

# Atributos BGP y control de políticas



# Agenda

---

- Atributos BGP
- Selección de trayectoria BGP
- Aplicación de políticas

# Atributos BGP



Las *herramientas* disponibles  
para el trabajo

# ¿Qué es un atributo?

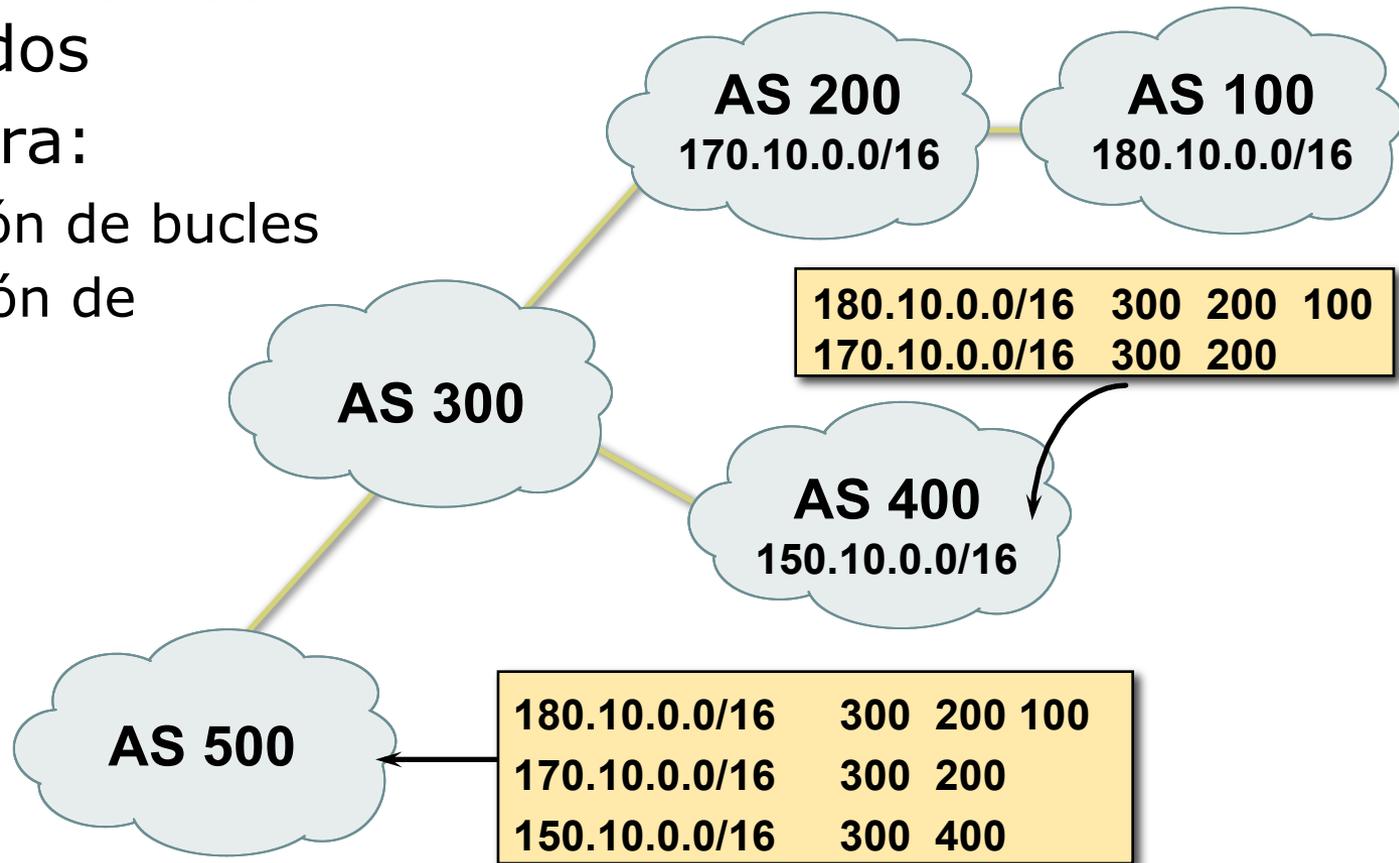
---



- ❑ Describe las características del prefijo
- ❑ Transitivos y no transitivos
- ❑ Algunos son mandatorios

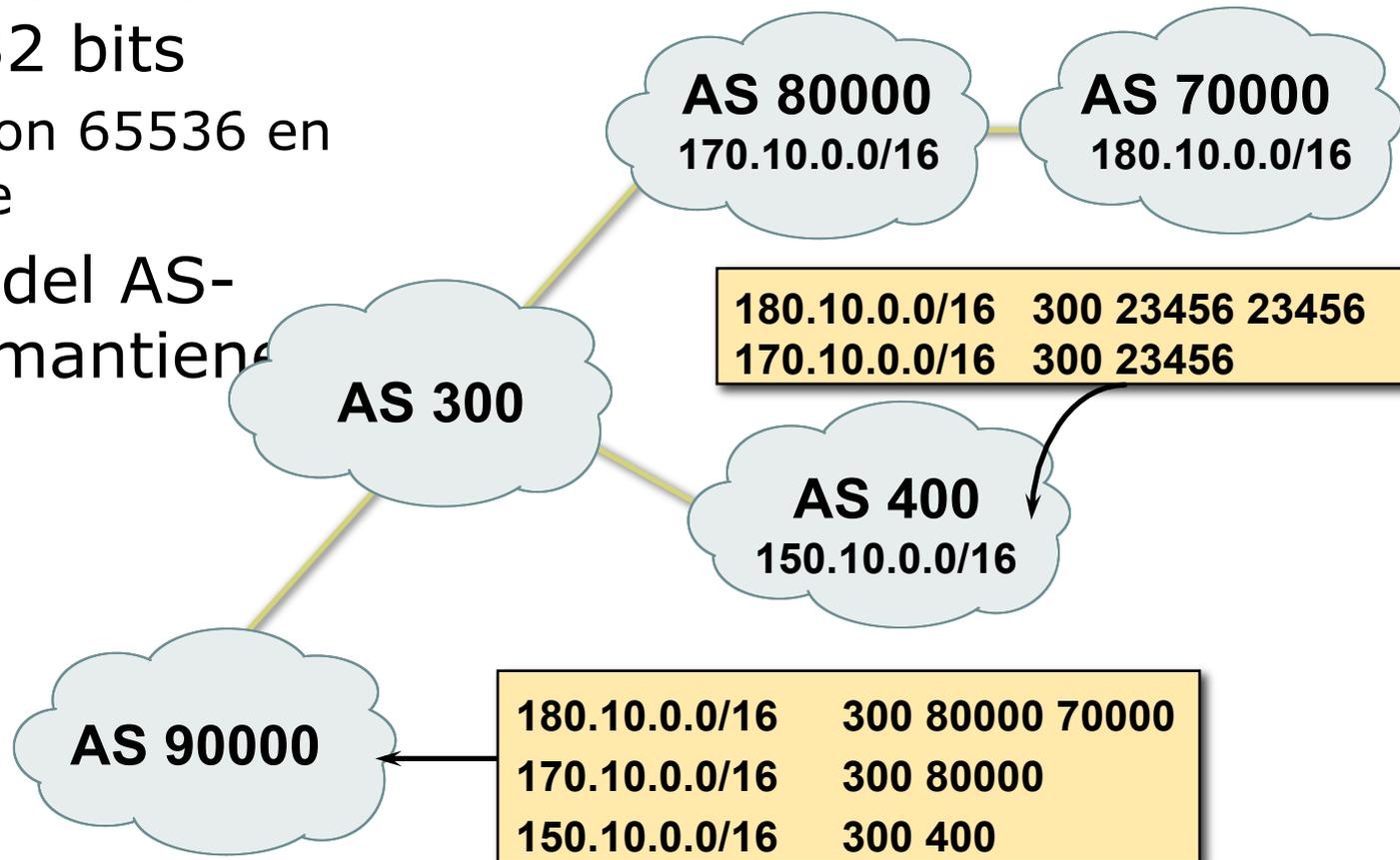
# AS-Path

- ❑ Secuencia de ASs atravesados
- ❑ Usado para:
  - Detección de bucles
  - Aplicación de políticas

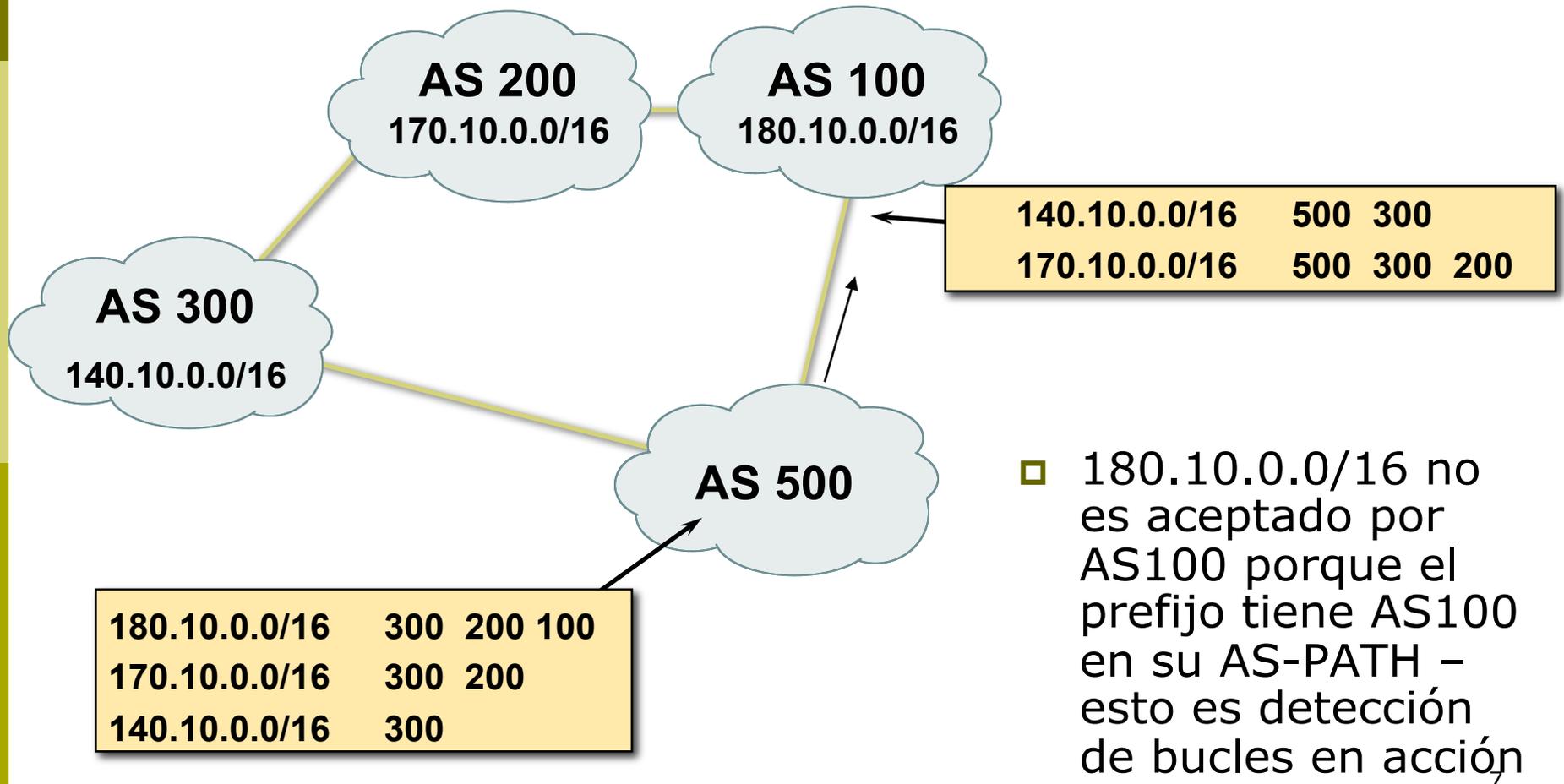


# AS-Path (con ASNs de 16 y 32 bits)

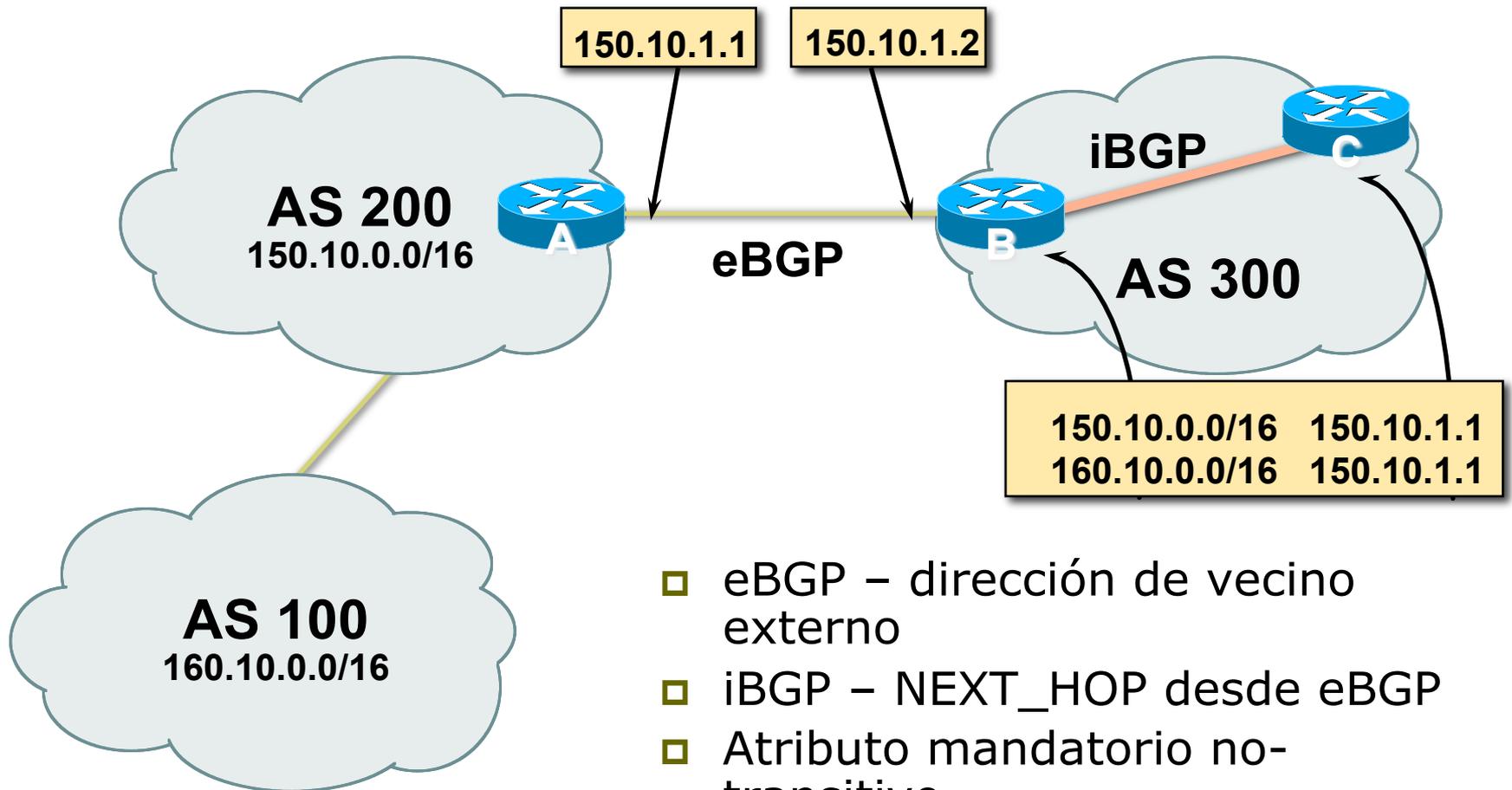
- Internet con ASNs de 16 y 32 bits
  - 32-bit son 65536 en adelante
- Longitud del AS-PATH se mantiene igual



# AS-Path – Detección de bucles

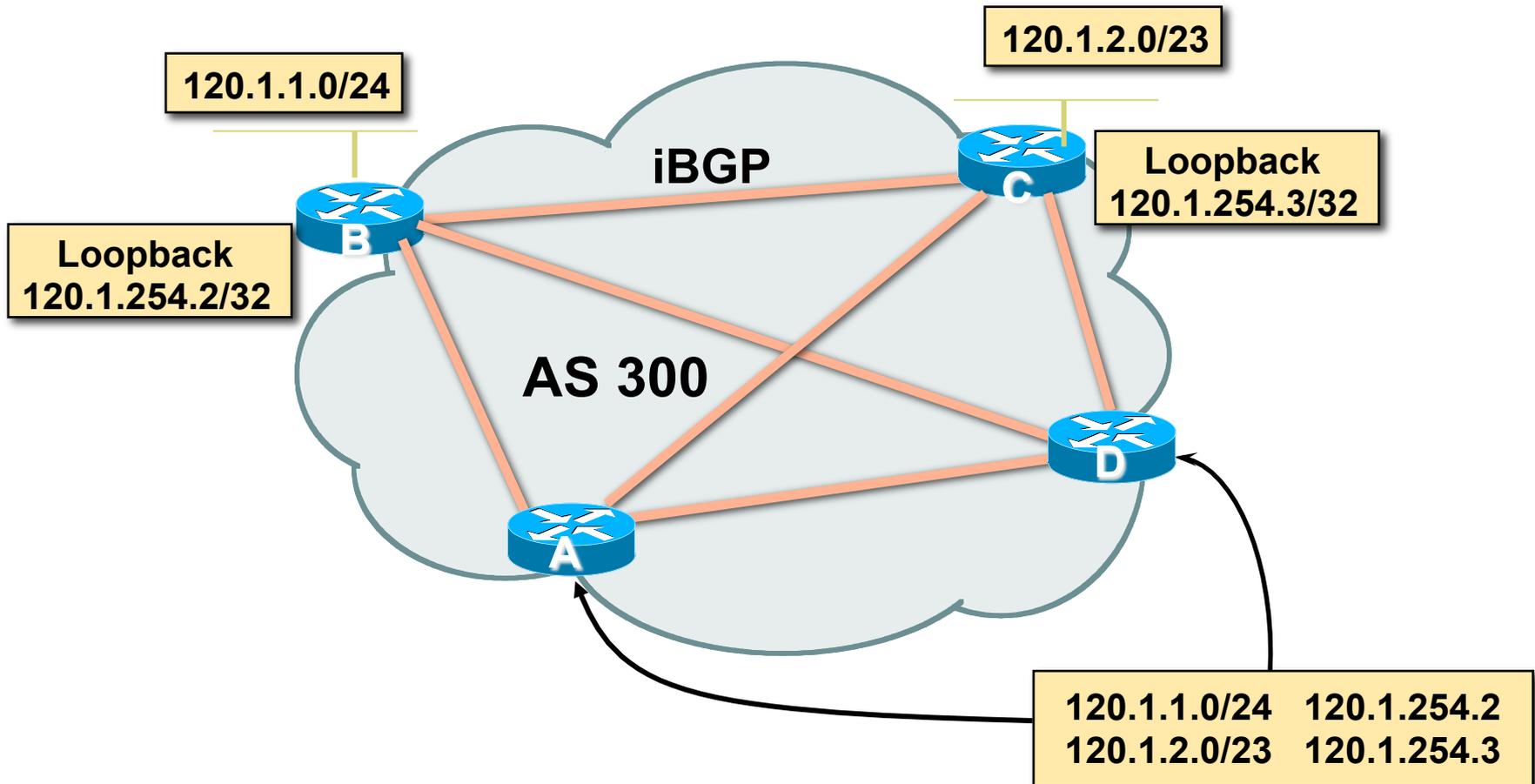


# Próximo salto – Next Hop



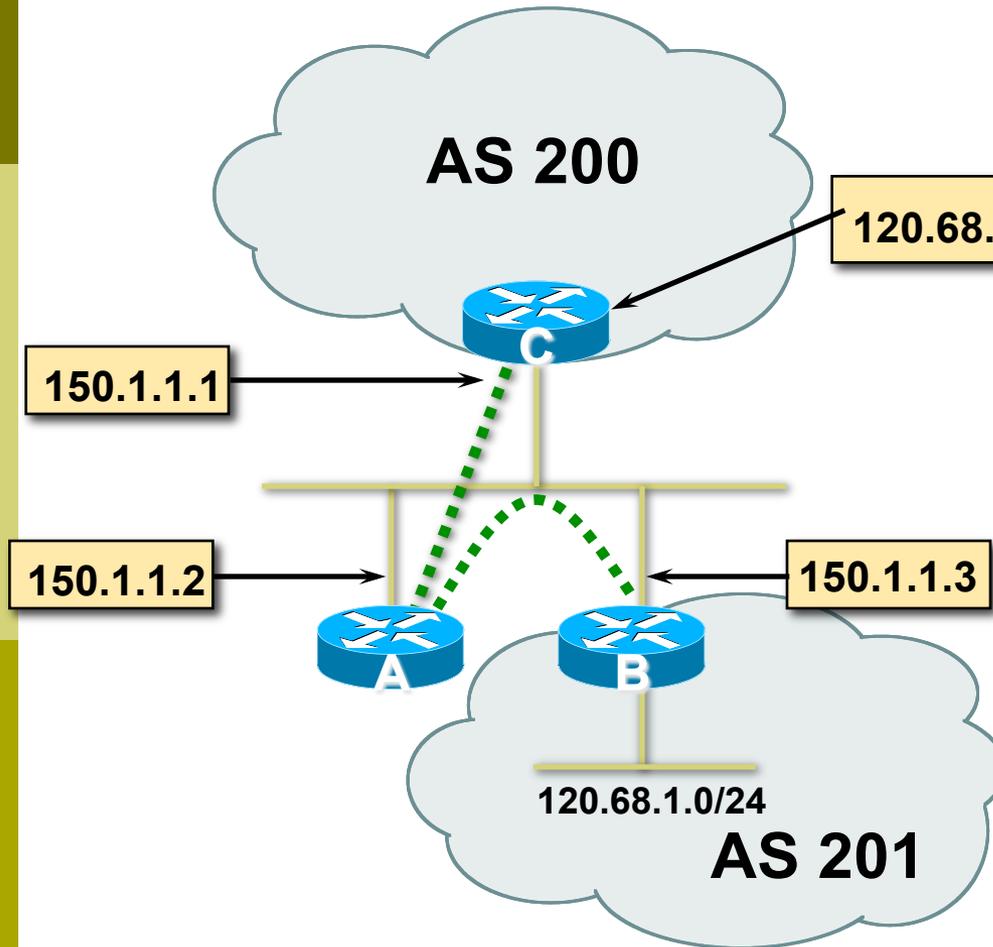
- eBGP – dirección de vecino externo
- iBGP – NEXT\_HOP desde eBGP
- Atributo mandatorio no-transitivo

# Próximo Salto en iBGP



- ❑ El próximos alto es la dirección loopback del enrutador
- ❑ Se hace una búsqueda recursiva en la tabla

# Próximo Salto de Tercero



- eBGP entre Router A y Router C
- eBGP entre Router A y Router B
- El prefijo 120.68.1/24 tiene la dirección de próximos alto 150.1.1.3 – esto se pasa a Router C en vez de 150.1.1.2
- Más eficiente
- No hace falta configuración extra

# Práctica Recomendada para Next Hop

---

- ❑ El comportamiento por defecto de Cisco IOS es que el next-hop externo se propague sin cambios a través de los enrutadores iBGP
  - Esto implica que el IGP debe incluir los next-hops externos
  - Si esto se olvida, las redes externas son invisibles
  - Cuando hay muchos vecinos eBGP, esto pone una carga innecesaria en el IGP
- ❑ La práctica recomendada en los ISPs es cambiar el next-hop externo por el enrutador local  
`neighbor x.x.x.x next-hop-self`

# Next Hop (Resumen)

---

- ❑ El IGP debe transportar la ruta a los next-hops
- ❑ Búsqueda recursiva en la tabla de rutas
- ❑ Desliga a BGP de la topología física
- ❑ Use “next-hop-self” para next hops externos
- ❑ Permite al IGP hacer decisiones inteligentes de reenvío

# Origen

---

- Transmite el origen del prefijo
- Es un atributo histórico
  - Usado en la transición de EGP a BGP
- Atributo transitivo y mandatorio
- Influye en la selección de la mejor ruta
- Tres valores: IGP, EGP, incomplete
  - IGP – generado por el comando network
  - EGP – generado por EGP
  - incomplete – redistribuido desde otro protocolo de enrutamiento

# Agregador

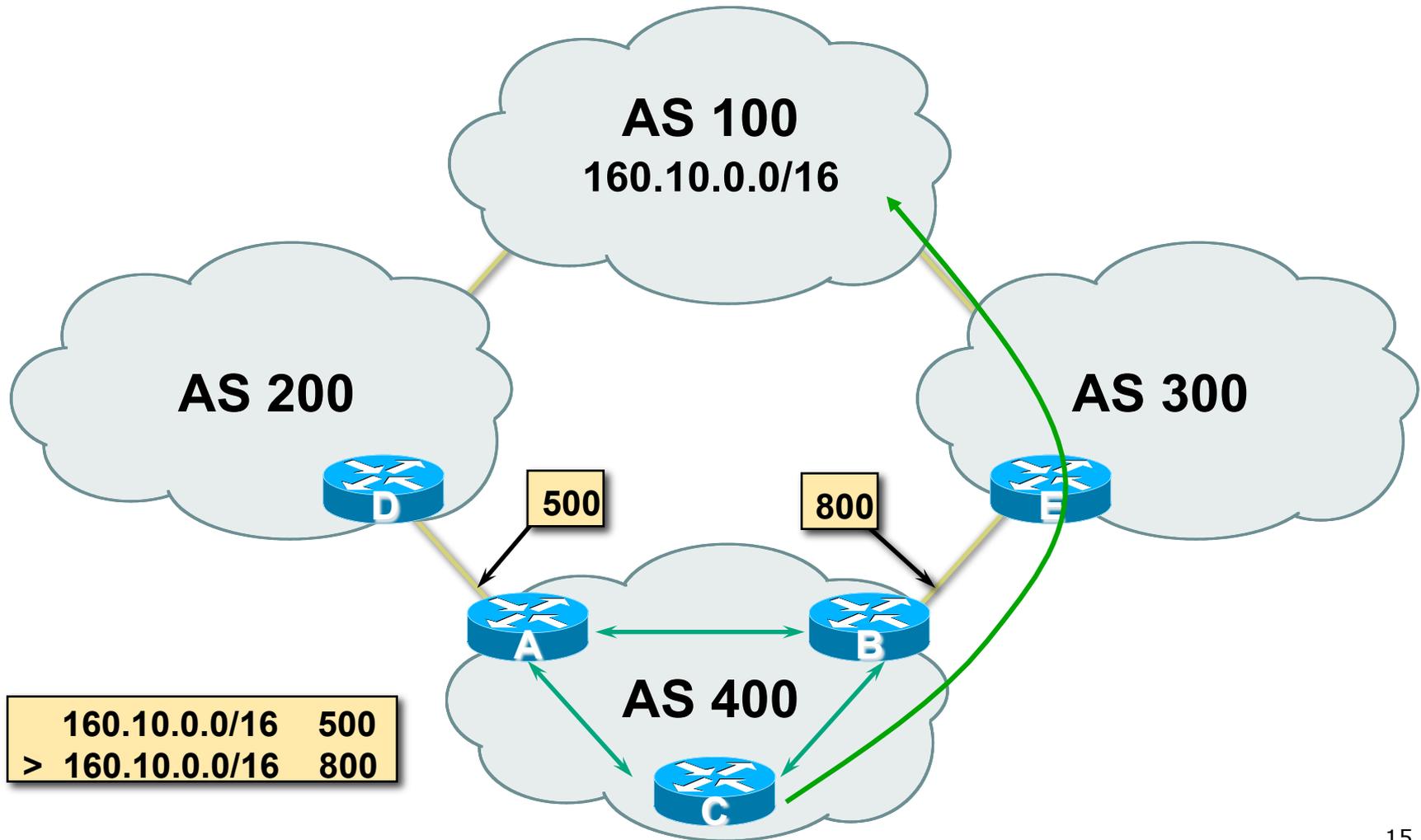
---

- ❑ Transmite la dirección IP del enrutador o interlocutor BGP que genera la ruta agregada
- ❑ Atributo opcional y transitivo
- ❑ Util para solución de problemas
- ❑ No influye la selección de la ruta
- ❑ Al usar "aggregate-address" se genera el atributo:

```
router bgp 100
```

```
aggregate-address 100.1.0.0 255.255.0.0
```

# Preferencia Local



# Preferencia Local

---

- ❑ Atributo opcional y no-transitivo
- ❑ Local al AS solamente
  - Valor por defecto es 100 (IOS)
- ❑ Usado en BGP para influir en la selección de la ruta
  - Determina el mejor camino para el tráfico *saliente*
- ❑ El camino con la preferencia local mayor gana

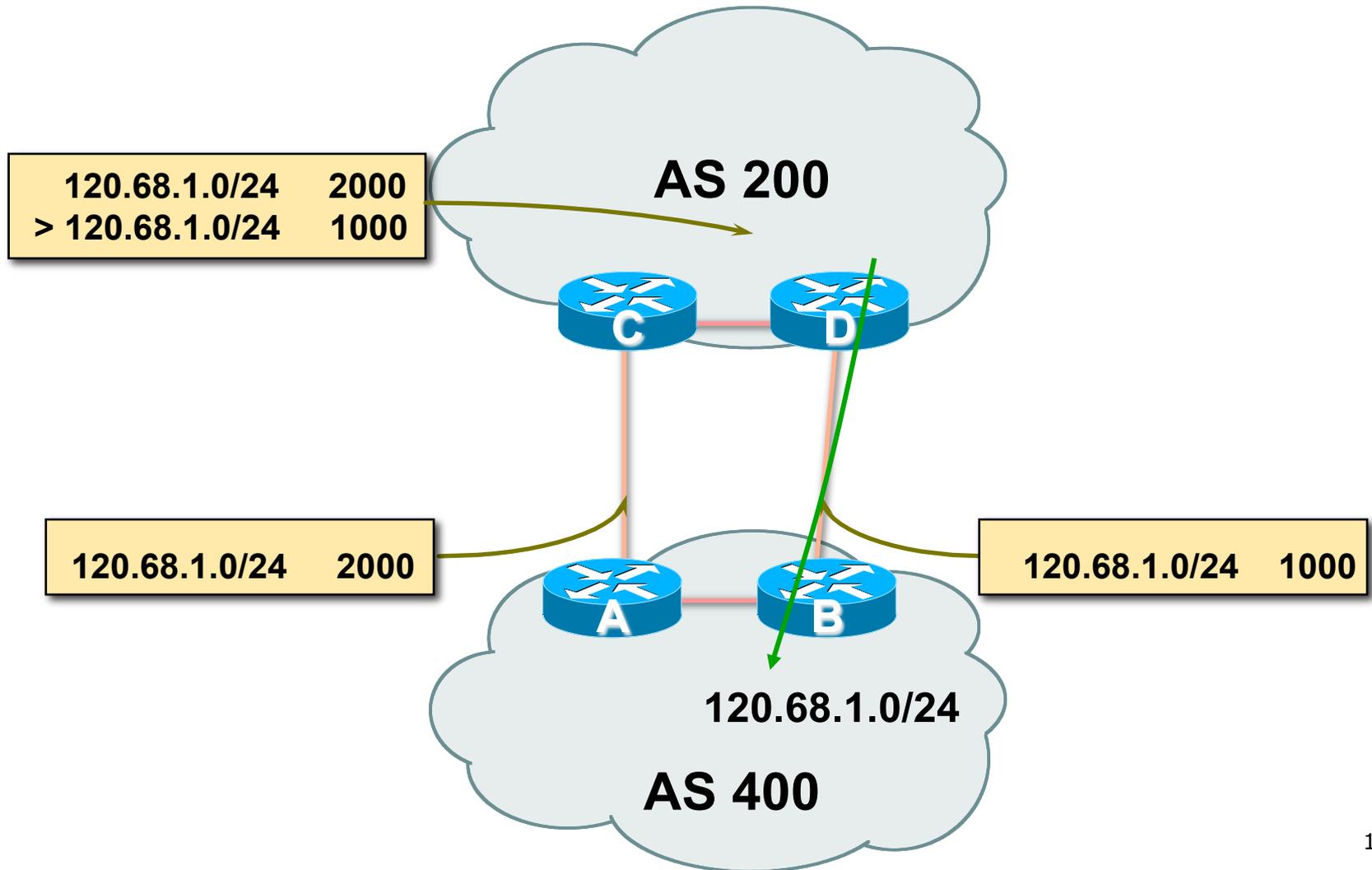
# Preferencia Local

---

## □ Configuración de Router B:

```
router bgp 400
  neighbor 120.5.1.1 remote-as 300
  neighbor 120.5.1.1 route-map local-pref in
!
route-map local-pref permit 10
  match ip address prefix-list MATCH
  set local-preference 800
route-map local-pref permit 20
!
ip prefix-list MATCH permit 160.10.0.0/16
```

# Multi-Exit Discriminator (MED)



# Multi-Exit Discriminator

---

- ❑ Inter-AS – no-transitivo y opcional
- ❑ Usado para transmitir la preferencia relativa de los puntos de entrada
  - Determina el mejor camino para el trafico entrante
- ❑ Comparable si las rutas son del mismo AS
  - `bgp always-compare-med` permite comparar los MEDs de diferentes ASes
- ❑ El camino con la MED menor gana
- ❑ La ausencia del atributo MED implica que el valor es cero (RFC4271)

# MED y Métrica IGP

---

- La métrica del IGP se puede transmitir por medio de la MED
  - **set metric-type internal** en route-map
    - Instruye a BGP que anuncie la MED que corresponda al valor de la métrica del IGP
    - Se monitorean los cambios (y se re-anuncia si es necesario) cada 600 seg.
    - **bgp dynamic-med-interval <secs>**

# Multi-Exit Discriminator

---

## ❑ Configuración del Router B:

```
router bgp 400
  neighbor 120.5.1.1 remote-as 200
  neighbor 120.5.1.1 route-map set-med out
!
route-map set-med permit 10
  match ip address prefix-list MATCH
  set metric 1000
route-map set-med permit 20
!
ip prefix-list MATCH permit 120.68.1.0/24
```

# Weight - Peso

---

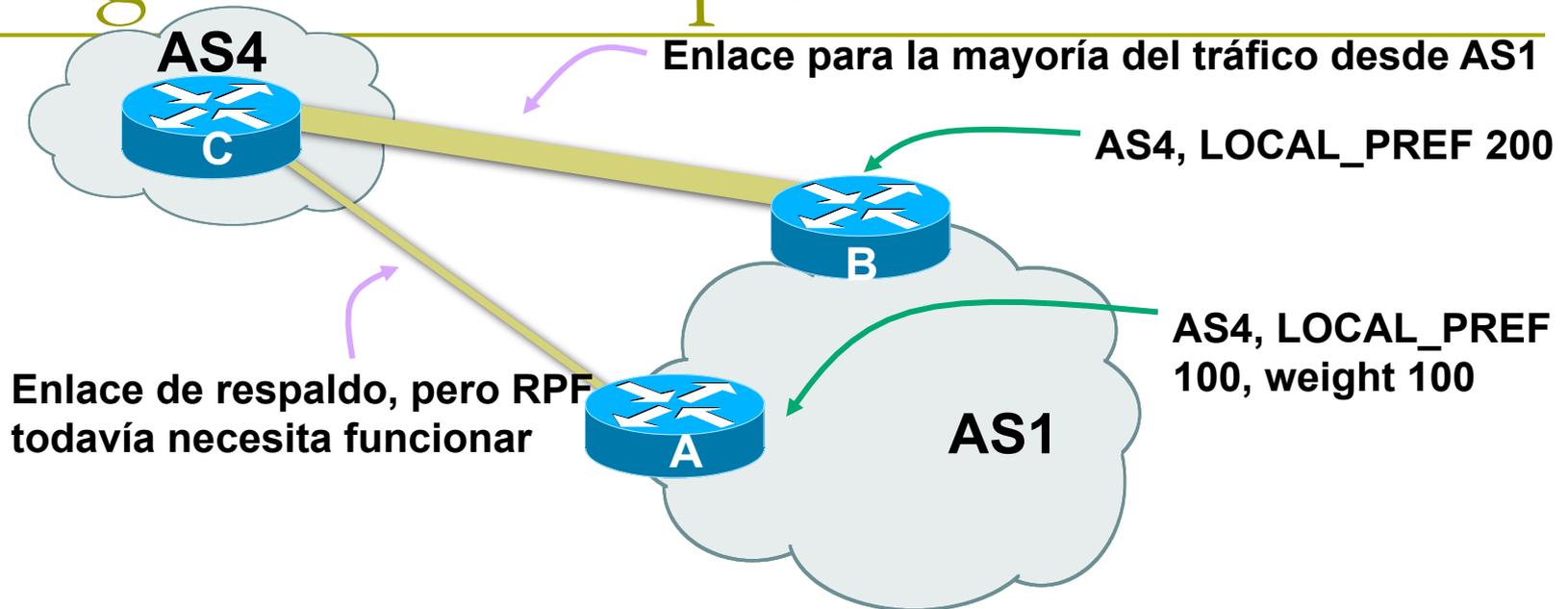
- ❑ No es realmente un atributo – local al enrutador
- ❑ El mayor peso gana
- ❑ Aplicado a todas las rutas de un vecino

```
neighbor 120.5.7.1 weight 100
```

- ❑ Asignar a ciertas rutas basado en filtro

```
neighbor 120.5.7.3 filter-list 3 weight 50
```

# Weight – Para implementar RPF



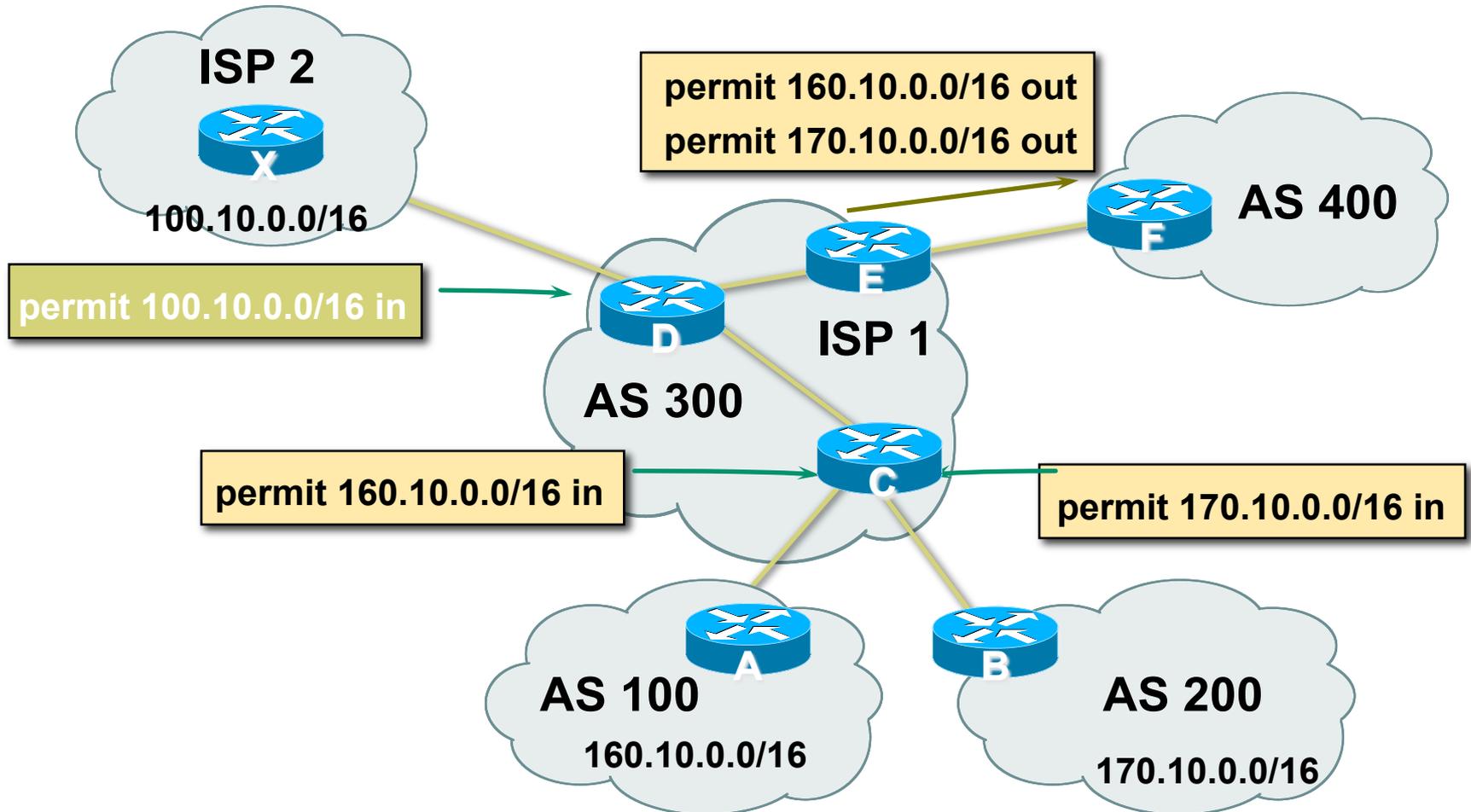
- El mejor camino a AS4 desde AS1 es siempre via B debido al local-pref
- Pero los paquetes que llegan a A desde AS4 a través del enlace directo de C a A pasarán el chequeo RPF porque ese camino tiene una prioridad gracias al *weight*
  - Si no se asignara el *weight*, el mejor camino de vuelta a AS4 sería por B, y el chequeo RPF no pasaría

# Comunidad

---

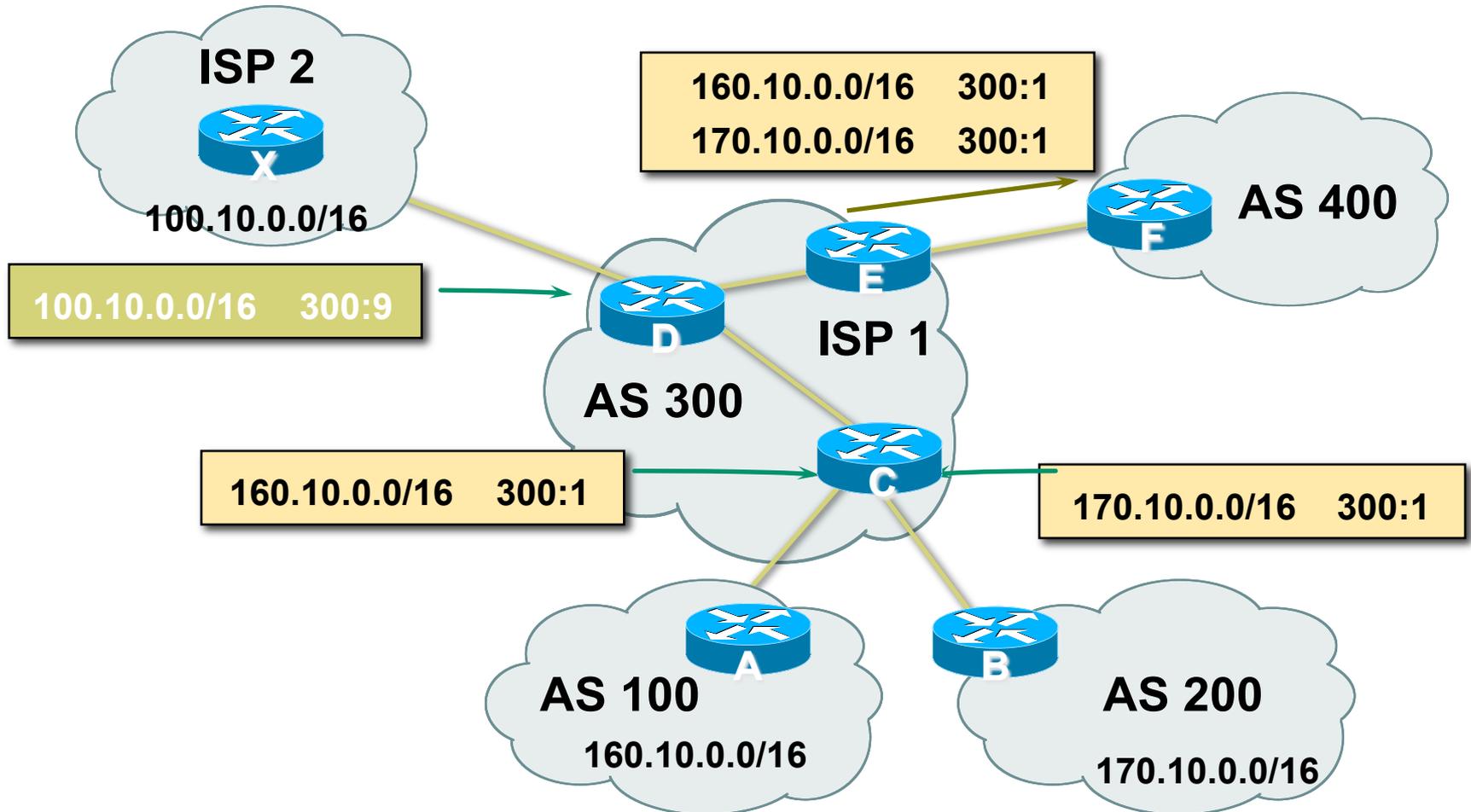
- ❑ Decritas en RFC1997
  - Atributo transitivo y opcional
- ❑ Entero de 32 bits
  - Representado como dos enteros de 16 bits (RFC1998)
  - El formato común es <local-ASN>:xx
  - 0:0 to 0:65535 and 65535:0 to 65535:65535 están reservados
- ❑ Usado para agrupar destinos
  - Cada destino podría ser miembro de diferentes comunidades
- ❑ Muy útil para aplicar políticas intra e inter-AS

# Ejemplo de Comunidad (antes)



# Ejemplo de Comunidad

## (después)

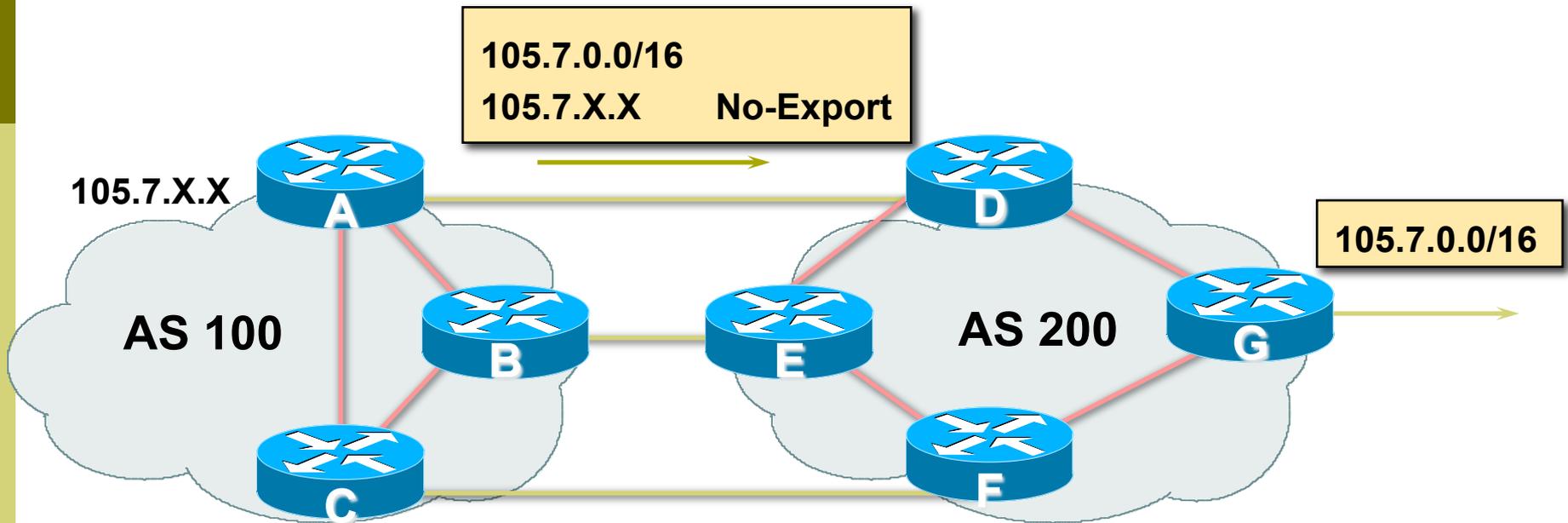


# Comunidades reconocidas

---

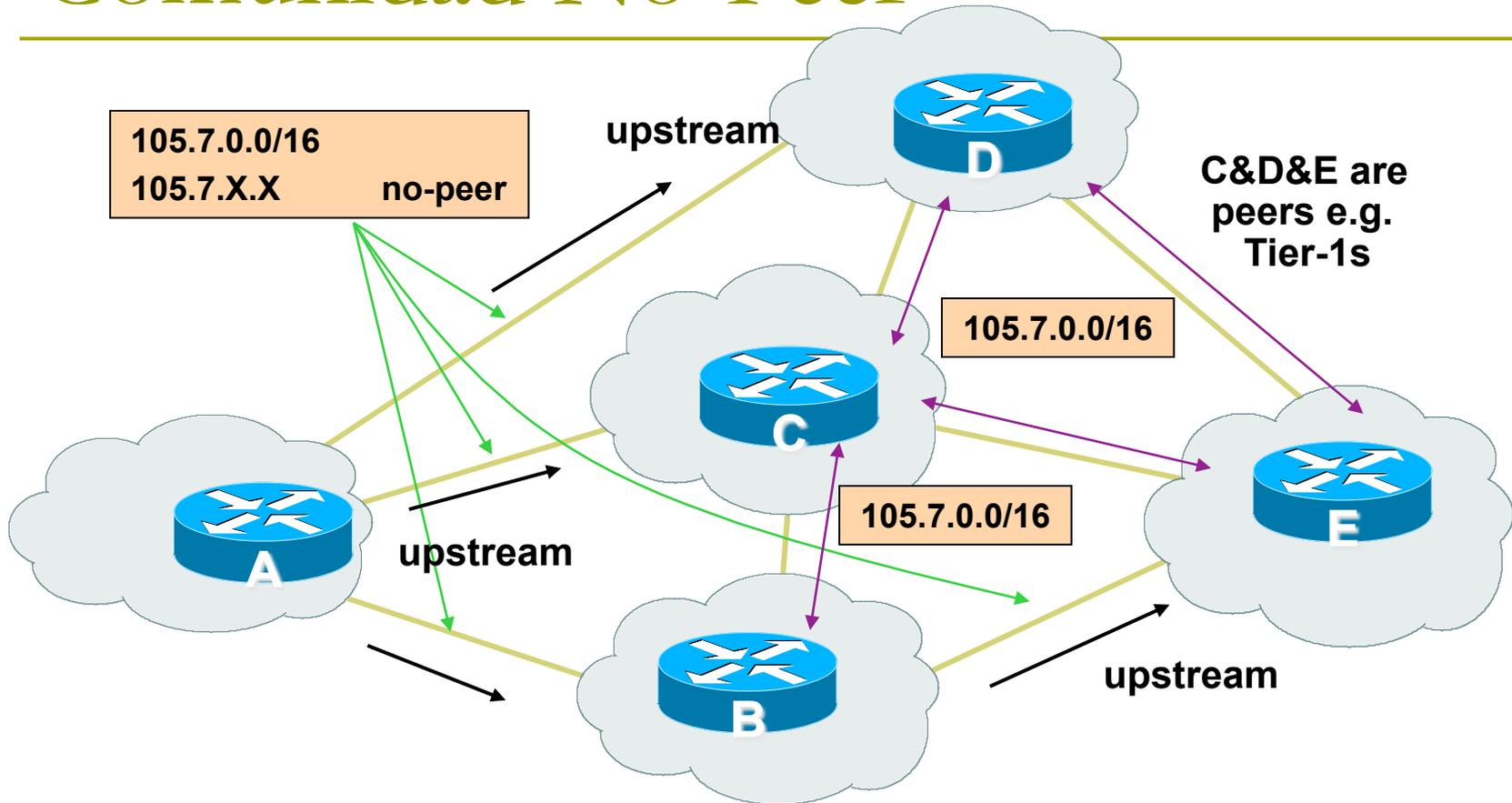
- Varias comunidades han sido definidas para ser reconocidas
  - [www.iana.org/assignments/bgp-well-known-communities](http://www.iana.org/assignments/bgp-well-known-communities)
- no-export 65535:65281
  - No anunciar a ningún vecino eBGP
- no-advertise 65535:65282
  - No anunciar a ningún vecino
- no-export-subconfed 65535:65283
  - No anunciar fuera del AS local (only used with confederations)
- no-peer 65535:65284
  - No anunciar a vecinos bi-laterales (RFC3765)

# Comunidad No-Export



- ❑ AS100 anuncia agregado y sub-prefijos
  - La intención es el balanceo de cargas
- ❑ Los sub-prefijos se marcan con la comunidad **no-export**
- ❑ El Router G en AS200 no anuncia los prefijos marcados con la comunidad **no-export**

# Comunidad No-Peer



- Los sub-prefijos marcados con **no-peer** no se envían a los vecinos bi-laterales
  - Sólo se envían a los proveedores de tránsito

# ¿Y los ASNs de 4 bytes?

---

- ❑ Las comunidades son usadas ampliamente para codificar políticas de enrutamiento de los ISPs
  - Atributo de 32 bits
- ❑ El formato RFC1998 es ahora la práctica común
  - ***ASN:number***
- ❑ Funciona bien con los ASNs de 2 bytes, pero los de 4 bytes no se pueden codificar
- ❑ Soluciones:
  - Use "ASN privado" para los primeros 16 bits
  - Espere a que se implemente <http://datatracker.ietf.org/doc/draft-ietf-idr-as4octet-extcomm-generic-subtype/>

# Resumen

## Atributos en Acción

---

```
Router6>sh ip bgp
```

```
BGP table version is 30, local router ID is 10.0.15.246
```

```
Status codes: s suppressed, d damped, h history, * valid, >  
best,
```

```
          i - internal, r RIB-failure, S Stale
```

```
Origin codes: i - IGP, e - EGP, ? - incomplete
```

Network Path	Next Hop	Metric	LocPrf	Weight	
*>i10.0.0.0/26	10.0.15.241	0	100	0	i
*>i10.0.0.64/26	10.0.15.242	0	100	0	i
*>i10.0.0.128/26	10.0.15.243	0	100	0	i
*>i10.0.0.192/26	10.0.15.244	0	100	0	i
*>i10.0.1.0/26	10.0.15.245	0	100	0	i
*> 10.0.1.64/26	0.0.0.0	0		32768	i
...					

# Algoritmo de Selección de Ruta



¿Por qué es ésta la mejor ruta?

# El algoritmo de selección de rutas de BGP en IOS: Primera Parte

---

- ❑ No considere la ruta si no hay una ruta al próximo salto
- ❑ No considere la ruta iBGP si no hay sincronización (Cisco IOS)
- ❑ El peso (weight) mayor (local al router)
- ❑ La preferencia local mayor (globalmente dentro del AS)
- ❑ Preferir ruta originada localmente
- ❑ El AS-Path más corto

# El algoritmo de selección de rutas de BGP en IOS: Segunda Parte

---

- Código de origen menor
  - IGP < EGP < incomplete
- Multi-Exit Discriminator (MED) menor
  - Si se usa `bgp deterministic-med`, ordenar las rutas antes de compararlas
  - Si se usa `bgp always-compare-med`, comparar todas las rutas
  - De lo contrario, las MEDs sólo se comparan si provienen del mismo AS (por defecto)

# El algoritmo de selección de rutas de BGP en IOS: Tercera Parte

---

- Preferir ruta eBGP sobre ruta iBGP
- La ruta con la menor métrica IGP al próximo salto
- Para las rutas eBGP:
  - Si se habilita *multipath*, instalar N rutas paralelas en la tabla
  - Si el router-id es el mismo, ir al próximo paso
  - Si el router-id no es el mismo, seleccionar la ruta más vieja

# El algoritmo de selección de rutas de BGP en IOS: Cuarta Parte

---

- ❑ El menor router-id (originator-id en el caso de las rutas "reflejadas")
- ❑ Cluster-list más corto
  - El cliente debe tomar en cuenta los atributos del Route Reflector!
- ❑ La dirección del vecino menor

# Aplicando Políticas con BGP



Cómo usar las *herramientas*

# Aplicando Políticas con BGP

---

- ❑ Política – Puede ser 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

---

- Filtro de prefijos por vecino
  - Configuración incremental
- Aplicado a la entrada o la salida
- Basado en números de red (utilizando el formato familiar de dirección IP/Máscara)
- Utilizar access-lists para filtrar prefijos se hizo obsoleto hace tiempo
  - **No recomendado!**

# Prefix-list - Sintaxis del Comando

---

## □ Sintaxis:

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

- `network/len`: El prefijo y su longitud

- `ge ge-value`: “mayor o igual que”

- `le le-value`: “menor o igual que”

## □ Ambos “ge” y “le” son opcionales

- Para especificar el rango de la longitud de prefijo a coincidir para los prefijos más específicos que red/longitud

## □ El número de secuencia también es opcional

- `no ip 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 permitir prefijos entre /12 y /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.

- ❑ Permitir 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

# Policy Control – Prefix List

---

## □ Example Configuration

```
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 – Lista de Prefijos

---

- Filtrar rutas basado en AS-Path
  - Entrada o salida
- 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 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 – Advertencias

---

- ❑ Una línea puede tener múltiples comandos “set”
- ❑ Una línea puede tener múltiples “match”
- ❑ Línea con solamente “match”
  - Sólo pasan los prefijos que coinciden, y los demás son descartados
- ❑ Línea con solamente “set”
  - Todos los prefijos coinciden y se les asigna algo
  - Las líneas siguientes se ignoran
- ❑ Línea con match/set y sin más líneas
  - Sólo se asigna algo a los prefijos que coinciden, y el resto se descarta

# 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 ! Don't forget this
```

# Route Maps – Coincidir prefijos

---

## □ Configuración de Ejemplo

```
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 – Filtrado de AS-PATH

---

## □ Configuración de Ejemplo

```
router bgp 100
  neighbor 102.10.1.2 remote-as 200
  neighbor 102.10.1.2 route-map filter-on-as-path in
!
route-map filter-on-as-path permit 10
  match as-path 1
  set local-preference 80
!
route-map filter-on-as-path 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 prepends

---

- Configuración para AS-PATH prepend

```
router bgp 300
  network 105.7.0.0 mask 255.255.0.0
  neighbor 2.2.2.2 remote-as 100
  neighbor 2.2.2.2 route-map SETPATH out
!
route-map SETPATH permit 10
  set as-path prepend 300 300
```

- Utilice su propio número de AS cuando haga “prepending”
  - De lo contrario, la detección de bucles de BGP puede causar desconexiones

# Route Maps – Coincidir Comunidades

---

## ❑ Configuración de Ejemplo

```
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 – Asignar Comunidades

---

## ▣ Configuración de Ejemplo

```
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
```

!

# Cambios de Políticas

---

- ❑ Sólo se aplican nuevas políticas 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)

# Cambios de Políticas

---

- Se pueden “limpiar” las sesiones BGP para grupos de vecinos configurados con ciertos parámetros comunes

□ **clear ip bgp <addr> [in|out]**

<addr> puede ser uno de los siguientes

<b>x.x.x.x</b>	IP del vecino
<b>*</b>	Todos los vecinos
<b>ASN</b>	Todos los vecinos en un AS
<b>external</b>	Todos los vecinos externos
<b>peer-group &lt;name&gt;</b>	Vecinos en un peer-group

# Atributos de BGP y Control de Políticas

