# Module 7 – BGP route de filtrage et fonctionnalités avancées

**Objectif : En utilisant le réseau configuré dans le Module 6, utiliser diverses méthodes de configuration des peerings BGP pour démontrer un filtrage voisin et des fonctionnalités IOS plus avancées.**

**Prérequis: Module 6**

Topologie :

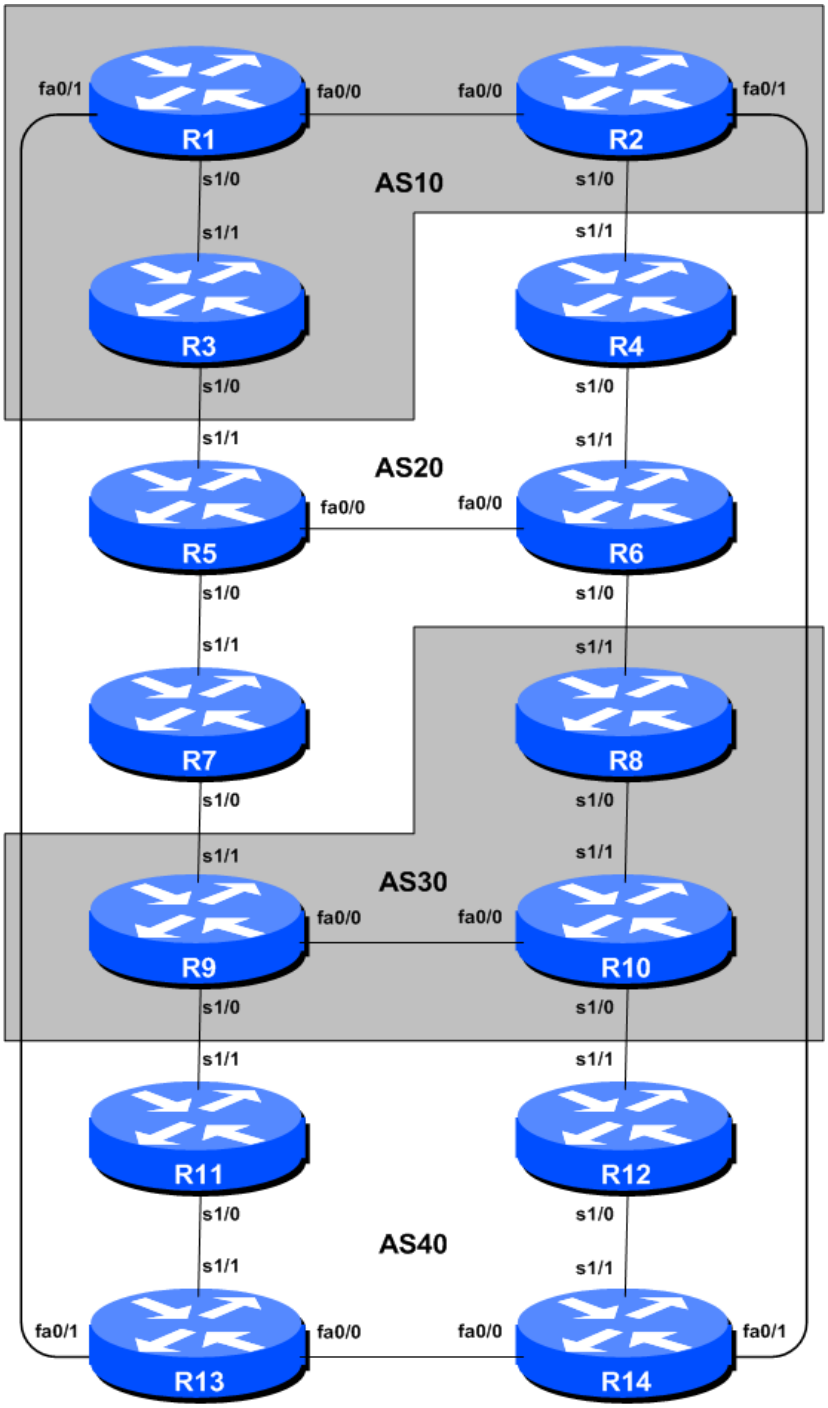



**Figure 1 – nombre de BGP AS**

## *Notes de laboratoire*

Le module précédent a fourni une introduction à la configuration de BGP externe, mais fourni aucun moyen de contrôler quels réseaux est annoncés à laquelle AS. Le but de ce module est d'initier l'étudiant aux types de politique de routage qui sont disponibles en utilisant BGP.

Dans les étapes 2 à 6, nous allons configurer chaque AS pour devenir un **AS non transitaire**, i.e. l'AS ne permet pas une connexion AS du trafic de le traverser pour atteindre une autre connexion AS. Cela signifie déconnexion dans la salle de classe - par exemple, AS3 ne sera plus en mesure de voir les réseaux de AS10, etc. Il s'agit d'une politique délibérée visant à démontrer l'efficacité du filtrage BGP utilisé.

Dans l'étape 2, nous y parvenons en configurant un filtrage de route sortante pour autoriser uniquement les préfixes locaux envoyé aux pairs eBGP. Dans le même temps, nous nous assurons que nos pairs nous envoient seulement leurs propres préfixes. Nous garantirons cela en configurant un filtre entrant. En général, il est recommandé de configurer les filtres dans les deux directions pour se protéger contre une mauvaise configuration à deux extrémités du peering. À l'étape 4, nous utilisons les filtres AS-PATH. Et à l'étape 6 nous utilisons des communautés BGP pour obtenir le même effet.

**Important:** chaque étape doit être réalisée et complétée par tout l'atelier **avant** que l'étape suivante commence. Ne pas commencer immédiatement l'étape suivante sans recevoir le feu vert des instructeurs de l'atelier. Si vous le faites, il est probable que le routage se brise, et les équipes d'autres routeurs peuvent être incapables de comprendre les résultats de la configuration qu'ils essaient de mettre en œuvre.

**Important:** conserver la configuration utilisée pour le module 6..

## *Exercice de laboratoire*

1. **Mettre en œuvre les politiques BGP.** Avant de faire des configurations dans ce module, il est important de noter comment s'y prendre pour la mise en œuvre des politiques BGP. Saisir les listes de *préfixes*, les filtres *as-path* ou *les plans de* route peuvent se faire avec le CLI, mais ils ne s'appliquent qu'aux mises à jour BGP reçus après que la configuration de la politique a été inscrite sur le routeur. C'est parce que BGP envoie des mises à jour incrémentielles, décrivant les changements de routes annoncés ou retirés. Pour appliquer la politique à l'ensemble de la table de routage BGP reçu ou envoyé au pair, la session BGP doit être "réinitialisé". Dans les anciennes versions d'IOS, ceci signifie littéralement détruire la session BGP, puis la restaurer Toutefois, comme on peut imaginer, cela provoque de graves des problèmes de stabilité sur le réseau du fournisseur de service, et 'RFC2918 : capacité de rafraîchir la route' a été ajouté à des implémentations de BGP plus modernes pour permettre les mises à jour aux sessions BGP lorsque les changements de politique sont nécessaires.

   Pour mettre en œuvre des changements de politique dans l'un des exemples suivants travaillés, utilisez les commandes suivantes routeur, par exemple pour mettre en œuvre la nouvelle politique entrant et sortant sur le peering entre routeurs 1 et 13:

```
Router1# clear ip bgp 10.10.15.14 out
Router1# clear ip bgp 10.10.15.14 in
```

**Note 1:** Ne pas oublier les sous-commandes *in* et *out* dans les commandes *clear* ci-dessus - les omettre mettra en œuvre une réinitialisation complete de la session BGP. Revoir la présentation BGP si vous ne comprenez pas pourquoi vous ne voulez jamais faire une réinitialisation matérielle (hard reset) sur une session BGP.

**Note 2:** Au lieu d'actualiser la session BGP en utilisant l'adresse IP voisin, il est également possible de se référer au voisinage par l'ASN. Puisque routeur 13 est en AS40, les commandes de remplacement serait:

```
Router1# clear ip bgp 40 out
Router1# clear ip bgp 40 in
```

***Checkpoint N° 1:*** *appeler l'assistant de laboratoire pour vérifier la connectivité. L'équipe de chaque routeur doit vérifier les peerings pour voir l'effet de cette étape. Utilisez le "show ip bgp" commande pour afficher la table BGP - veiller à ce que les préfixes externes appris par BGP ont maintenant une adresse locale de saut-suivant (next-hop). Utilisez la "trace" de commande pour montrer que la connectivité du réseau n'est pas affectée.*

## *Filtrage à l'aide de listes de préfixes*

2. **Configurer le filtre de préfixes basé sur l'adresse réseau:** cette étape configure le filtrage de route de préfixe basé sur l'adresse du réseau. Ceci est fait en utilisant les listes de préfixes, et c'est une méthode de contrôle des réseaux qui sont échangés dans les peerings BGP. Le but ici c'est de configurer les peerings eBGP de sorte que seuls les réseaux du **voisinage** soit échangés.

**Exemple: Routeur R13 (peering avec R1)**

```
!
ip prefix-list out-peer permit 10.40.0.0/20
ip prefix-list out-peer deny 0.0.0.0/0 le 32
!
ip prefix-list in-peer permit 10.10.0.0/20
ip prefix-list in-peer deny 0.0.0.0/0 le 32
!
router bgp 40
 no synchronization
 network 10.40.0.0 mask 255.255.240.0
 neighbor 10.10.15.13 remote-as 10
 neighbor 10.10.15.13 description eBGP peering with Router1
 neighbor 10.10.15.13 prefix-list out-peer out
 neighbor 10.10.15.13 prefix-list in-peer in
 no auto-summary
!
```

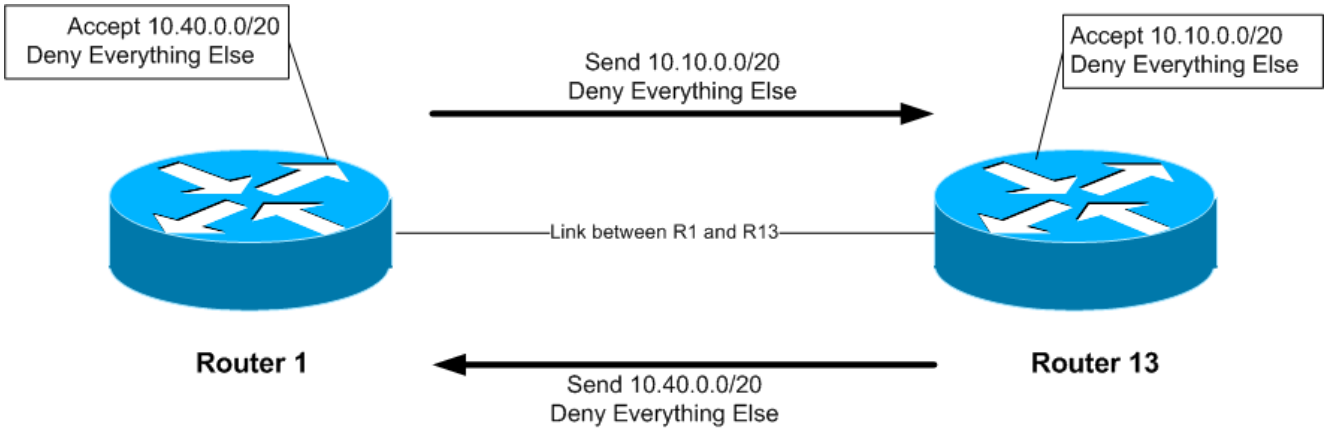
**Exemple: Routeur R1 (peering avec R13)**

```
!
ip prefix-list out-peer permit 10.10.0.0/20
ip prefix-list out-peer deny 0.0.0.0/0 le 32
!
ip prefix-list in-peer permit 10.40.0.0/20
ip prefix-list in-peer deny 0.0.0.0/0 le 32
```

```
!
router bgp 10
 no synchronization
 network 10.10.0.0 mask 255.255.240.0
 neighbor 10.10.15.14 remote-as 40
 neighbor 10.10.15.14 description Peering with Router13
 neighbor 10.10.15.14 prefix-list out-peer out
 neighbor 10.10.15.14 prefix-list in-peer in
 no auto-summary
!
```

**Note :** une liste de préfixes IOS a toujours implicitement un *deny any* comme dernière instruction, même s'il ne figure pas dans la configuration. Certains FAI ajoute l'implicite *deny any* comme ils le jugent bonnes pratiques et une précaution de sécurité.

**Note:** ces listes de préfixes ne s'appliquent qu'aux peerings avec autres AS. Ceux-ci sont appelés peerings **externes** (en utilisant eBGP). Il n'est généralement pas nécessaire d'appliquer de tels filtres pour les peerings iBGP.




***Checkpoint N° 2:*** *appeler l'assistant de laboratoire pour vérifier la connectivité. L'équipe de chaque routeur doit vérifier les peerings pour voir l'effet de cette étape. Utiliser les commandes "show ip bgp neigh x.x.x.x advertise|route"..*

## ARRÊTEZ et ATTENDEZ ICI

3. **Supprimez la configuration de l'exemple précédent.** Cette étape va démontrer comment supprimer la configuration entrée dans l'exemple précédent. C'est essentiel, avant de passer à l'étape suivante.

**Exemple: Routeur R1**

```
Router1#conf t
Router1(config)#router bgp 10
!
! D'abord supprimer la liste de préfixes de peering BGP avec R13
!
Router1(config-router)#no neighbor 10.10.15.14 prefix-list out-peer out
Router1(config-router)#no neighbor 10.10.15.14 prefix-list in-peer in
!
```

```
      ! Maintenant supprimer les listes de préfixes par eux-même
      !
      Router1(config)#no ip prefix-list out-peer
      Router1(config)#no ip prefix-list in-peer
      !
      ! C'est la configuration jolie et soignée, la façon dont il devrait être.
      !
      Router1(config)#end
      !
      ! Maintenant, effacez le peering BGP afin que la politique ancienne soit
supprimée
      !
      Router1#clear ip bgp 40 out
      Router1#clear ip bgp 40 in
      Router1#
```

## *filtres AS-PATH*

4. **Configurer le filtre de préfixes basé sur l'attribut du chemin de AS:** cette étape configure le filtrage de route de préfixe basé sur le chemin AS. Ceci est fait en utilisant le chemin AS des listes d'accès, et une autre méthode de contrôle des réseaux qui sont échangés dans les peerings BGP.

   **Exemple: Routeur R14 (peering avec R2)**

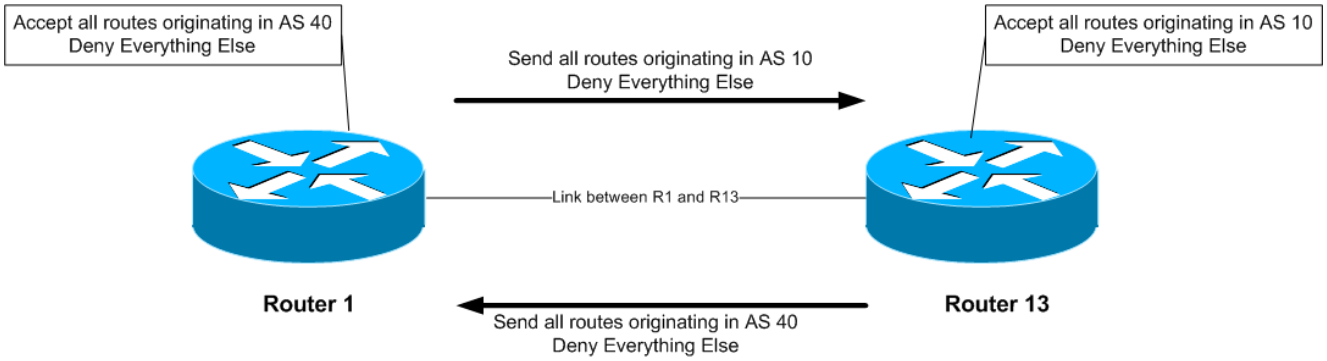
```
ip as-path access-list 2 permit ^$
ip as-path access-list 3 permit ^10$
!
router bgp 40
 neighbor 10.40.15.18 remote-as 10
 neighbor 10.40.15.18 filter-list 2 out
 neighbor 10.40.15.18 filter-list 3 in
!
```

**Exemple: Routeur R2 (peering avec R14)**

```
ip as-path access-list 2 permit ^$
ip as-path access-list 3 permit ^40$
!
router bgp 10
 neighbor 10.40.15.17 remote-as 40
 neighbor 10.40.15.17 filter-list 2 out
 neighbor 10.40.15.17 filter-list 3 in
!
```

**Q.** Pourquoi la liste sortant du filtre correspond à l'as-path nul et non le local numéro d'AS dans les exemples ci-dessus?

**R.** Parce que l'attribut AS-PATH est définie après les listes des préfixes, les filtres as-path et les plan de route. Si l'AS local a été inclus dans la liste de configuration de filtre sortant, les préfixes seraient ignorés comme l'attribut AS-PATH ne serait pas réglé à ce stade.




Pour vérifier que l'expression régulière fonctionne comme prévu, utilisez la commande EXEC "***show ip bgp regexp*** *<regular-expression>*" pour afficher tous les chemins qui correspondent à l'expression régulière spécifiée. N'oubliez pas que la commande *clear ip bgp <neighbour address> [in | out]* est nécessaire pour mettre en œuvre ce filtre.

Notez que les filtres de chemin AS permettent tous les préfixes émis par le voisinage AS – dans l'exemple précédent utilisant les filtres de préfixe, seulement l'agrégat du voisinage AS traverse les filtres.

***Checkpoint N° 3:*** *appeler l'assistant de laboratoire pour vérifier la connectivité. Une fois que l'instructeur de laboratoire donne le feu vert à la classe, supprimez l'attribut de configuration et la liste de filtres et passez à l'étape suivante.*

## ARRÊTEZ et ATTENDEZ ICI

5. **Supprimez la configuration de l'exemple précédent.** Cette étape va démontrer comment supprimer la configuration entrée dans l'exemple précédent. C'est essentiel, avant de passer à l'étape suivante.

**Exemple: Routeur 2**

```
Router1#conf t
Router1(config)#router bgp 10
!
! D'abord supprimer la liste de filtre de peering BGP avec R14
!
Router1(config-router)#no neighbor 10.40.15.17 filter-list 2 out
Router1(config-router)#no neighbor 10.40.15.17 filter-list 3 in
!
! Maintenant supprimer les listes de filtre par eux-même
!
Router1(config)#no ip as-path access-list 2
Router1(config)#no ip as-path access-list 3
!
! C'est la configuration jolie et soignée, la façon dont il devrait être.
!
Router1(config)#end
!
! Maintenant, effacez le peering BGP afin que la politique ancienne soit
supprimée
```

```
!
Router1#clear ip bgp 10.40.15.17 in
Router1#clear ip bgp 10.40.15.17 out
Router1#
```

## *Communautés BGP pour le filtrage (et route-map)*

6. **Introduction à BGP communautés route-map:** Cette étape introduit les équipes de routeur à l'aide des communautés BGP pour le marquage, l'identification, et éventuellement de filtrage des préfixes. Nous obtenons des résultats finaux similaires à ce que nous avons réalisé dans les étapes précédentes en utilisant le prefix-list et les filtres as-path .

   Sur tous les routeurs, configurez BGP pour envoyer une communauté pour ***tous les préfixes*** appartenant à l'AS local annoncé pour les pairs BGP externes. La communauté doit être sous la forme de *[numéro d'AS]: [numéro de routeur]*. Par exemple, Router8 doit utiliser la communauté 30:8 .

   **Exemple sur Routeur R8:**

```
! Afficher communautés dans 16-bits :16-bit de format plutôt que
! comme un entier 32 bits.
!
ip bgp-community new-format
!
ip prefix-list out-match permit 10.30.0.0/20 le 26
!
route-map outfilter permit 10
 match ip address prefix-list out-match
 set community 30:8
!
router bgp 30
 neighbor 10.20.15.17 remote-as 20
 neighbor 10.20.15.17 route-map outfilter out
 neighbor 10.20.15.17 send-community
!
```

   **Note:**
   1) Un attribut de communauté peut être vue dans la table BGP via *show ip bgp <network>* commande.
   2) Un attribut de communauté est un champ de 32 bits. Par convention convenue d'IETF il est divisé dans deux champs de 16 bits pour faciliter l'interprétation. Le top 16 bits contient le numéro d'AS, tandis que la partie inférieure de 16 bits représente un nombre entier qui a une signification particulière pour les deux AS impliqués dans le peering. Les exceptions à cette règle sont les chaînes bien connues des attributs de communauté tels que ***no-export*** ou ***local - as***.

   **Q:** Pourquoi *send-community* est nécessaire pour les peerings eBGP?
   **R:** Comme les valeurs de communauté ne sont pas transmises par défaut entre les pairs BGP, vous devez indiquer au routeur de le faire explicitement.

***Checkpoint N° 4:*** *appeler l'assistant de laboratoire pour vérifier la connectivité. L'équipe de chaque routeur doit vérifier à nouveau leur peerings pour voir quel est l'effet cette fois-ci.*

## ARRÊTEZ et ATTENDEZ ICI

7. **Supprimez la configuration de l'exemple précédent.** Cette étape va démontrer comment supprimer la configuration entrée dans l'exemple précédent. C'est essentiel, avant de passer à l'étape suivante.

   **Exemple: Routeur 9**

```
Router8#conf t
Router8(config)#router bgp 30
!
! D'abord supprimer le plan de route de peering eBGP
!
Router8(config-router)#no neighbor x.x.x.x route-map outfilter out
!
! Maintenant supprimer le plan de route lui-même
!
Router8(config)#no route-map outfilter
!
! Et maintenant retirer la liste de préfixes utilisée par le plan de route
!
Router8(config)#no ip prefix-list out-match
!
! C'est la configuration jolie et soignée, la façon dont il devrait être.
!
Router8(config)#end
!
! Maintenant, effacez le peering BGP afin que la politique ancienne soit
supprimée
!
Router8#clear ip bgp 10.20.15.17 out
Router9#
```

## *BGP communautés*

8. **La définition de communautés BGP.** À l'étape 6 le réseau de la communauté appartient a été générée à l'endroit où un routeur BGP a parlé à un autre routeur BGP. Si cette situation est valable pour démontrer comment définir les communautés, le scénario le plus courant est celui où un FAI attache une communauté à un réseau lorsque le réseau est injecté dans la table de routage BGP.

   Chaque équipe de routeur doit assigner une communauté au bloc du ***réseau du client*** (le /26) qui provient de BGP. Consultez la documentation BGP pour savoir comment faire. Chaque routeur doit définir une communauté de format *[numéro d'AS]: [numéro de routeur]* exactement comme dans l'étape précédente.

   **Exemple pour Routeur R3:**

```
ip bgp-community new-format
!
```

```
route-map community-tag permit 10
 set community 10:3
!
router bgp 10
 no synchronization
 network 10.10.0.128 mask 255.255.255.192 route-map community-tag
 neighbor 10.20.15.1 remote-as 20
 neighbor 10.20.15.1 send-community
 no auto-summary
!
ip route 10.10.0.128 255.255.255.192 null0
```

Vérifiez que le réseau apparaît avec sa communauté dans la table de routage BGP.

**Q:** Pourquoi faire votre externe, mais pas votre intérieur, les pairs voient la communauté sur le réseau?
**R:** Voir plus haut. Tous les peerings exigent que la sous-commande *send-community* de BGP pour l'attribut communauté à être envoyé entre les pairs BGP.

9. **Communautés sur les peerings BGP internes.** Suite à l'étape précédente, maintenant configurer les peerings internes afin que l'attribut de communauté de votre réseau soit envoyé à ses pairs locaux.

   **Astuce:** Pour ce faire, il suffit d'ajouter dans la ligne de configuration *neighbor x.x.x.x send-community* pour tous les peerings iBGP. N'oubliez pas d'actualiser les sessions BGP de peering afin que le changement de configuration puisse être mis en œuvre.

***Checkpoint N° 5:*** *Appelez l'assistant de laboratoire et démontrez comment la communauté a été configurée pour votre réseau en utilisant la commande "show ip bgp". En outre, démontrez que vous pouvez voir les communautés fixées par vos pairs BGP internes et externes.*

10. **Configurez le filtre de préfixe entrant sur la base d'attributs de communauté.** Le but ici est de n'accepter que les réseaux qui sont reçus par le voisinage des pairs BGP externes (Cela ressemble à ce qui a été tenté aux étapes 2 et 4 avec le préfixe et chemin de filtrage AS). Par exemple, R13 ne devrait accepter que le réseau créé par R1, et doit utiliser la connaissance de la communauté R1 relié au réseau afin d'atteindre cet objectif.

    **Exemple sur Routeur R13:**

```
ip community-list 3 permit 10:1
!
route-map infilter permit 10
 match community 3
!
router bgp 40
 neighbor 10.10.15.13 route-map infilter in
!
```

Le choix du numéro de la liste de communauté appartient à chaque équipe de routeur – il n'est pas annoncé dans n'importe quel peering BGP ou utilisé d'une autre manière en dehors de l'identifiant de la liste de communauté (Comparez avec le numéro de la liste d'accès).

11. **Réglez l'attribut de la préférence locale sur les routes eBGP reçues.** Dans cet exemple, la préférence locale se règle sur les routes correspondant au filtre de communauté à l'étape 10. Conservez le plan de route utilisé à l'étape 10 - une ligne de configuration supplémentaire lui sera ajouté. Vous voulez aussi permettre aux autres réseaux entendus à travers les filtres avec la préférence locale fixée à la valeur par défaut.

    **Q.** Pourquoi?
    **R.** Sans la deuxième directive d'autorisation le plan de route met en œuvre un refus par défaut et aucun autre préfixe ne passera à travers.

    **Exemple:**

```
route-map infilter permit 10
 match community 3
 set local-preference 120
!
route-map infilter permit 20
```

    Notez qu'après une nouvelle politique est définie, la session BGP doit être actualisée afin que la nouvelle politique puisse alors être appliquée. Le routeur ne tient pas automatiquement toutes les mises à jour qu'il reçoit du pair donc c'est nécessaire. Cela se fait à l'aide de la commande exec "***clear ip bgp*** <*peer address*> ***in***".

***Checkpoint N° 6:*** *Appelez l'assistant de laboratoire et de démontrez comment les routes émises par vos pairs eBGP maintenant ont la préférence locale définie sur 120. Montrez également comment les autres routes ont la préférence locale par défaut de 100.*

## *Peer-groups BGP*

12. **Configurez la fonctionnalité de peer-group pour les pairs iBGP.**-les groupes de pairs BGP aident à réduire la charge de processeur du routeur dans la génération et l'envoi de mises à jour de pairs qui ont la même politique. Cette étape configure les groupes de pair BGP pour les voisinages iBGP dans chaque AS. Remplacer la configuration individuelle pour chaque pair avec une configuration de groupes de pair iBGP, comme indiqué dans l'exemple ci-dessous.

    **Exemple pour Routeur R9:**

```
router bgp 30
 neighbor ibgp-peers peer-group
 neighbor ibgp-peers description Peer-Group used for all iBGP peers
 neighbor ibgp-peers remote-as 30
 neighbor ibgp-peers update-source loopback 0
 neighbor ibgp-peers send-community
 neighbot ibgp-peers next-hop-self
 neighbor ibgp-peers password cisco
!
```

    **Note :** l'ancienne configuration de pre-*peer-group* **doit** être retirée avant la conversion de la configuration de non-*peer-group* en utilisant les peer-groups.

```
router bgp 30
 no neighbor 10.30.15.224
```

```
 no neighbor 10.30.15.226
!
 neighbor 10.30.15.224 peer-group ibgp-peers
 neighbor 10.30.15.226 peer-group ibgp-peers
!
```

**Q:** Quels sont les avantages de l'utilisation de *peer-groups*?

**R:** Les groupes de pairs BGP permettent une configuration courante à utiliser pour plusieurs pairs BGP. L'application la plus courante est pour iBGP. Tous les pairs BGP internes dans un réseau de FAI ont tendance à avoir la même relation entre eux. Plutôt que d'avoir une configuration substantielle par les pairs, et d'avoir à changer tous les peering lorsque les détails doivent être modifiés, la configuration peut être mise dans un groupe de pairs, et que le groupe de pairs doit être modifié pour modifier la configuration de peering pour tous les pairs iBGP. Ceci réduit considérablement la surcharge de travail nécessaire pour apporter des changements, la CPU du routeur pour le traitement et la nettoie de manière significative la configuration pour voir.

Il est fortement recommandé que la sous-commande *peer-group* soit le moyen par "défaut" de configurer tous les pairs BGP. Comme il a été indiqué précédemment, la plupart des pairs iBGP ont la même configuration, il est donc d'une grande utilité pour le routeur, le personnel opérationnel et les ingénieurs réseaux de simplifier les configurations chaque fois que possible. En outre, une configuration qui fait un usage intensif des groupes de pairs est généralement beaucoup plus facile à lire que celui qui a une configuration distincte par les pairs, en particulier dans les réseaux avec un grand nombre de pairs.

**Note: à l'avenir, chaque fois que BGP est configuré dans l'atelier, il est prévu que les groupes de pairs seront utilisées dans le cadre de la configuration BGP (et très certainement pour n'importe quelle configuration iBGP).**

13. **Résumé :** ce module présente les fonctionnalités de base disponibles pour configurer les peerings BGP dans Cisco IOS. Le lecteur est invité à essayer de nouvelles permutations des exemples de configuration données ici. Utilisation de la communauté gagne en popularité que la fonctionnalité est désormais reconnue pour donner des avantages considérables dans le contrôle de la politique de routage entre les différents AS. L'actualisation de route et les groupes de pairs sont aussi largement utilisés dans les backbones des FAI comme ils rendent considérablement faciliter l'administration et la configuration d'un réseau opérationnel.

## *Questions de révision:*

1. Pourquoi l'utilisation de Route Actualiser (Route Refresh) est la meilleure façon de mettre en œuvre la nouvelle politique BGP?

2. Pourquoi les filtres AS-PATH fournissent-ils moins granularité que les filtres de préfixe pour filtrer une session BGP? Et qui est recommandé dans un réseau du FAI et pourquoi?

3. Quand l'attribut de la communauté devrait-il être fixé sur un préfixe BGP?

4. IOS envoie-t-il des communautés BGP par défaut pour iBGP? pour eBGP? Sinon, qu’est-ce que les opérateurs doivent retenir à faire?