passe par le voisin le plus proche à l'intérieur du réseau IP***

- ***Et si tous ceci ne suffit pas, on pourra regarder la route la plus vieille et enfin, on pourra regarder la route qui a été annoncée par le routeur qui a l'adresse IP la plus petite.***

Processus de décision BGP

- ***Avec ce processus, le routeur BGP va sélectionner une route « pour une destination » et qui sera sa meilleure route pour cette destination.***
  - ***La meilleure route pour chaque destination, est copiée directement dans la table de forwarding du routeur***
  - ***Dans certains cas, on peut placer dans la table de forwarding plusieurs routes vers la même destination***

Processus de décision BGP

- Une fois qu'on a construit la RIB avec la **meilleure route pour chaque destination**, il va falloir qu'on détermine quelle information qu'on annonce à chacun de nos voisins BGP.

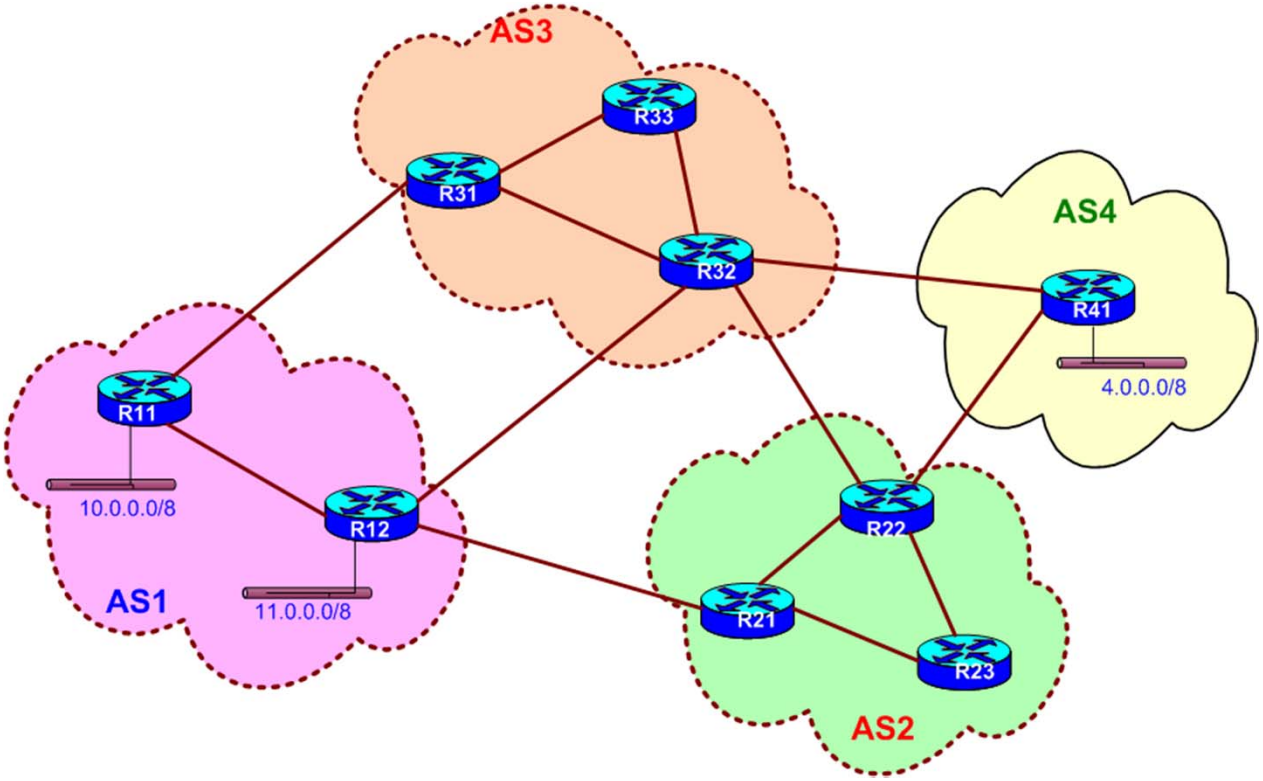
- Pour chaque voisin BGP, on va avoir un **filtre pour l'exportation des messages BGP**, ce filtre va nous indiquer qu'elles sont les routes qu'on **peut annoncer à son voisin** :
  - ***Selon le type de liaison qu'on a avec son voisin****, on va lui annoncer toutes nos routes ou une partie des routes, ou bien seulement les routes de nos clientes, …etc.*
  - *Pour chaque voisin, on va spécifier de façon flexible le type de routes qu'on peut annoncer*

- **Toute la puissance de BGP va résider dans les filtres d'importation et d'exportation des routes et dans le processus de décision**

- **Pour simplifier le processus de décision, on permettrait au gestionnaire de réseau d'activer ou de désactiver chacun des critères de sélection**

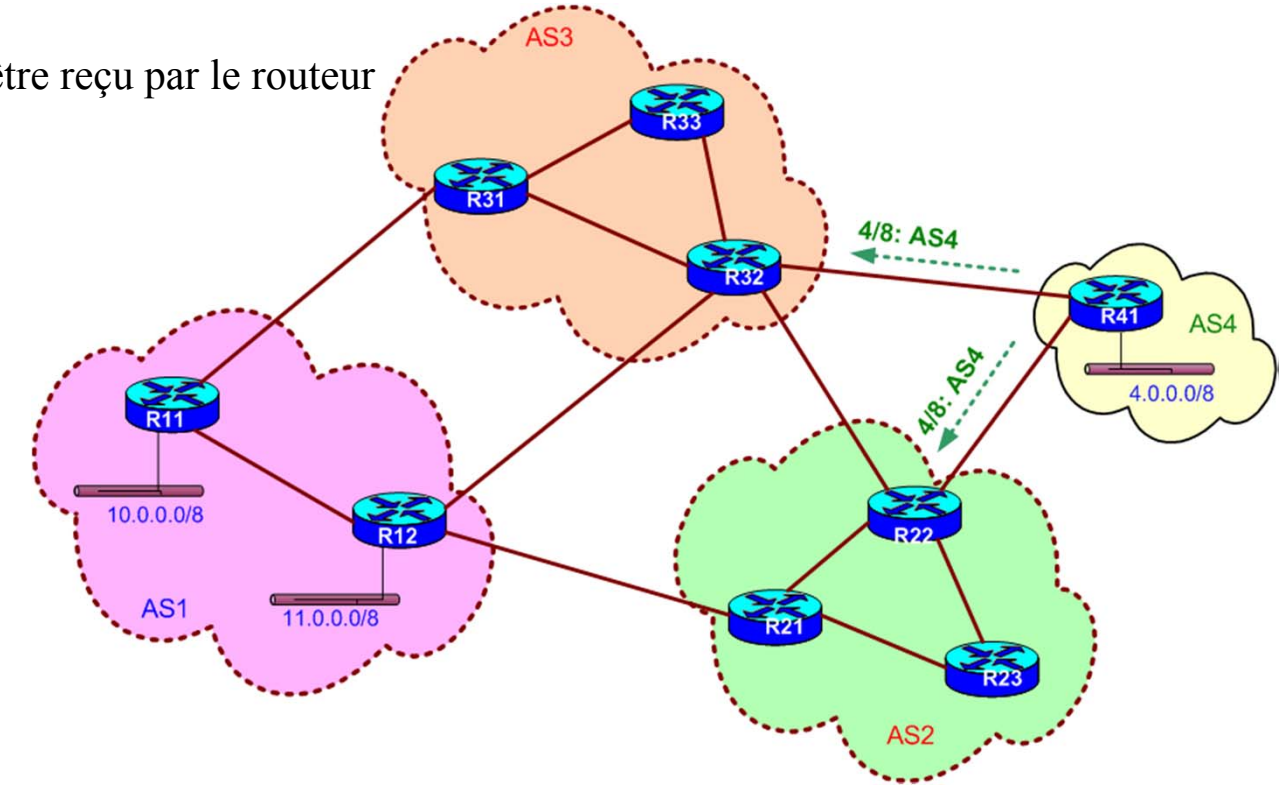
BGP – Contrôle de trafic

- **Exemple : un petit réseau composé de 4 AS (AS1, AS2, AS3 et AS4) :**
  - **Hypothèse** : tous les **SA** ont pris comme **politique d'annoncer toutes** les routes à tout le monde
  - **But** : regarder comment la route vers le réseau **4/8** dans **AS4** va petit à petit se propager via BGP à l'intérieur du réseau (⇔ **Distribution de la route vers 4.0.0.0/8)**



BGP : Exemple de base – étape 1-

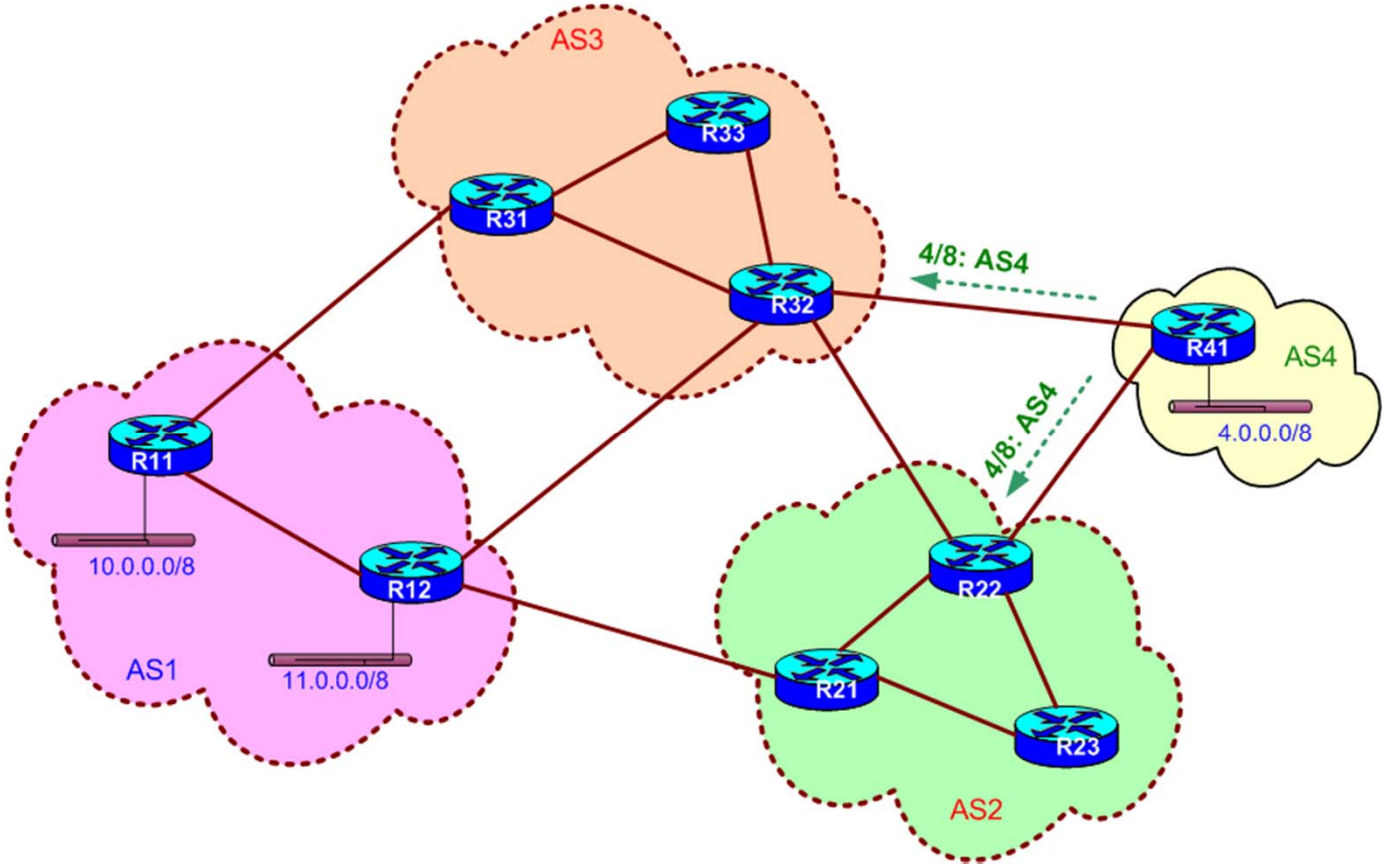
- D'abord, le routeur **R41** qui va annoncer cette route vers ses deux voisins qui sont les routeurs **R32 et R22.**
  - **R41** annonce sa route en indiquant le **préfixe de la destination 4/8** et en mettant comme attribut dans **l'AS_PATH** le fait que cette route démarre à l'intérieur de **l'AS4**
  - Ce message **UPDATE** va être reçu par le routeur **R32** et par le routeur **R22**.



- **N.B :** A ce stade, seulement **R32** et **R22** ont une route vers **4.0.0.0/8** (en supposant qu'ils acceptent tous les deux cette route)

BGP : Exemple de base –étape 2.1-

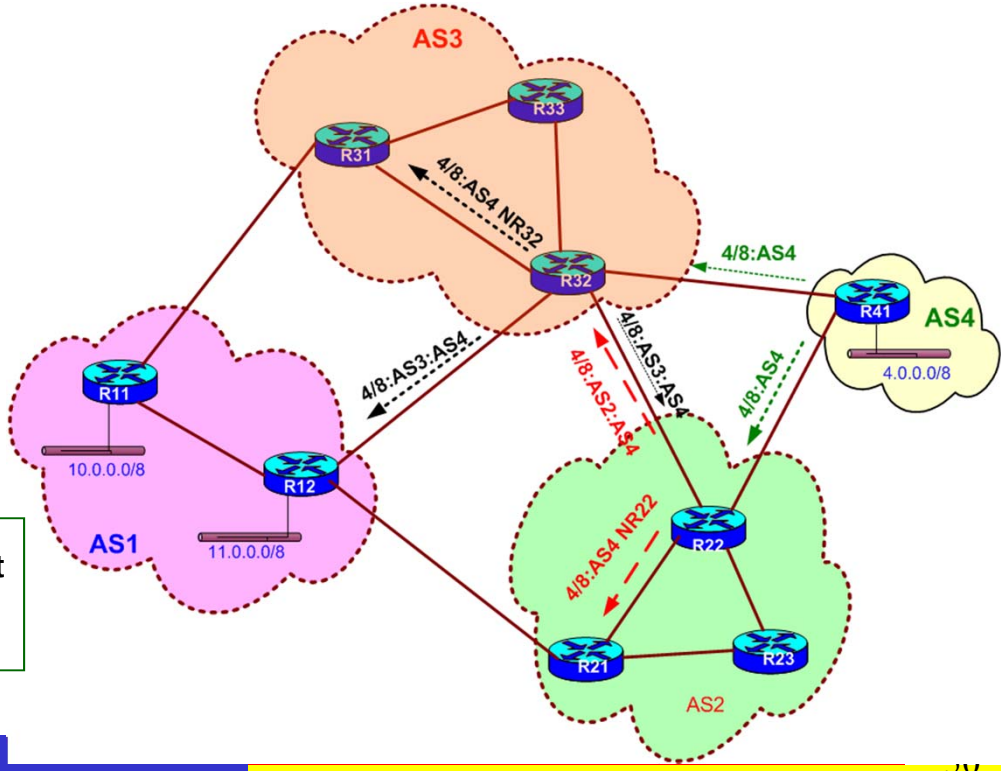
- Chacun des routeurs **R32 et R22** va traiter le message **UPDATE**
  - ✓l'insérer dans leurs **RIB** respectives
  - ✓l 'insérer dans leurs tables de forwarding
  - ✓et redistribuer l'information à leurs voisins.



- **Dans R32 ;**

1. **On annonce la route en direction du routeur R12 qui fait partie de AS1.** Durant cette annonce, on va dire à R12 que cette route est à destination de 4/8 et passe par AS3 et AS4.

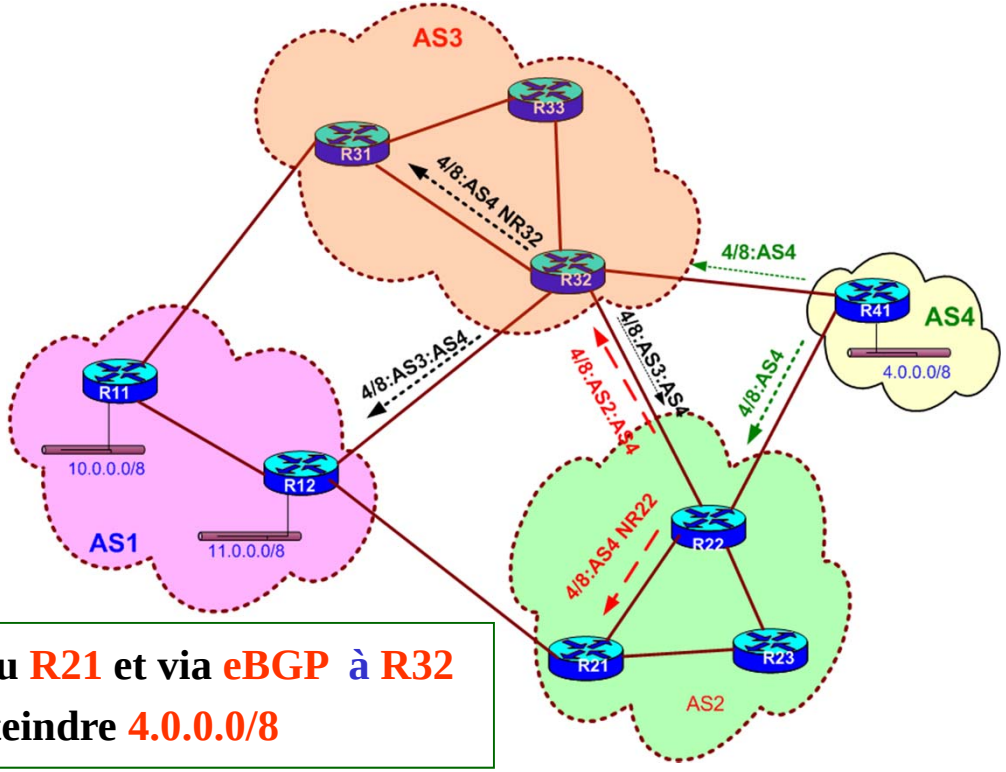
2. **On annonce également cette inforamation à R31 via iBGP.** dans ce cas, R32 va ajouter un attribut spécial à l'intérieur de la route pour informer R31 qu'il est le le **NEXT_HOP** pour cette route (⇔ Si R31 a des paquets à destination de 4/8, il suffira de les confier à R32)



- **R32 annonce ses routes vers 4.0.0.0/8 via iBGP au R31 et via eBGP aux routeurs R12 et R22**

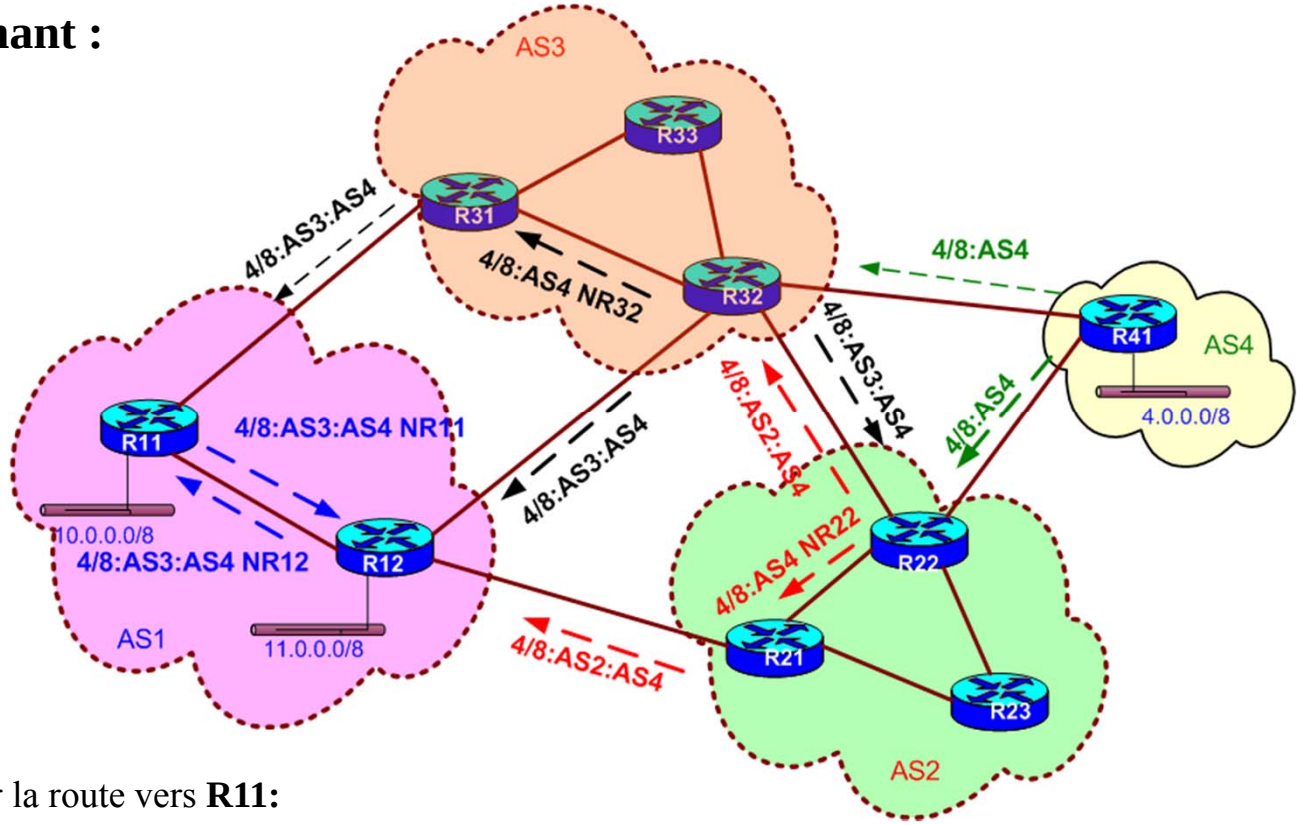
- **Au niveau de R22 ;**

1. **On va rediffuser l'information en direction de R21 avec comme NEXT_HOP : R22**
2. **Elle est aussi annoncé en direction de R32.**
   - Lorsque l'information annoncée par **R22** arrive au niveau de **R32,** ceci ne l'intéresse pas. *Il va préfère utiliser le réseau direct pour joindre le réseau 4/8.*
3. Même chose pour le routeur **R22** qui reçoit la route via **R32**. Cette route ne l'intéresse pas, *il préfère la route directe pour joindre **4/8 via R41**.*



- **R22 annonce ses routes vers 4.0.0.0/8 via iBGP au R21 et via eBGP à R32**
- **R22 continue à préférer le lien R22-R41 pour atteindre 4.0.0.0/8**

- **Dernier élément maintenant :**



- **R31** va annoncer la route
- Ainsi que **R12** qui va ré annoncer la route vers **R11:**
  1. **R31** a reçu la route via **R32**, il va annoncer à **R11** qu'il existe une route en direction de **4/8** qui passe par lui et qui a comme **AS_PATH: AS3:AS4**.
  2. **R12** va faire la même chose en annonçant la route via **iBGP** vers **R11** en disant à R11 qu'il existe une route vers **4/8** qui a comme **AS_PATH :AS3:AS4** et dont le **NEXT_HOP = R12**
- **R21** qui va également annoncer la route

Contrôle de trafic

- Dans la suite, on va présenter les mécanismes qui vont nous permettre de contrôler les trafic qui ***sort du réseau*** et le trafic qui ***rentre dans le réseau***
- Les deux types de mécanismes sont ***complètement différents***

-1-

BGP et le Contrôle de Trafic Sortant

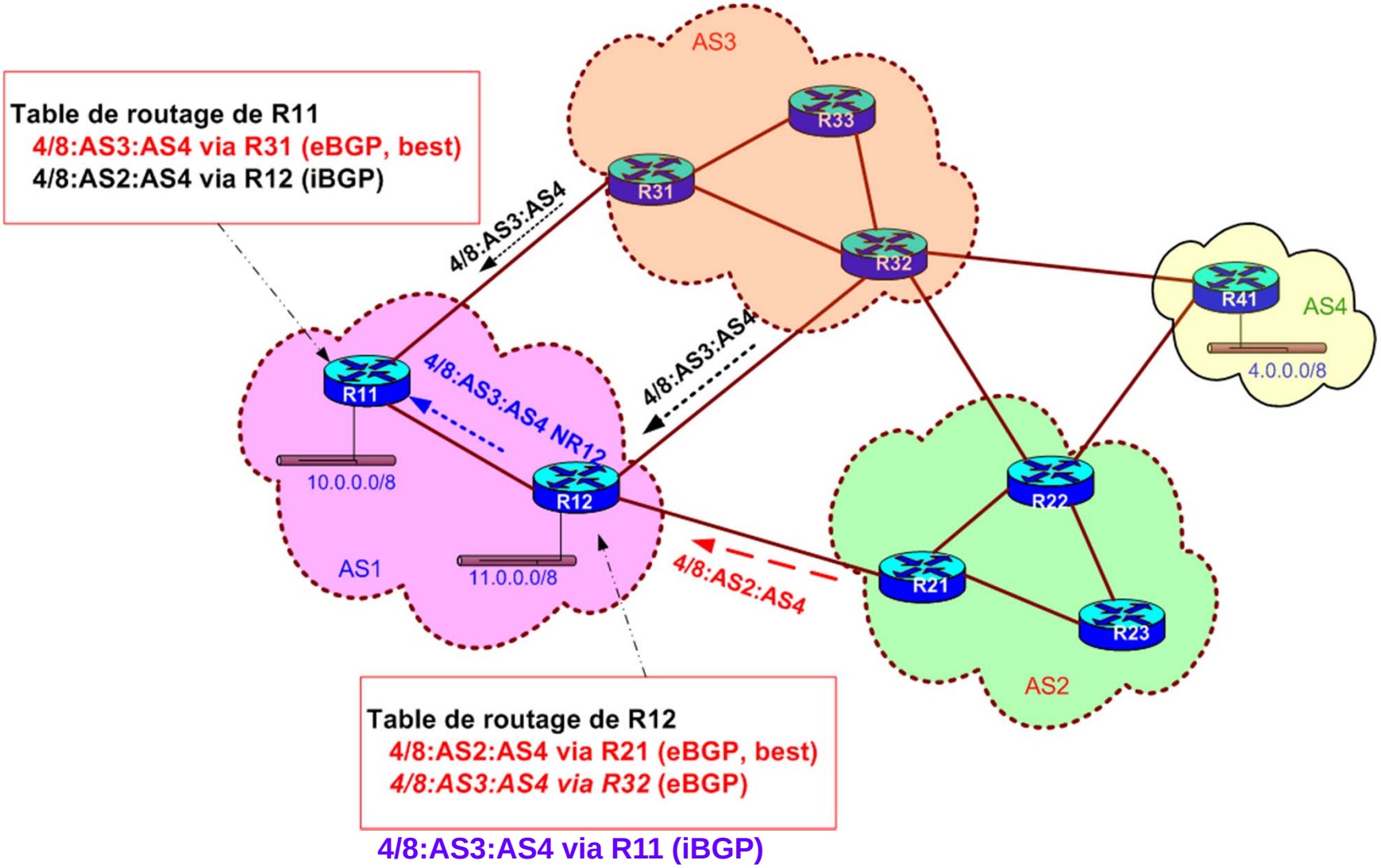
Contrôle de trafic sortant

- **Objectifs : on souhaite par exemple que :**
  - Le trafic sortant passe par plusieurs liaisons externes
  - Ou bien, préférer certaines liaisons par rapport à d'autres, et qui sont des liaisons **moins chères** pour certaines destinations

- **Comment réaliser ce contrôle?**
  - L'administrateur d'AS utilise le **processus BGP** de tous les **routeurs de frontière** pour **influencer la sélection de la meilleur route pour chaque destination.**
  - **On peut s'appuyer sur trois mécanismes :**

1. S'appuie sur des **filtres en entrée** pour ignorer « changer » quelques routes apprises (ou bien changer les attributs des routes qu'on apprises)
2. Ou bien, utiliser les **poids** pour influencer le **choix local** du « meilleur chemin » sur un **routeur particulier (⇔ contrôle de trafic qui affecte un routeur particulier)**
3. Ou bien, utiliser l'attribut **Local-Pref** aux routes annoncées via **iBGP** à l'intérieur du AS à fin d'influencer les autres routeurs de bordure **de l'AS**

- ***Retour à l'exemple précédent :*** Situation avant utilisation du processus de décision BGP



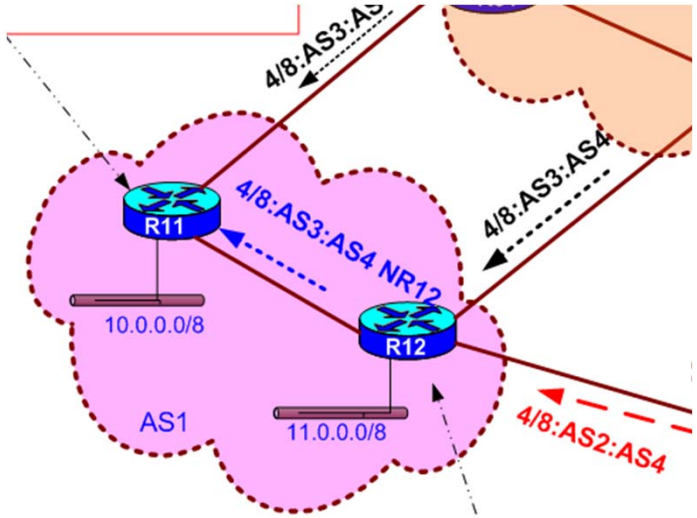
Contrôle de trafic sortant - situation de base -

- **AS2 et AS3 sont des AS de transit :**
- Le but de l'exemple est d'essayer de contrôler ce qui se passe au niveau de **AS1**(**on va se considérer comme administrateur de AS1**)
- Si on laisse les chose fonctionner sans rien contrôler au niveau de **AS1,** le chemin qu'on va utiliser pour joindre le réseau **4/8** va dépendre des annonces que les **routeurs R11 et R12 vont recevoir de l'extérieur**

  R11 va recevoir une annonce de **R31**
  R12 va recevoir des annonces de **R32** et de **R21. Ensuite, R12** va annoncer des informations à **R11** sur la base de ce **qu'il sélectionné lui-même.**

- **Au niveau de R11, on a appris :**

  Une route via **AS3** (via eBGP)
  Une route via **AS2** et qui a été apprise via **R12** (via **iBGP**)

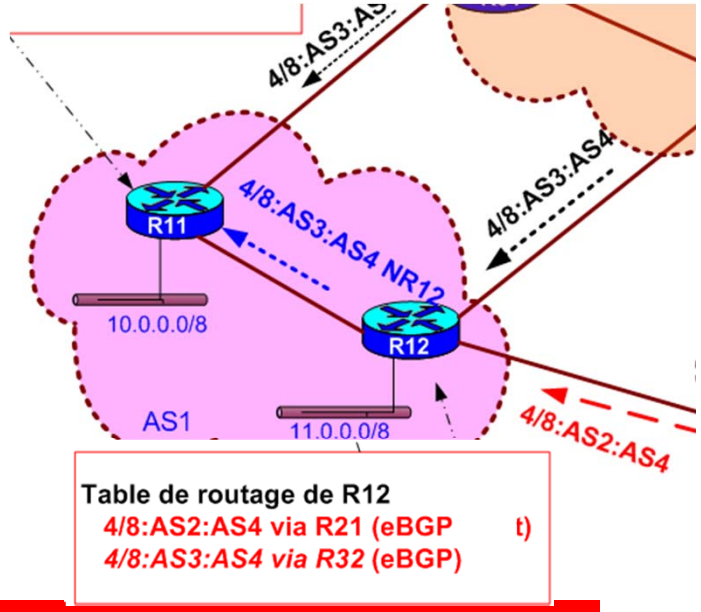


Contrôle de trafic sortant - situation de base -

- **le processus de décision BGP, on va favoriser les routes qu'on a appris via eBGP (pourquoi ?)**
  - ☞ Via **eBGP** ⇔ on va confier le paquet à **transmettre à un autre SA**, donc c'est un autre SA qui va payer pour transmettre le paquet.
  - ☞ Via **iBGP** ⇔ la route est apprise à l'intérieur du réseau => on va payer pour transmettre le paquet

Contrôle de trafic sortant - situation de base -

- **Dans l'état actuel**, le chemin qui sera suivi à destination de **4/8** va dépendre **simplement des annonces qu'on aura reçues de l'extérieur** (⇔ *Donc, ce sont **les autres** qui vont influencer la façon dont on veut faire le trafic)*
  - Dans cet exemple, R_{11} sélectionne la route via R_{31} par ce qu'il préfère la route apprises via eBGP par rapport à celle apprise via iBGP
  - R_{12} a deux routes candidates valides pour atteindre 4.0.0.0/8.
    - Le processus de décision **ne permet pas de sélectionner une route parmi d'autres**.
    - La décision du processus inclut d'autres étapes pour s'assurer qu'une seule meilleur route est sélectionnée. Ces étapes peuvent être par exemple,
      - → l'age de la route (préférer la route la plus ancienne car supposée d'être plus stable que les plus récentes)
      - → Ou bien l'adresse IP du routeur qui a annoncé la route.



- La meilleur route sélectionnée sera **annoncé** par **R_{12}** via **iBGP** en direction de **R_{11}**

Utilisation de l’attribut : « poids »

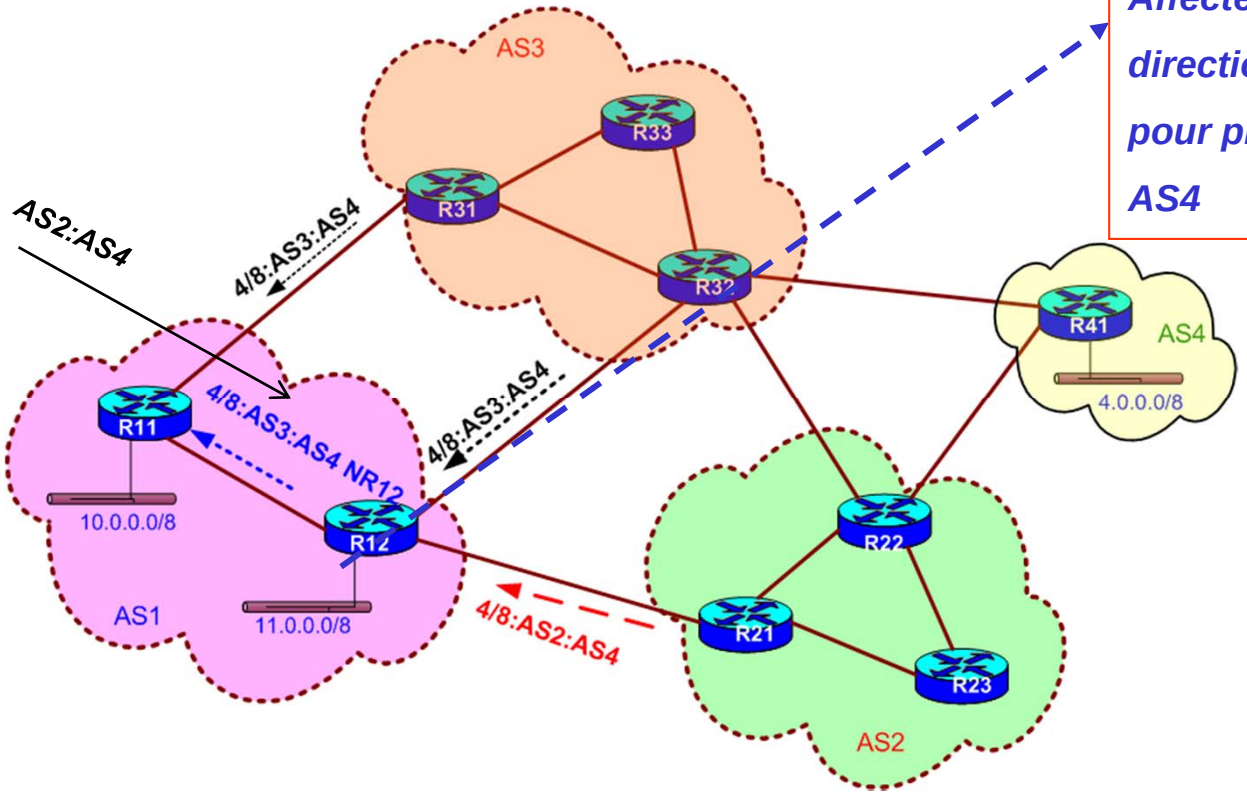
Contrôle de trafic sortant – utilisation de poids -1-

- Le ***poids*** est un attribut qu'on peut ajouter l**ocalement** pour une route au niveau d'un ***routeur particulier***.

- On peut configurer les filtres d'import des routes pour déterminer quelles sont les routes qui vont avoir le ***poids le plus élevé.***
  - Pour cela, on associe un poids à chaque route qui est importée par le routeur, mais ce poids est associé uniquement à l'intérieur d'un routeur (⇔ n'est jamais redistribué)

Poids affectés localement aux routes dans un routeur

Affecter un poids aux routes en direction de AS4 apprises de AS2 pour préférer AS2 pour atteindre AS4



- Sur cette figure, R12 enverra des paquets vers 4.0.0.0/8 par le lien R12-R21 puisque ces annonces ont eu **le plus grand poids**
- **R_{11}** a reçu deux annonces à destination de R_{41}: le premier sur le lien **R_{11}-R_{31}** via **R_{31}** et qui a comme chemin **AS3:AS4 , le second via R12** et qui a comme chemin **AS2:AS4. les deux chemins sont équivalent en terme du nombre d'AS.** Le processus de décision exécuté au niveau de **R_{11}** avantage les routes **eBGP** par rapport aux routes **iBGP**, (⇔ R11 choisit le lien **R11-R31**)

Contrôle de trafic sortant – utilisation de poids –2.2-

- Dans notre exemple (figure précédent), on va configurer le routeur **R12** de la manière suivante :
  - Au niveau de **R12**, associer à la route reçue via **AS2** un **poids plus élevé** et on mettra un **poids plus faible** à la route reçue via **AS3.**
  - De cette façon, lorsque **R12** voit une route reçue via **AS2**, il va la considérer comme étant toujours meilleurs que celle qui passe par **AS3**
  - En effet, dans le processus BGP, on commence par comparer les routes par rapport **aux poids** ⇔ de cette façon, on est sûr que les routes qui ont un poids élevé vont être sélectionnés en premier sur un routeur
- **<u>Attention :</u>**
  - ⇨ Ce choix **n'aura un impact** que au niveau du routeur **R12**
  - ⇨ Si on veut que notre choix affecte aussi **R11**, il va falloir intervenir sur un autre attribut qui est l'attribut **LOCAL_PREF**

Utilisation de l’attribut : « Local_Pref »

Contrôle de trafic sortant –utilisation de Local-Pref -1-

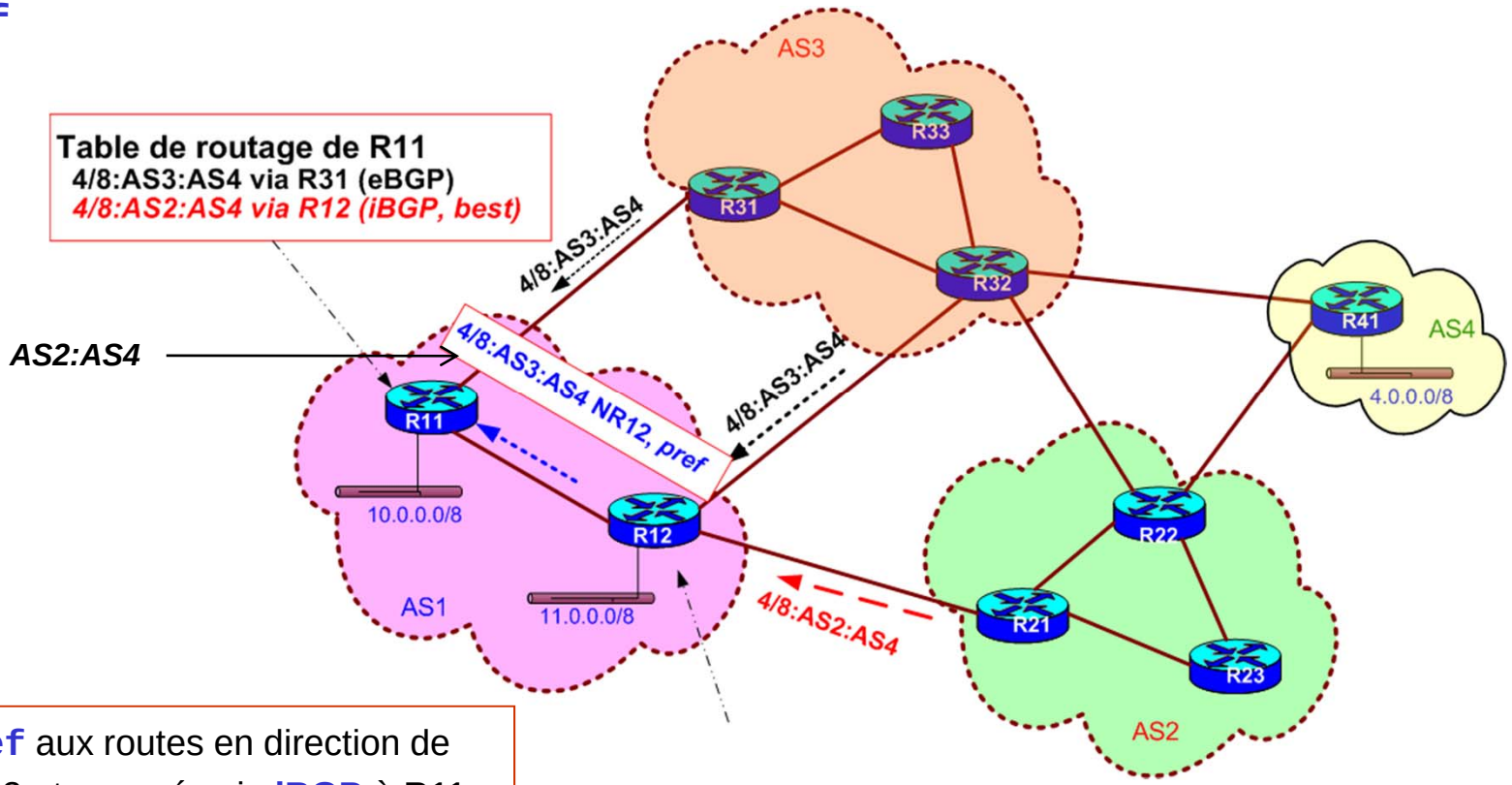
- L'attribut **Local_Pref** va permettre d'annoncer une route qu'on a préféré à tous les routeurs BGP qui **font partie du même SA** ⇔
  - Si un routeur BGP ajoute une préférence **Local_Pref** à une route qu'il reçue (via **eBGP**), **cette attribut va repartir en direction des routeurs** (BGP) **qui font partie du même Système Autonome** (via **iBGP**)

⇨ **Remarque :**

- Lorsque un autre routeur BGP du même SA annonce la route en direction des autres routeurs BGP **des autres SA**, il ne va jamais l'annoncer avec un attribut **Local_Pref**

  ⇔ Donc, il n'existe qu'au niveau de la session **iBGP** à l'intérieur de SA)

- **Local-pref**



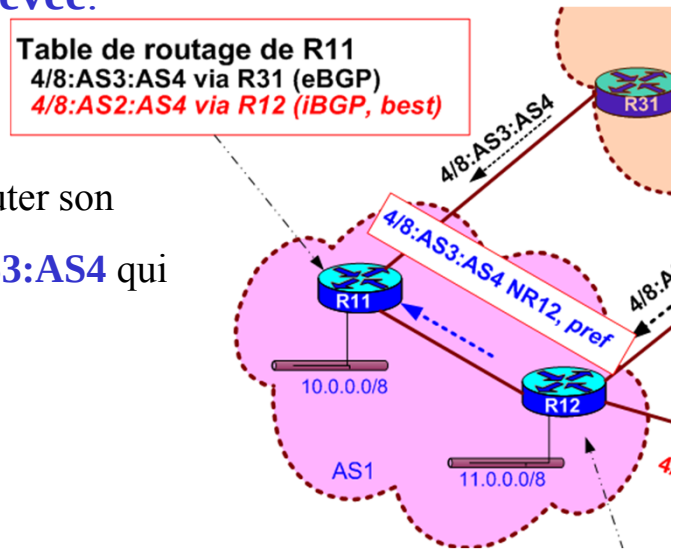
Attacher **local-pref** aux routes en direction de AS4, apprises via AS2 et envoyée via **iBGP** à R11 pour préférer AS2 pour atteindre AS4

- Dans cet exemple, l'attribut local-pref est inséré par R12 avant la distribution de la route via iBGP influencera la décision de routage de tous les routeurs BGP à l'intérieur de AS1.
- Ceci implique que **lorsque** local-pref est employé, une **certaine coordination** parmi les routeurs de bordure est exigé.

Contrôle de trafic sortant –utilisation de Local-Pref –2.2-

⇔ Si R_{12} ajoute une préférence **Local_Pref** à une route qu'il reçue, cette attribut va repartir en direction de **R11**, mais **R11**, lorsqu'il annonce la route en direction de R_{31}, il ne va jamais l'annoncer avec un attribut **Local_Pref**

- Concrètement, au niveau de R_{12}, on va utiliser l'attribut **Local_Pref** de façon à associer une ***préférence élevée*** à la route reçue via **R21** de façon a **passer par AS2**.
  - C'est cette route (via **AS2**) que R_{12} va annoncer à R_{11}
- R_{11} va recevoir via **iBGP** une route vers **4/8** qui passe par AS_2:AS_4 et dont le **NEXT_HOP = R_{12}** (pour sortir de **AS1**) et qui a **une préférence élevée**.



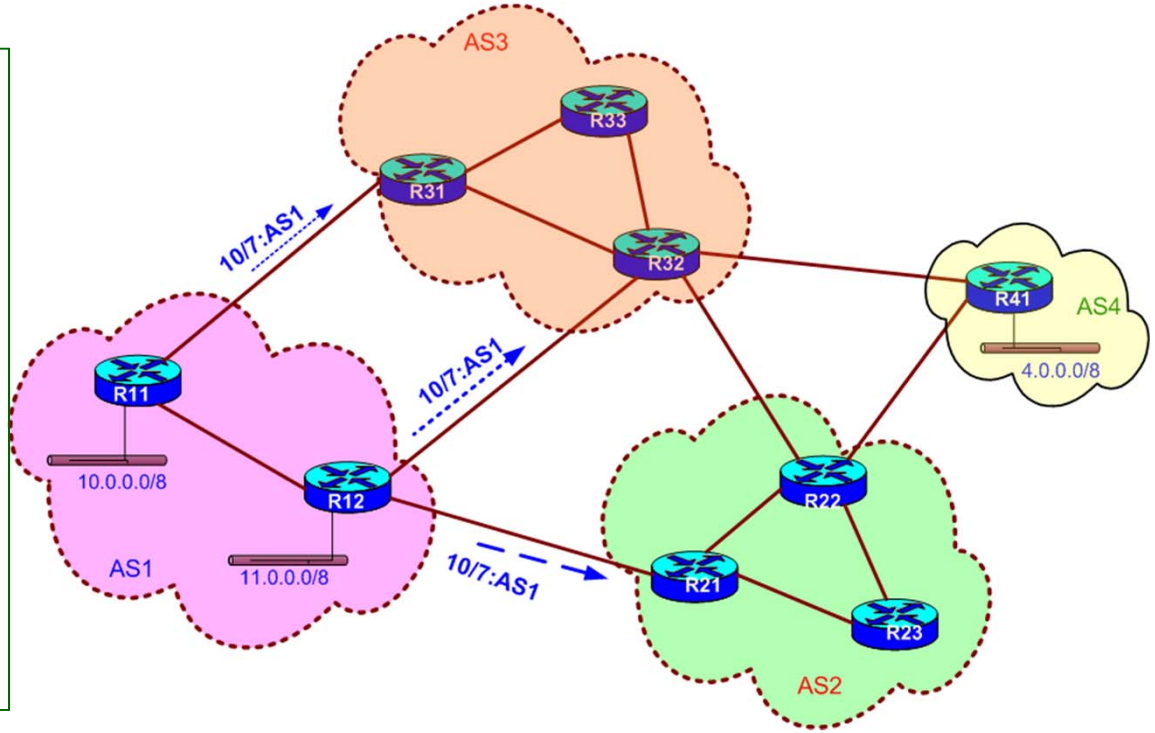
- **<u>Choix de la meilleur route au niveau de R_{11} :</u>**
  - R_{11} va mettre cette route à l'intérieur de sa **RIB-IN**, et lorsqu'il va exécuter son processus de décision, il va comparer cette route avec celle reçue via **AS3:AS4** qui est une route **eBGP** reçue du routeur R_{31}
  - **Dans le processus de décision, on va regarder :**
    1. **Le poids**
    2. La valeur **local_pref** de façon à préférer les routes avec une préférence importante, donc, on va préférer la route reçue de R_{12} par rapport à la route reçue de R_{31}

-2-
BGP et le Contrôle du Trafic Entrant

Contrôle du trafic entrant – exemple –1-

- **Situation avant d'utiliser le processus de BGP :**
  - Les flux des **paquets vers AS1** (⇔ trafic entrant) dépendra du tunning utilisé sur le processus de décision **des Systèmes autonomes AS2, AS3** et **AS4**
  - Pour contrôler le flux de paquets qui entre dans un réseau donné, ***nous devons intervenir sur les annonces envoyées par ce réseau***

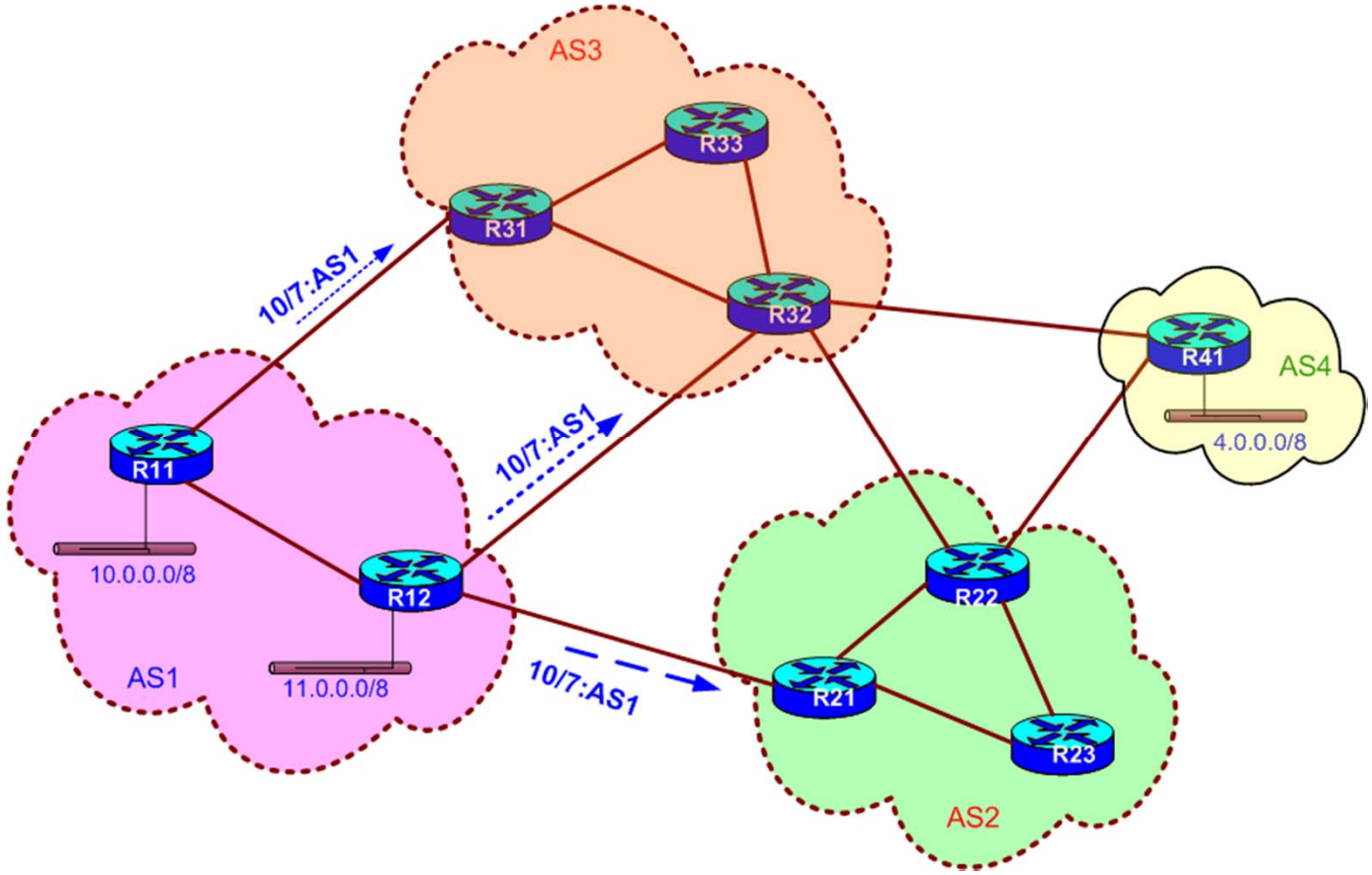
- Pour simplifier les tables de routage, on, essaie d'annoncer les routes de la façon la plus agrégée possible (⇔ on pourra annoncer un réseau **10/7**)
- **Alors dans ce cas**, le chemin via duquel, on va recevoir nos paquets va dépendre simplement de la façon dont nos voisins vont configurer leurs routeurs BGP



Contrôle du trafic entrant – exemple –2-

♦ Si les voisins utilisent l'attribut **poids** ou bien **local_pref**, alors, c'est eux qui vont prendre la **décision** sur le chemin que les paquets vont prendre pour entrer dans notre réseau

⇔ **Si on fait rien, on dépend des voisins**



Contrôle du trafic entrant

« Annonces sélectives »

Contrôle du trafic entrant : Annonces sélectives -1 -

- Annonces Sélective : Qu'est ce qu'on peut faire pour ne pas dépendre des voisins ?

❑ ***solution*** :

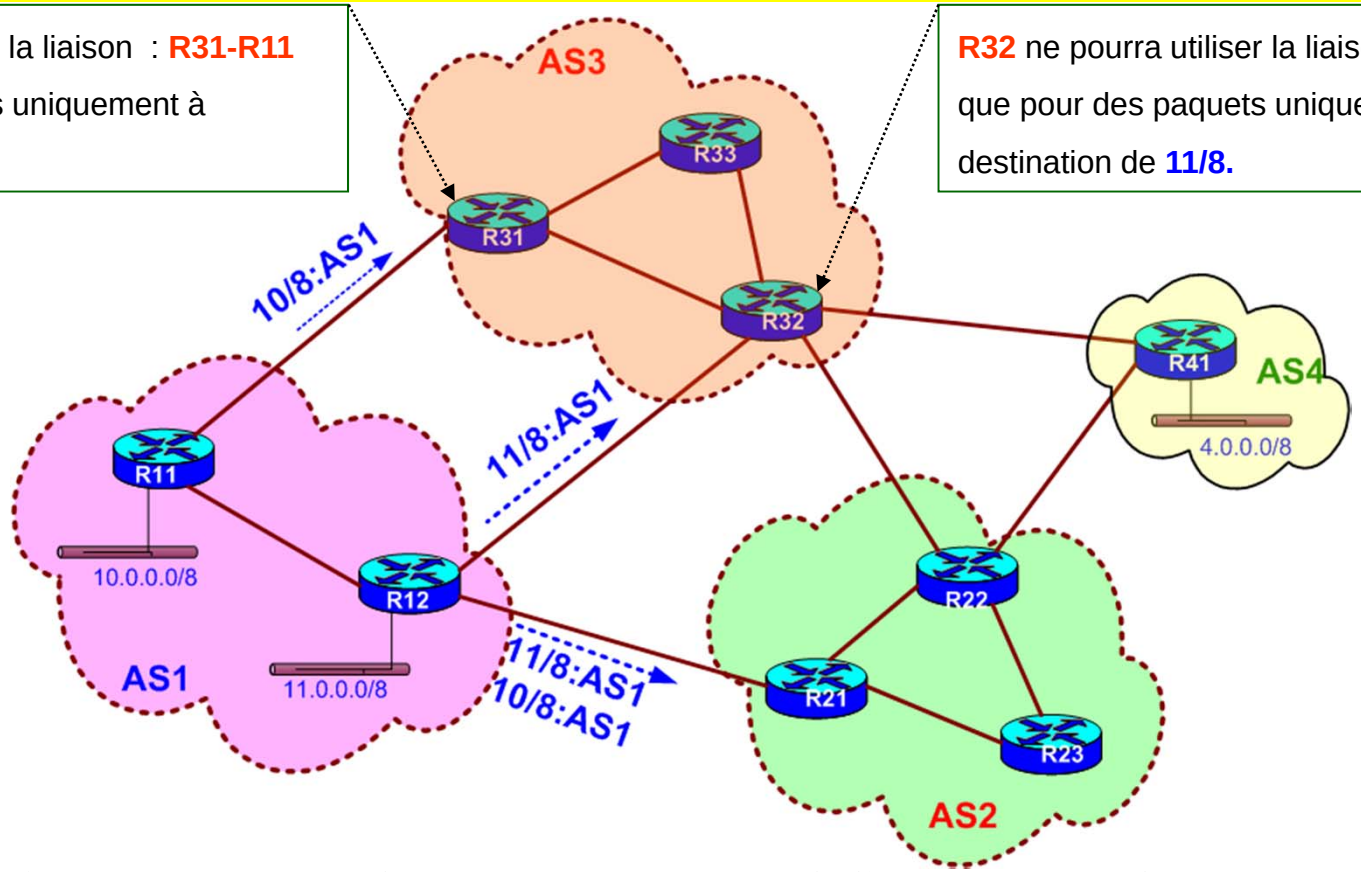
→ Si on veut forcer certains paquets à être reçus via certaines lignes, alors il ne faut pas qu'on annonce toutes nos routes sur toutes nos lignes.

→ Il faut alors, seulement annoncer sur certaines de nos lignes une partie de nos préfixes et pas l'entièreté de nos préfixes.

Contrôle du trafic entrant : Annonces sélectives - 2.0 -

R31 ne pourra utiliser la liaison : **R31-R11** que pour des paquets uniquement à destination de **10/8**

R32 ne pourra utiliser la liaison **R32–R12** que pour des paquets uniquement à destination de **11/8.**



- **<u>Exemple</u> :** **AS1** force **AS3** à envoyer les paquets vers **10/8** sur le lien **R31-R11** et les paquets vers **11/8** sur le lien **R32-R12** .
  - ***<u>N.B :</u>*** *C'est une méthode commune employée pour **équilibrer le trafic** sur les liens externes,*
- **<u>Problème majeur</u> :** Si le lien **R11-R31** tombe, **AS3** ne peut plus utiliser le lien **R12-R32** pour atteindre **10/8** et serait forcé à utiliser le route via **AS2.**

Contrôle du trafic entrant : Annonces sélectives - 2.1 -

- **Dans cet exemple :**
  - dans **AS1,** on a deux **sous réseaux 10/8** et **11/8**. Si on veut recevoir les paquets qui sont à destination de **10/8** via **R11** plutôt que via **R12** et inversement pour les paquets qui sont destinés à **11/8**, alors :

1. **Au niveau de R11 annoncer uniquement le réseau 10/8 sur la liaison R31-R11.**
   - Dans ce cas, R31 ne pourra utiliser la liaison **R31-R11** que pour des paquets uniquement à destination de **10/8**
2. **De même, si R12 annonce uniquement 11/8 sur la liaison vers R32,**
   - Dans ce cas, R32 ne pourra utiliser la liaison **R32–R12** que pour des paquets uniquement à destination de **11/8.**

- **De cette façon,** on va forcer, **AS3** à envoyer des paquets via certaines lignes qui sont des lignes bien choisies.

- **Remarque :** C'est une solution séduisante pour la **répartition de trafic**, par contre moins intéressant pour la **tolérance aux pannes** ⇔ Si on perd la liaison **R31–R11**, on ne pourra plus joindre **10/8** que via **AS2**

On peut faire mieux !!

Contrôle du trafic entrant

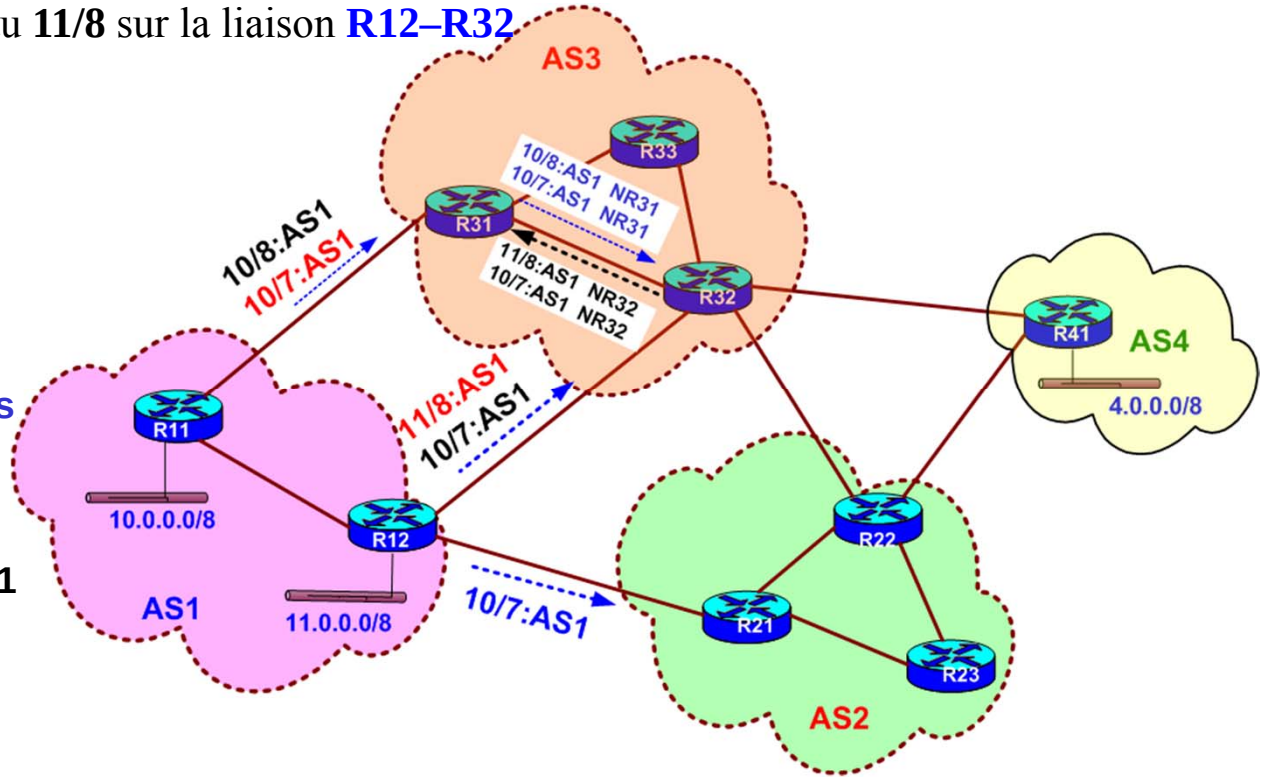
« Le préfixe le plus spécifique »

Contrôle du trafic entrant : Le préfixe le plus spécifique -1 -

- **Le préfixe le plus spécifique : Objectif**
  - On sait qu'avec **BGP**, on peut annoncer une route pour chaque préfixe, mais rien n'empêche d'annoncer des « ***sous préfixes*** » dans un ***préfixe*** déjà annoncé.
- **Idée de base :** annoncer ***un préfixe global sur tous les liens*** pour la **redondance** mais **préférer** quelques liens pour une **partie de ce préfixe.**
  - ➢ Donc, annoncer différentes **routes de recouvrement** sur tous les liens (c.a.d.)
    - le préfixe global d'IP est annoncé sur tous les liens (vers tous les SA)
    - **Sous Préfixe 1** de ce préfixe est seulement annoncé sur **Lien N°1**
    - **Sous Préfixe 2** de ce préfixe est seulement annoncé sur **Lien N°2**
    - …….
- Conclusion : on annonce sur toutes les liaisons, un préfixe globale qui correspond à notre réseau qui couvre l'entièreté de nos destinations, et en plus d'annoncer ça, sur la liaison 1, on va annoncer un premier sous-ensemble de nos préfixes, …etc.

- De cette façon, grâce à l'utilisation du préfixe le plus spécifique, on va indiquer quelle liaison on va utiliser pour recevoir les paquets

- Supposons qu'on veut recevoir via AS3 les paquets à destination de 10/8 via **R11** et à destination de 11/8 via **R12**,
  1. annoncer au niveau de R11 et au niveau de R12 en direction de AS3 le réseau **10/7** sur les deux liaisons (**pour la redondance sur les deux liaisons**).
  2. En plus de ça,
     - annoncer le sous réseau **10/8** sur la liaison **R11–R31**
     - annoncer le sous réseau **11/8** sur la liaison **R12–R32**



Remarque :

- Comparé à l'utilisation des **annonces sélectives**, l'avantage principal d'employer des préfixes plus spécifiques est que si le lien **R11-R31 tombe**, alors les paquets vers **10/8** seront encore envoyés par AS3 via R32

- **Au niveau des tables de routages de R31 et de R32 :**

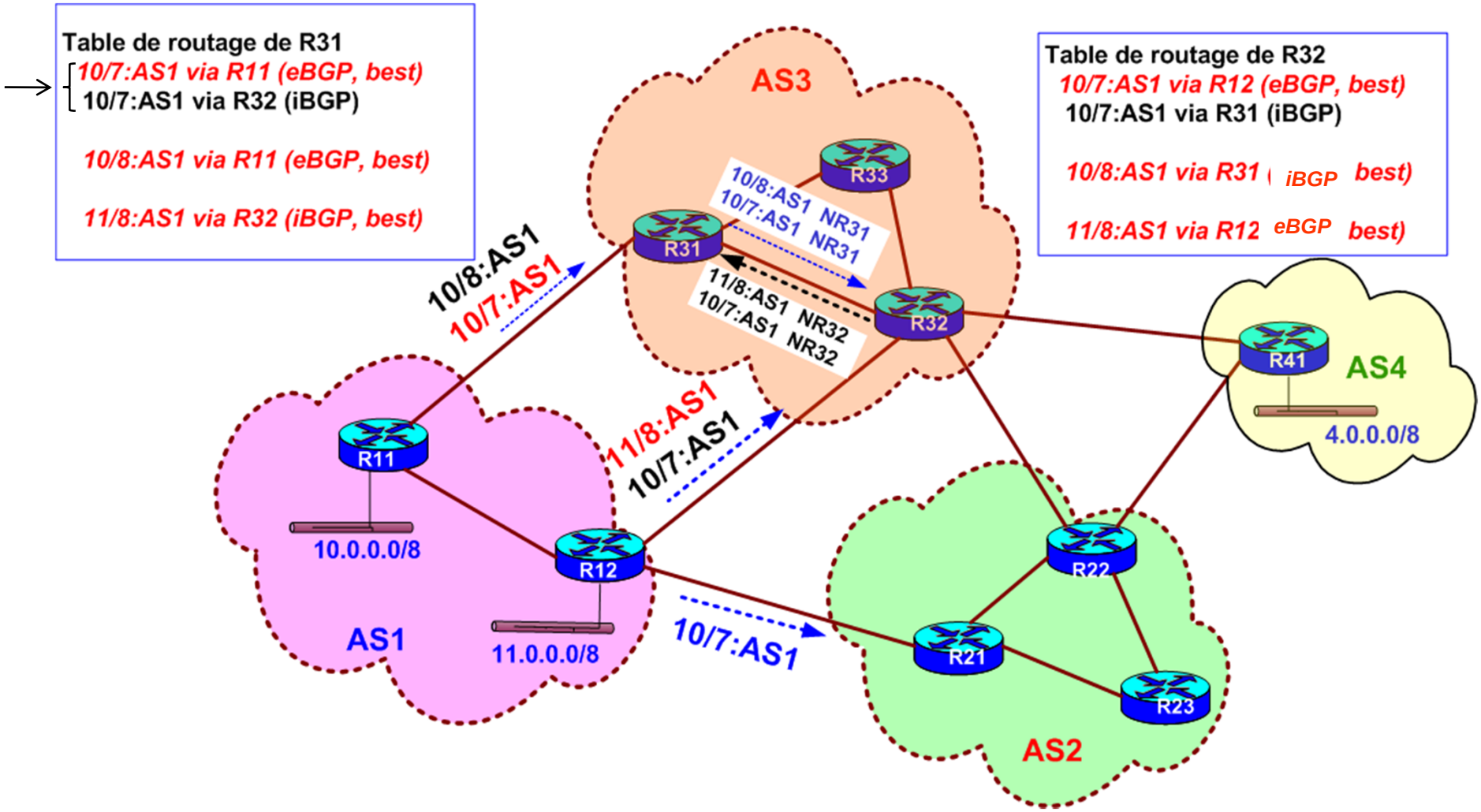


Table de routage de R31
10/7:AS1 via R11 (eBGP, best)
10/7:AS1 via R32 (iBGP)

10/8:AS1 via R11 (eBGP, best)

11/8:AS1 via R32 (iBGP, best)

♦ Au niveau des tables de routages de R31 et de R32 :

1. *Au niveau de R31 :*

- on aura appris deux routes différentes vers **10/7** : une qu'il a appris via **eBGP** via **R11** et une qu'il a appris via **iBGP** via **R32.**
  - Donc, au niveau du processus de décision ⇔ on va préférer ce qui vient de **eBGP**
- Il a aussi appris une route vers **10/8** via **R11**, c'est la seule route qu'il a apprise, donc automatiquement va être annoncée **comme la meilleur**
- Il a également appris via **iBGP** via **R32**, une route en direction de **11/8** et cette route en direction de **11/8**, c'est la seule qu'il a reçue et donc **automatiquement c'est la meilleur**
- ***Donc pour R31,***
  - La meilleure route vers **10/8 via R11.**
  - La meilleure route vers **11/8, via R32-R12**

Table de routage de R32
10/7:AS1 via R12 (eBGP, best)
10/7:AS1 via R31 (iBGP)

10/8:AS1 via R31 iBGP , best)

11/8:AS1 via R1 eBGP , best)

2. C'est la même chose, en sens inverse, au niveau de R32

Contrôle du trafic entrant : Le préfixe le plus spécifique –2.4 -

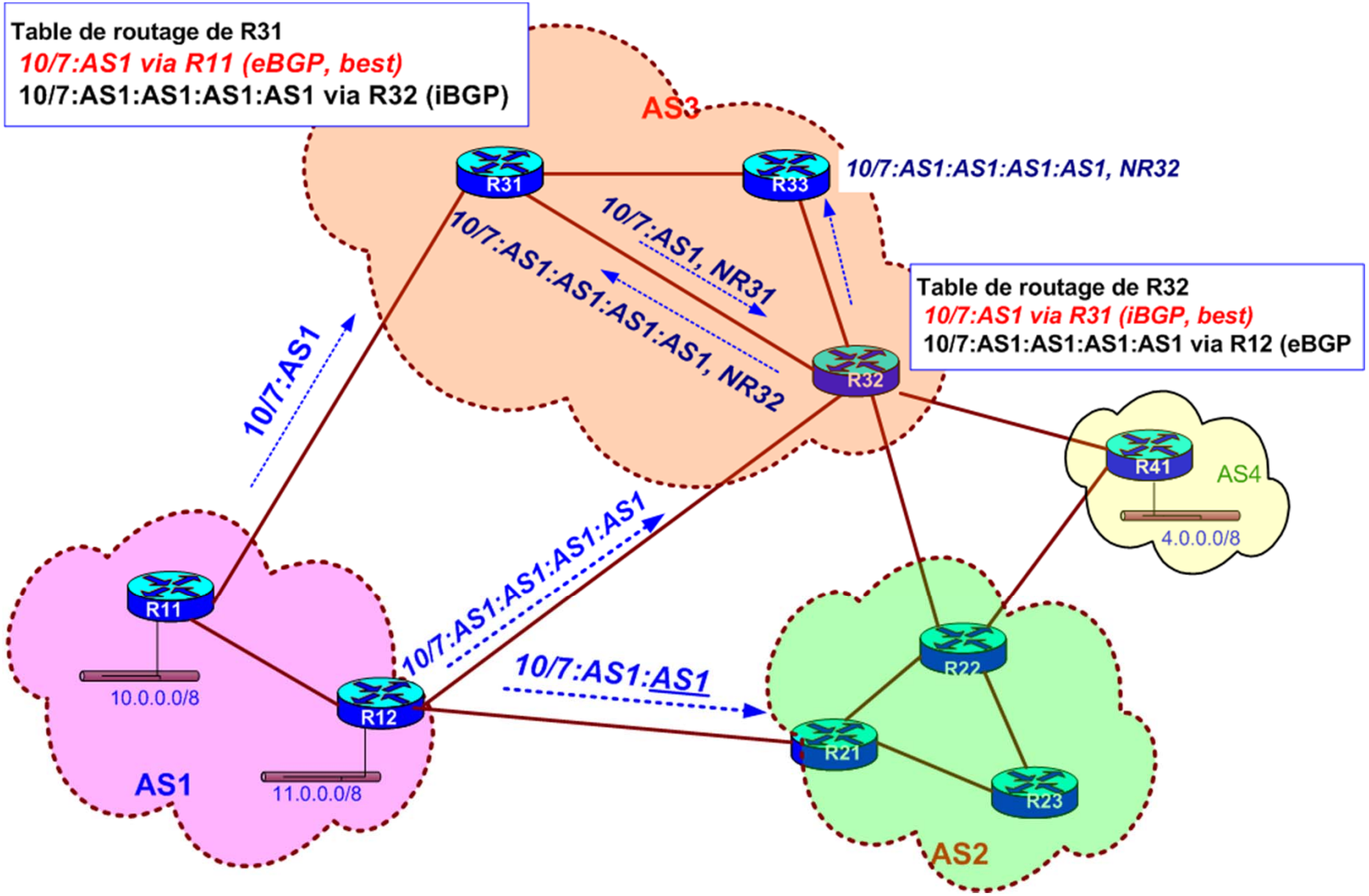
- **<u>Bilan :</u>** Cette solution permet à AS1 de contrôler le trafic en provenance de AS3
- **Inconvénient de cette solution :**
  - pollution des tables de routages (qui sera reflétée sur l'entièreté de l'Internet) : En effet, on va annoncer aux 13 000 SA que AS1 peut joindre 10/7, 10/8 et aussi 11/8
- **Solution :**
  - activation des filtres par certains ISP du type : si vous annoncez des routes plus longues de /22, ils les oublient (certains ISP commencent à filtrer les routes sur base d'allocation des blocs IP par RIPE, par APNIC, …etc.).
- **Remarque :**
  - Ici, ça un impact sur AS3, mais aussi sur tous les SA de l'Internet
  - Le réseau **10/7** dans cet exemple ne servira quasiment à rien

Contrôle du trafic entrant
« AS_Path prepending »

Contrôle du trafic entrant « AS_Path prepending » -1-

- Prepending : Principe de base :
  - Autre méthode qu'on peut utiliser pour contrôler le trafic qui rentre, c'est d'utiliser le troisième critère (comparer la longueur de la AS-PATH des routes)
  - Idée de base : configurer le routeur pour que, sur **certaines liaisons** on annonce une route avec une AS-PATH qui est plus grande que ce qui devrait être (⇔ ***ce qu'on appelle le Prepending***)

Contrôle du trafic entrant « AS_Path prepending » - 2.0 -

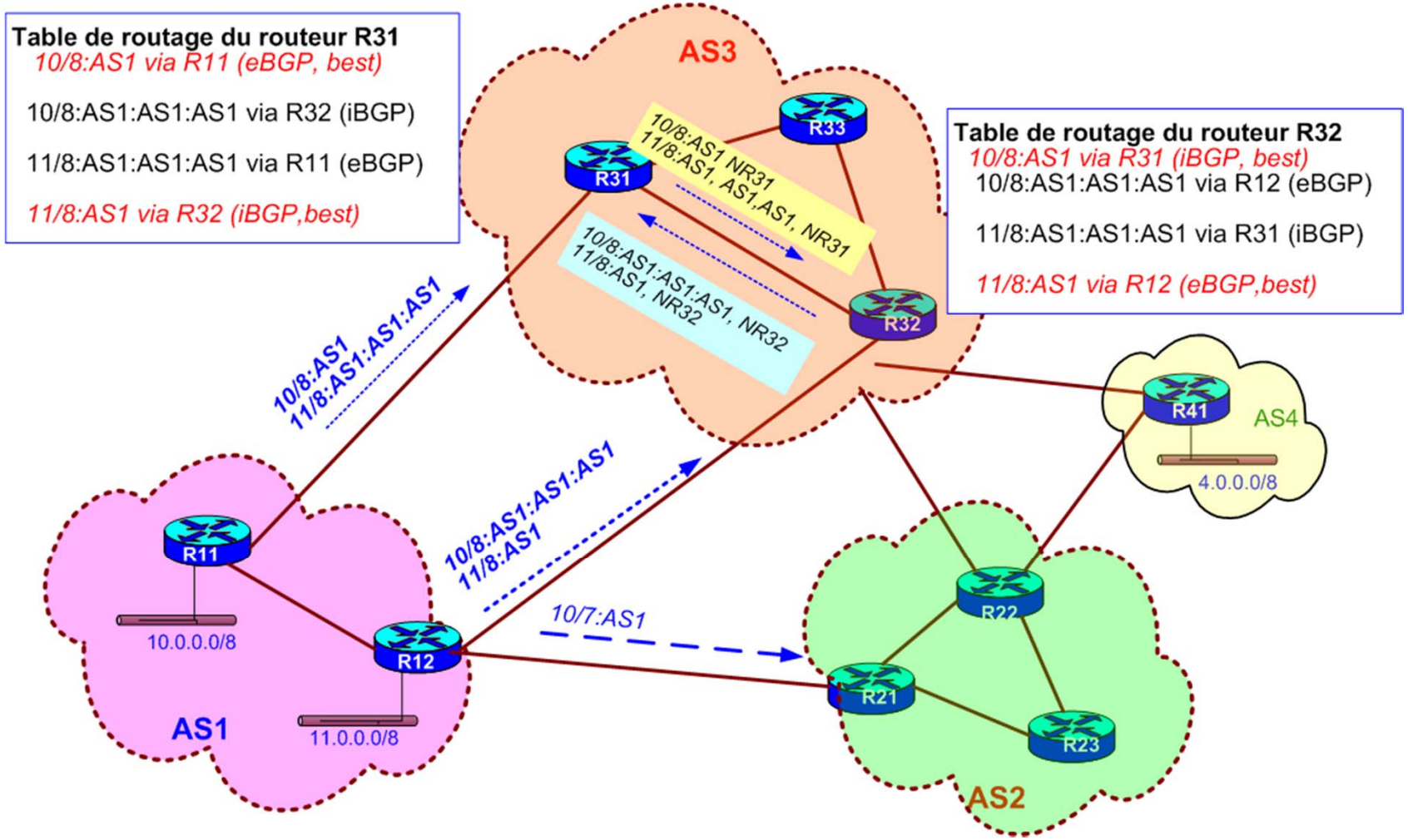


Contrôle du trafic entrant « AS_Path prepending » - 3.1-

- ***Sur l'exemple précédent :***
  - Au niveau de **R12**, si on veut demander à **R32** de ne pas utiliser la liaison **R12-R32**, c'est d'annoncer une route avec un **AS_PATH** qui plutôt de contenir **AS1**, va contenir **4 fois AS1** pour dire à tout le monde sur Internet que cette route est de mauvaise qualité
- ***On peut faire ça aussi sur différentes lignes pour répartir le trafic.***
  - Ceci n'influencera que les systèmes autonomes pour lesquels, le critères **Shortest_ AS_PATH** est un critère **discriminant**.
- ***Si on a des SA importants qui utilisent le poids ou bien le Local_Pref. Le prepending pourrait ne pas influencer ceux-la.***
- **N.B :**
  1. il faut se rappeler qu'une route avec **AS_PATH** court sera toujours préféré sur une route avec **AS_PATH** long (aujourd'hui, les routes qu'on utilise sur Internet vont avoir une taille de 4 à 5 AS en moyen)
  2. Le **Local_Pref** et le **poids** ne sont utiles que lorsqu'on a **un choix à faire**. Donc si on annonce une seule route vers un routeur BGP, il n y aura pas de comparaison à faire

Contrôle du trafic entrant « AS_Path prepending » - 3 -

- On peut combiner l'attribut AS-Path prepending avec l'attribut de Préfixe le plus spécifique (si R31 et R32 n'utilisent Local_Pref et le poids, on va automatiquement sélectionner la route qui a l'AS_PATH la plus court)



Contrôle du trafic entrant

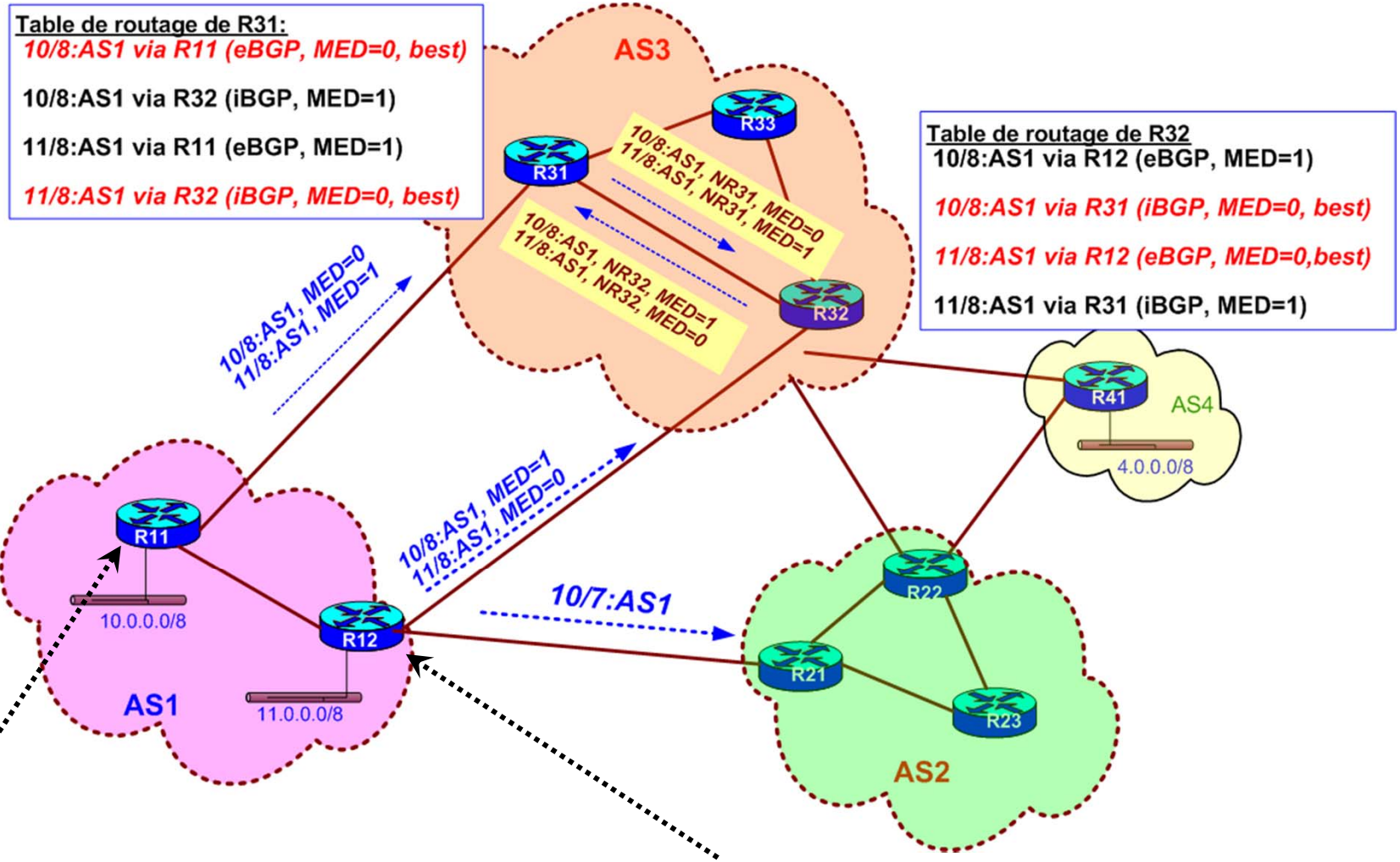
« Multi Exit Discriminator »

Contrôle du trafic entrant « Multi Exit Discriminator » -1-

- Un autre critère sur lequel, on peut intervenir **pour contrôler le trafic entrant** est d'intervenir sur l'attribut **MED**
- ***Idée de base :***
  - Associer l'attribut MED aux routes
  - Celles qui a le MED le plus faible sera sélectionnée comme étant la meilleur route

- **Une chose à noter est que le MED n'a d'intérêt que entre deux AS voisins (⇔ ça ne peut influencer qu'un voisin direct) directs** et qui ont plusieurs **liaisons entre eux** de façon à **repartir le trafic** sur ces différentes liaisons.
  - *Sur notre exemple, c'est le cas entre AS1 et AS3*
- **Principe :**
  - Utiliser l'attribut MED pour annoncer **le coût** (calculé via le protocole de routage intra-domaine qui permet de rejoindre la destination).

Contrôle du trafic entrant « Multi Exit Discriminator » -2.0-



R11 va annoncer :

11/8 se trouve à un coût = 1, donc il va annoncer un **MED** =1

10/8 se trouve à un coût = 0, donc il va annoncer un **MED** =0

R12 va annoncer :

11/8 se trouve à un coût = 0, donc il va annoncer un **MED** =0

10/8 se trouve à un coût = 1, donc il va annoncer un **MED** =1

- ***Sur l'exemple précédent :***
  - **Si R12 peut rejoindre 10/8 et 11/8, il va annoncer**

    11/8 se trouve à un coût = 0, donc il va annoncer un **MED** =0

    10/8 se trouve à un coût = 1, donc il va annoncer un **MED** =1
  - **R11 va faire la même chose, il va annoncer**

    11/8 se trouve à un coût = 0, donc il va annoncer un **MED** =1

    10/8 se trouve à un coût = 1, donc il va annoncer un **MED** =0

- Au niveau de R31 et de R32, lorsqu'on va comparer les routes qui ont été reçues, on va utiliser le **MED** comme discriminant entre les routes
  - on préférera la liaison **R31-R11** pour les paquets à destination de 10/8
  - on préférera la liaison **R32-R12** pour les paquets à destination de 11/8

Control du trafic entrant : Bilan

- **Avantages et inconvénients : Au niveau de contrôle de trafic « entrant », quatre solutions possibles**

1. **Annonces sélectives :**
   - Fonctionne toujours, mais si un seul préfixe est annoncé sur un lien simple, il peut devenir inaccessible en cas **panne sur ce lien**

2. **Des préfixes plus spécifiques :**
   - Est meilleur que les annonces sélectives en cas panne
   - Mais augmente de manière significative la taille de toutes les tables de BGP

3. **Ajouter d'AS-Path prepending :**
   - Peut donner de bon résultat, mais pour cela, il faut être capable de déterminer la Bonne **AS_PATH** à annoncer ce qui est en général difficile à déterminer en pratique

4. **Multi-Exit-Discriminator :**
   - Exige habituellement un accord Bi-directionnel entre les ISP voisins (AS pairs)
   - Seulement utile entre les pairs directs (⇔ n'a pas d'impact que entre voisins)