BGP - Control de Politicas

Aplicando Políticas con BGP

- Política – basada en el AS-Path, comunidad o prefijo.
- Rechazar/aceptar rutas determinadas.
- Asignar atributos para influenciar la selección de la trayectoria.
- Herramientas:
  - Prefix-list (Filtra los prefijos)
  - Filter-list (Filtra los ASs)
  - Route-maps y comunidades

Control de Políticas – Prefix List

- Configuración incremental.
- Aplicado a la entrada o la salida.
- Basado en números de red (utilizando el formato familiar de dirección IP/Máscara).
- Prefix-list termina con un "denegar" implícito por defecto.
- Utilizar access-lists en los IOS de Cisco para filtrar prefijos se hizo obsoleto hace tiempo.
  - **No recomendado!**

Prefix Lists – Command Syntax

- Sintaxis:

  [no] ip[v6] prefix-list *list-name* [seq *value*]
  permit|deny *network/len* [ge *value*] [le *value*]

  ***network/len*:** El prefijo y su longitud

  ge *value*: "mayor o igual que"

  le *value*: "menor o igual que"

- Ambos "ge" y "le" son opcionales
  - Usado para especificar el rango de la longitud del prefijo a coincidir para los prefijos que son más específicos que red/longitud.

- El número de secuencia también es opcional
  - **no ip[v6] prefix-list sequence-number** desactiva la muestra de números de secuencia.

Listas de Prefijos – Ejemplos

- Denegar ruta por defecto

```
ip prefix-list EG deny 0.0.0.0/0
```

- Permitir el prefijo 35.0.0.0/8

```
ip prefix-list EG permit 35.0.0.0/8
```

- Denegar el prefijo 172.16.0.0/12

```
ip prefix-list EG deny 172.16.0.0/12
```

- En 192/8 permitir hasta /24

```
ip prefix-list EG permit 192.0.0.0/8 le 24
```

  - Esto permite todos las longitudes de prefijo en el bloque 192.0.0.0/8, excepto /25, /26, /27, /28, /29, /30, /31 and /32.

Listas de Prefijos – Ejemplos

- En 192/8 denegar /25 y superiores

```
ip prefix-list EG deny 192.0.0.0/8 ge 25
```

  - Esto niega todas las longitudes de prefijo /25, /26, / 27, /28, /29, /30, /31 y /32 en el bloque 192.0.0.0/8.
  - Efecto idéntico al ejemplo anterior.
- En 193/8 permite prefijos entre /12 and /20

```
ip prefix-list EG permit 193.0.0.0/8 ge 12 le 20
```

  - Esto niega todas las longitudes de prefijo /8, /9, /10, / 11, /21, /22, … y superiores en el bloque 193.0.0.0/8.
- Permit todos los prefijos

```
ip prefix-list EG permit 0.0.0.0/0 le 32
```

  - 0.0.0.0 coincide con todas las posibles direcciones, "0 le 32" coincide con todas las longitudes de prefijo.

Prefix Lists – Ejemplo Completo

- Ejemplo de configuracion

```
router bgp 100
 network 105.7.0.0 mask 255.255.0.0
 neighbor 102.10.1.1 remote-as 110
 neighbor 102.10.1.1 prefix-list AS110-IN in
 neighbor 102.10.1.1 prefix-list AS110-OUT out
!
ip prefix-list AS110-IN deny 218.10.0.0/16
ip prefix-list AS110-IN permit 0.0.0.0/0 le 32
!
ip prefix-list AS110-OUT permit 105.7.0.0/16
ip prefix-list AS110-OUT deny 0.0.0.0/0 le 32
```

Control de Políticas – Filter List

- Filtrar rutas basado en AS-Path
  - Entrada o salida
- Se hace referencia en la configuración en BGP neighbour como:

```
neighbor <addr> filter-list <N> [in|out]
```

- Se hace referencia en la configuración principal como:

```
ip as-path access-list <N> [permit|deny] ...
```

- La lista de acceso As-Path termina con una denegacion implicita por defecto.

Filter List – Ejemplo

- Ejemplo de configuración:

```
router bgp 100
 network 105.7.0.0 mask 255.255.0.0
 neighbor 102.10.1.1 filter-list 5 out
 neighbor 102.10.1.1 filter-list 6 in
!
ip as-path access-list 5 permit ^200$
!
ip as-path access-list 6 permit ^150$
```

Control de Políticas – Expresiones Regulares

- Como las expresiones regulares de Unix
  - . Coincidir con un carácter
  - * Cualquier número de ocurrencias de la expresión anterior.
  - + Una ocurrencia de la expresión anterior
  - ^ Principio de línea
  - $ Fin de línea
  - \ Escapar un carácter de expresión regular
  - _ Inicio, fin, espacio, braqueta
  - | O
  - () paréntesis para contener expresión
  - [] braquetas para rangos numéricos

Control de Políticas – Expresiones Regulares

- Ejemplos simples

.*	Coincidir con cualquier cosa
.+	Coincidir al menos con un carácter
^$	Rutas locales a este AS
_1800$	Originadas por AS1800
^1800_	Recibidas desde AS1800
_1800_	via AS1800
_790_1800_	via AS1800 y AS790
_(1800_)+	multiple AS1800 en secuencia (para AS-PATH prepends)
_\(65530\)_	via AS65530 (confederaciones)

Control de Políticas – Expresiones Regulares

- Ejemplos no tan simples

^[0-9]+$	Coincidir con AS_PATH de longitud 1
^[0-9]+_[0-9]+$	AS_PATH de longitud 2
^[0-9]*_[0-9]+$	AS_PATH longitud 1 ó 2
^[0-9]*_[0-9]*$	AS_PATH longitud 0, 1 ó 2
^[0-9]+_[0-9]+_[0-9]+$	AS_PATH de longitud 3
_(701|1800)_	Cualquier prefijo que haya pasado por AS701 o AS1800
_1849(_.+_)12163$	Origen en AS12163 y pasando por AS1849

Control de Políticas – Mapas de Rutas

- Un route-map es como un "programa" en IOS
- Tiene números de línea, como los programas
- Cada línea es una condición/acción distinta
- El concepto es, básicamente: :
  - Si hay coincidencia, ejecutar expresión y terminar
  - Si no
  - Si hay coincidencia, ejecutar expresión y terminar
  - Si no, etc
- El comando "continue" permite aplicar múltiples condiciones y acciones en un mismo route-map

Route Maps – Reglas

- Una línea puede tener múltiples comandos "set"
  - Todas las declaraciones "set" se implementan

```
route-map SAMPLE permit 10
 set community 300:1
 set local-preference 120
!
```

- Una línea puede tener múltiples "match"
  - Todas las condiciones deben coincidir "match"

```
route-map SAMPLE permit 10
 match community 1
 match ip address prefix-list MY-LIST
 set local-preference 300
!
```

Route Maps – Reglas

- Una declaración de "match" puede tener varios comandos
  - Al menos un comando debe coincidir

```
route-map SAMPLE permit 10
 match ip address prefix-list MY-LIST OTHER-LIST
 set community 300:10
!
```

- Route-map con solo un "match"
  - Sólo pasan los prefijos que coinciden, y los demás son descartados

```
route-map SAMPLE permit 10
 match ip address prefix-list MY-LIST
!
```

Route Maps – Reglas

- Línea con solamente una declaración “set”
  - Todos los prefijos coinciden y se les asigna algo
  - Las líneas siguientes se ignoran

```
route-map SAMPLE permit 10
 set local-preference 120
!
route-map SAMPLE permit 20
 remark This line is ignored
 set community 300:5
!
```

Route Maps – Reglas

- Línea con match/set y sin más líneas
  - Sólo se asigna algo a los prefijos que coinciden, y el resto se descarta

```
route-map SAMPLE permit 10
 match ip address prefix-list MY-LIST
 set local-preference 120
!
```

Route Maps – Advertencias

- Ejemplo
  - Omitir la tercera línea abajo significa que los prefijos que no coincidan list-one o list-two se descartan

```
route-map SAMPLE permit 10
 match ip address prefix-list LIST-ONE
 set local-preference 120
!
route-map SAMPLE permit 20
 match ip address prefix-list LIST-TWO
 set local-preference 80
!
route-map SAMPLE permit 30
 remark Don't forget this
!
```

Route Maps – Coincidencia de prefijos

- Ejemplo de configuración:

```
router bgp 100
 neighbor 1.1.1.1 route-map INFILTER in
!
route-map INFILTER permit 10
 match ip address prefix-list HIGH-PREF
 set local-preference 120
!
route-map INFILTER permit 20
 match ip address prefix-list LOW-PREF
 set local-preference 80
!
ip prefix-list HIGH-PREF permit 10.0.0.0/8
ip prefix-list LOW-PREF permit 20.0.0.0/8
```

Route Maps – Coincidencia de prefijos

- Comentario:
  - Si la dirección coincide con HIGH-PREF coloque local-pref 120, y luego salir.
  - De lo contrario si la dirección coincide con LOW-PREF, coloque local-pref 80, y luego salir.
  - Ninguna otra condición, por lo que todos los demás prefijos se eliminan.

Route Maps – AS-PATH filtering

- Ejemplo de configuración

```
router bgp 100
 neighbor 102.10.1.2 remote-as 200
 neighbor 102.10.1.2 route-map FILTER-ON-ASPATH in
!
route-map FILTER-ON-ASPATH permit 10
 match as-path 1
 set local-preference 80
!
route-map FILTER-ON-ASPATH permit 20
 match as-path 2
 set local-preference 200
!
ip as-path access-list 1 permit _150$
ip as-path access-list 2 permit _210_
```

Route Maps – AS-PATH filtering

- Comentario:
  - Si el prefijo originado desde AS150, entonces, establezca local pref a 80, y la salir.
  - Otherwise if prefix transited AS210 (ie AS210 appears in the path), then set local-pref to 200, and exit
  - De lo contrario, si el prefijo transitó AS210 (es decir AS210 aparece en el camino), entonces, establezca local pref a 200, y la salir.
  - Ninguna otra condición, por lo que todos los demás prefijos se eliminan.

Route Maps – AS-PATH prepends

- Ejemplo de configuracion de AS-PATH prepend

```
router bgp 100
 network 105.7.0.0 mask 255.255.0.0
 neighbor 102.10.1.2 remote-as 300
 neighbor 102.10.1.2 route-map SETPATH out
!
route-map SETPATH permit 10
 set as-path prepend 100 100
!
```

- Utilice su **propio** número de AS cuando haga "prepending"
  - De lo contrario, la detección de bucles de BGP puede causar desconexiones
  - Inserción deliberada de otros ASN se llama "envenenamiento AS PATH"

Route Maps – Coincidir Comunidades

- Ejemplo de configuracion

```
router bgp 100
 neighbor 102.10.1.2 remote-as 200
 neighbor 102.10.1.2 route-map FILTER-ON-COMMUNITY in
!
route-map FILTER-ON-COMMUNITY permit 10
 match community 1
 set local-preference 50
!
route-map FILTER-ON-COMMUNITY permit 20
 match community 2 exact-match
 set local-preference 200
!
ip community-list 1 permit 150:3 200:5
ip community-list 2 permit 88:6
```

Route Maps – Coincidir Comunidades

- Comentario:
  - Si prefijo pertenece a las comunidades 150: 3 Y 200: 5, entonces, establezca local pref a 50, y la salir.
  - De lo contrario, si el prefijo pertenece SOLO a la comunidad 88: 6, entonces, establezca local pref a 200, y la salir.
  - Ninguna otra condición, por lo que todos los demás prefijos se eliminan.

Comunidad - Procesamiento de Lista

- Nota:
  - Cuando varios valores se configuran en la mismo declaracion de lista comunitaria, se crea una condición lógica "AND". Todos los valores de la comunidad deben coincidir para satisfacer una condición "AND".

```
ip community-list 1 permit 150:3 200:5
```

  - Cuando varios valores se configuran en declaraciones lista comunitaria separadas, se crea una condición lógica "OR". La primera lista que coincide con una condición se procesa.

```
ip community-list 1 permit 150:3
ip community-list 1 permit 200:5
```

Route Maps – Asignar Comunidades

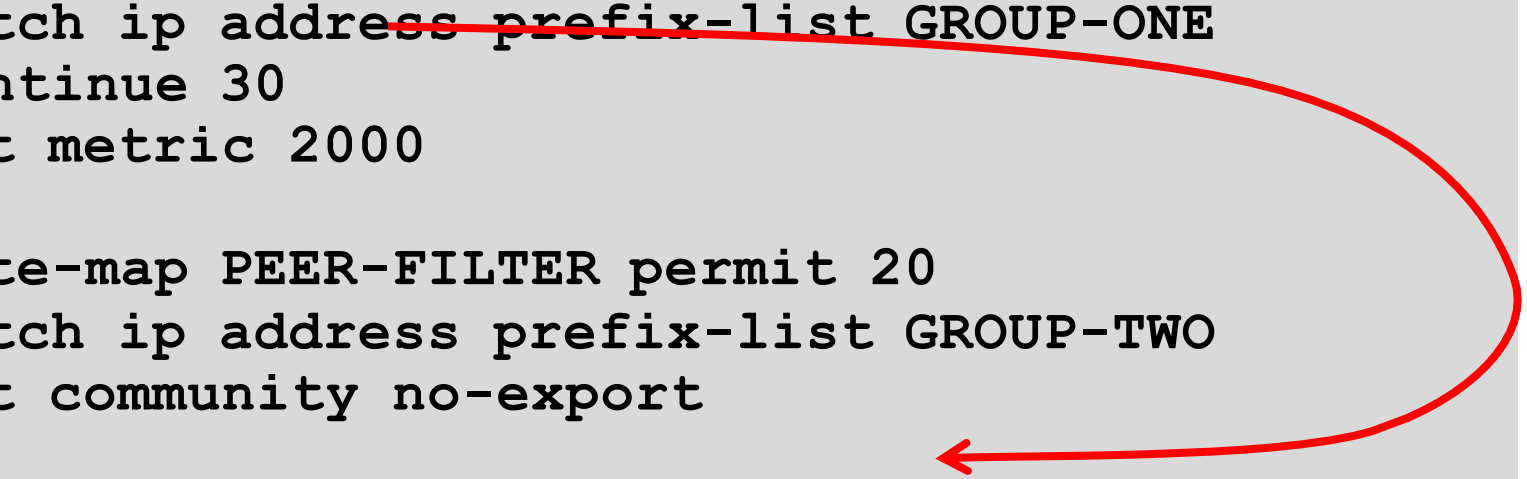
- Ejemplo de Configuracion

```
router bgp 100
 network 105.7.0.0 mask 255.255.0.0
 neighbor 102.10.1.1 remote-as 200
 neighbor 102.10.1.1 send-community
 neighbor 102.10.1.1 route-map SET-COMMUNITY out
!
route-map SET-COMMUNITY permit 10
 match ip address prefix-list NO-ANNOUNCE
 set community no-export
!
route-map SET-COMMUNITY permit 20
 match ip address prefix-list AGGREGATE
!
ip prefix-list NO-ANNOUNCE permit 105.7.0.0/16 ge 17
ip prefix-list AGGREGATE permit 105.7.0.0/16
```

Route Map - Comando Continue

- Múltiples condiciones y acciones en un mismo route-map (para relaciones de vecinos BGP solamente).

```
route-map PEER-FILTER permit 10
 match ip address prefix-list GROUP-ONE
 continue 30
 set metric 2000
!
route-map PEER-FILTER permit 20
 match ip address prefix-list GROUP-TWO
 set community no-export
!
route-map PEER-FILTER permit 30
 match ip address prefix-list GROUP-THREE
 set as-path prepend 100 100
!
```



Orden de procesamiento de la política BGP

- Para políticas aplicadas a un vecino BGP específico, la secuencia siguiente se aplica:
  - Para actualizaciones entrantes, el orden es:
    1. Route-map
    2. Filter-list
    3. Prefix-list
  - Para actualizaciones salientes, el orden es:
    1. Prefix-list
    2. Filter-list
    3. Route-map

Administración de Cambios en la Política

- Nuevas políticas sólo se aplican a las actualizaciones que pasan por el enrutador DESPUES de que la política haya sido introducida o cambiada
- Para facilitar los cambios de políticas en la tabla BGP completa, las sesiones de BGP tienen que ser "refrescadas"
  - Esto se logra "limpiando (clear)" la sesión BGP en sentido de entrada o salida, por ejemplo:

```
clear ip bgp <neighbour-addr> in|out
```

- NO SE OLVIDE de in o out — de lo contrario la sesión BGP se reiniciará por completo (hard reset)

Administración de Cambios en la Política

- Habilidad para "limpiar" las sesiones BGP para grupos de vecinos configurados con ciertos parámetros comunes criteria.

- **clear ip bgp <addr> [in|out]**

```
<addr>                  puede ser uno de los siguientes
x.x.x.x                 IP del vecino
*                       Todos los vecinos
ASN                     Todos los vecinos en un AS
external                Todos los vecinos externos
peer-group <name>       Vecinos en un peer-group
```

BGP - Control de Politicas

Talleres ISP