

Ruteo externo: BGP

Redes y Servicios Avanzados en Internet

Ruteo Externo
BGP – Políticas y configuraciones

Alejandro Sabolansky
Nicolás Macia

Aplicando políticas en BGP

- El ruteo entre AS se controla a través de una política que está escrita, que debe ser traducida en el protocolo de ruteo.
- BGP permite controlar qué y cómo se aprenden y qué y cómo se publican los distintos prefijos de red que se procesan en un router.
- Las políticas se implementan mediante filtros:
 - ¿Qué prefijos quiero o no aprender?
 - ¿Qué preferencias se aplican a determinados prefijos?
 - ¿Qué características de los prefijos quiero evaluar? AS-PATH, origin, netblock, etc.
- Las herramientas que permiten crear políticas son:
 - prefix-list
 - filter-list
 - route-maps

Prefix list

- Se aplican a la entrada o a la salida de lo que se intercambia.
- Se basa en prefijos de red que se anuncian o publican:
 - 163.10.50.0/24
 - 10.0.0.0/8
- Se componen de un conjunto de reglas que:
 - Se evalúan de arriba hacia abajo. Si hay matching se hace lo que la regla indica y se termina la evaluación. Si no hay matching se evalúa la siguiente regla.
 - al final tienen un “denegar todo” implícito.
- No es recomendado su uso.

Ejemplo prefix-list

```
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
```

Expresiones regulares

- Coincide con un elemento posible
- * Cero, uno o muchos elementos del tipo del que precede al *
- .* Coincide con cualquier cosa
- + Uno o muchos elementos del tipo del que precede al +
- .+ Coincide con cualquier cosa que al menos tenga un elemnto

Expresiones regulares en BGP

<code>^</code>	Comienzo
<code>\$</code>	Fin
<code>^\$</code>	Elemento vacío. En BGP las rutas locales del AS, cuando se publican a otros AS, el ASPATH está vacío
<code>_1500\$</code>	Rutas que fueron originadas en el AS 1500
<code>^1500_500_</code>	Rutas que se recibieron del AS 1500 y que éste las recibió del AS 500
<code>_1500_</code>	Rutas que pasaron por el AS 1500

Filter list

- Se aplican a la entrada o a la salida de lo que se intercambia.
- Se basa en el atributo AS-PATH de los prefijos de red que se anuncian o publican:
- Se componen de un conjunto de reglas que:
 - Se evalúan de arriba hacia abajo. Si hay matching se hace lo que la regla indica y se termina la evaluación. Si no hay matching se evalúa la siguiente regla.
 - al final tienen un “denegar todo” implícito.
- No es recomendado su uso.

Ejemplo filter list

```
router bgp 100
  neighbor 220.200.1.1 remote-as 210
  neighbor 220.200.1.1 filter-list 5 out
  neighbor 220.200.1.1 filter-list 6 in
!
ip as-path access-list 5 permit ^200$
ip as-path access-list 6 permit ^150$
```


Route maps

- Es una herramienta que se puede utilizar para implementar políticas de ruteo, entre otras cosas.
- Se componen de una secuencia numerada de políticas.
- La secuencia del route-map se ejecuta en orden ascendente.
 - Si el tráfico evaluado aplica entonces se hace lo que la secuencia utilizada indica (permit o deny).
 - Si el tráfico no aplica, se evalúa la siguiente secuencia del route-map.
 - Por defecto al final hay deny implícito.

```
router bgp 2222
  network 10.0.2.0/24
  neighbor 10.0.2.1 remote-as 33333
  neighbor 10.0.1.1 remote-as 11111
  neighbor 10.0.1.1 route-map ejemplo in

route-map ejemplo deny 10
...

route-map ejemplo deny 12
...

route-map ejemplo permit 20
...

route-map ejemplo permit 30
...
```

Route maps

Cada secuencia del route-map está compuesta de:

- Una acción a ejecutar, **permit** o **deny**, sobre el tráfico que matcheo en la secuencia evaluada.
- Pueden estar compuestos de cero, una o mas comandos “**match**” para indicar restricciones al tipo de tráfico que se debe considerar para aplicar la secuencia del route-map que se está evaluando.
- Pueden estar compuestos de cero, una o más comandos “**set**” para manipular los atributos de los elementos que aplicaron en la secuencia del route-map que se está evaluando.

Route maps (Cont.)

- Una línea con solo “match”
 - Pasan los prefijos que coinciden y los que no, serán evaluados en la siguiente secuencia del route-map
- Una línea con solo “set”:
 - Manipula atributos de los elementos que aplicaron en la secuencia del route-map evaluado.

```
router bgp 5692
  neighbor 10.0.1.2 remote-as 5692
  neighbor 10.0.0.1 remote-as 1000
  neighbor 10.0.0.1 route-map setear_preferencia_local in
  network 10.0.1.0/24

route-map setear_preferencia_local permit 10
  match ip address 5
  set local-preference 110

route-map setear_preferencia_local permit 20

access-list 5 permit 10.0.3.0 0.0.0.255
access-list 5 deny any
```

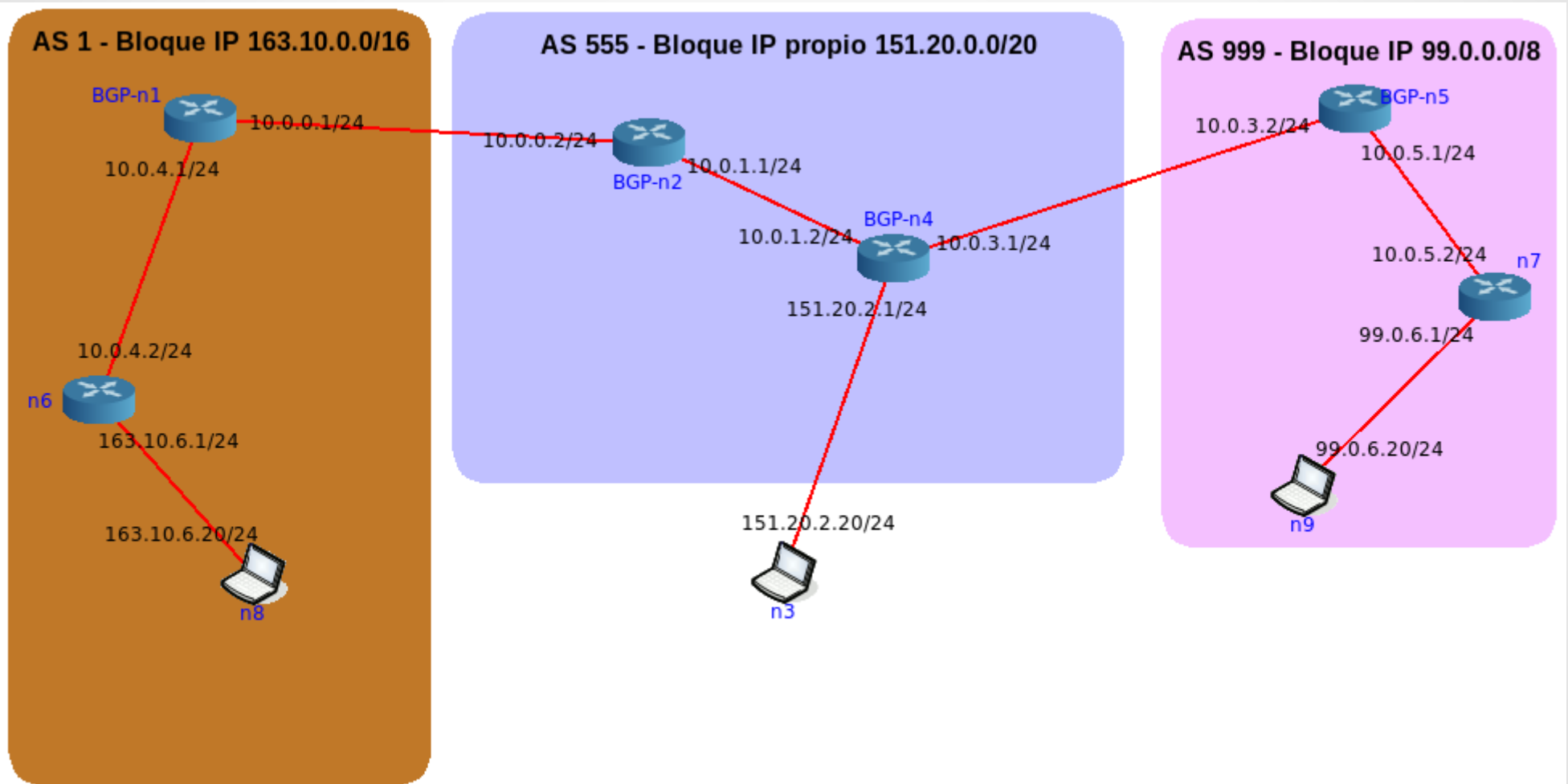
Filtrado por AS-PATH con route-maps

```
router bgp 100
  neighbor 220.200.1.2 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
  !
route-map filter-on-as-path permit 30
  !
ip as-path access-list 1 permit _150$
ip as-path access-list 2 permit _210_
```

Aplicación de políticas

- Cuando se modifica la política, la misma es implementada la próxima vez que el prefijo es anunciado o recibido por el router.
- Para que los cambios de políticas sean realizados de inmediato, las sesiones BGP deben ser “refrescadas”.
 - Para ello es posible utilizar el comando clear en sentido entrante o saliente:
 - `clear ip bgp <neighbor_adress> in | out`
 - Esto es conocido con soft reset
- Si se omite el in o el out, se realizará un hard reset, lo que no es recomendado.

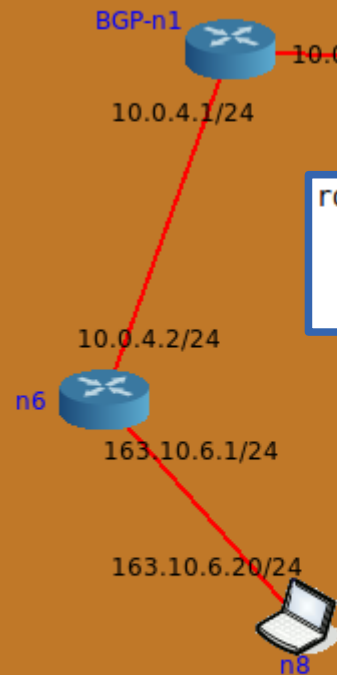
Peerings / Transit AS



Peerings / Transit AS

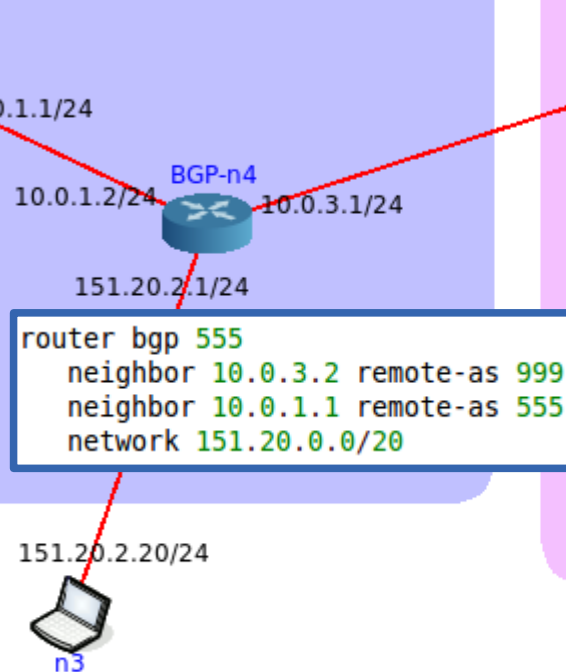
```
router bgp 1
  neighbor 10.0.0.2 remote-as 555
  network 163.10.0.0/16
```

AS 1 - Bloque IP 163.10.0.0/16



```
router bgp 555
  neighbor 10.0.0.1 remote-as 1
  neighbor 10.0.1.2 remote-as 555
  network 151.20.0.0/20
```

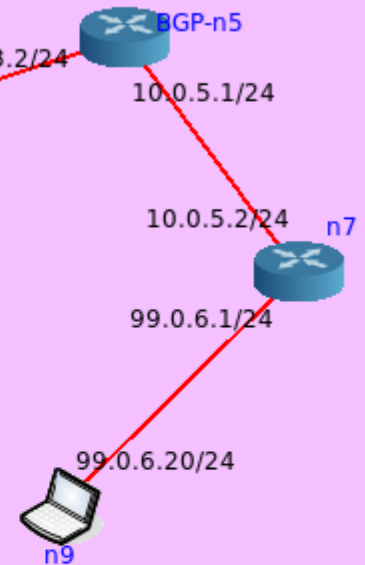
AS 555 - Bloque IP propio 151.20.0.0/20



```
router bgp 555
  neighbor 10.0.3.2 remote-as 999
  neighbor 10.0.1.1 remote-as 555
  network 151.20.0.0/20
```

```
router bgp 999
  neighbor 10.0.3.1 remote-as 555
  network 99.0.0.0/8
```

AS 999 - Bloque IP 99.0.0.0/8

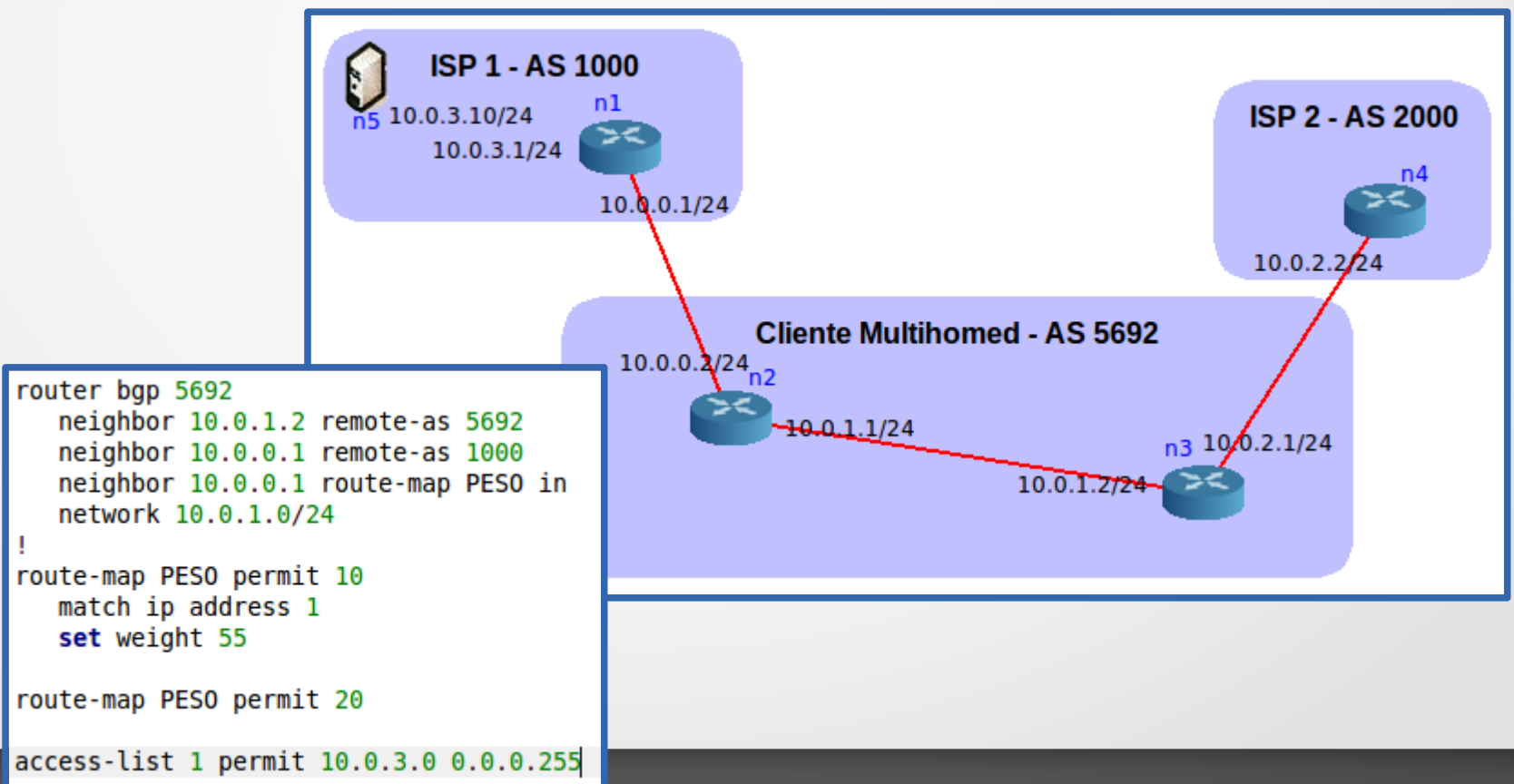


Atributos BGP

- Las rutas BGP tienen asociadas distintos atributos
- Algunos propagados con las rutas, otros no (transitivos)
- Se utilizan en el proceso de elección de la mejor ruta
 - Weight / Local Preference
 - Multi-exit discriminator / Origin
 - AS_PATH / Next hop
 - Community

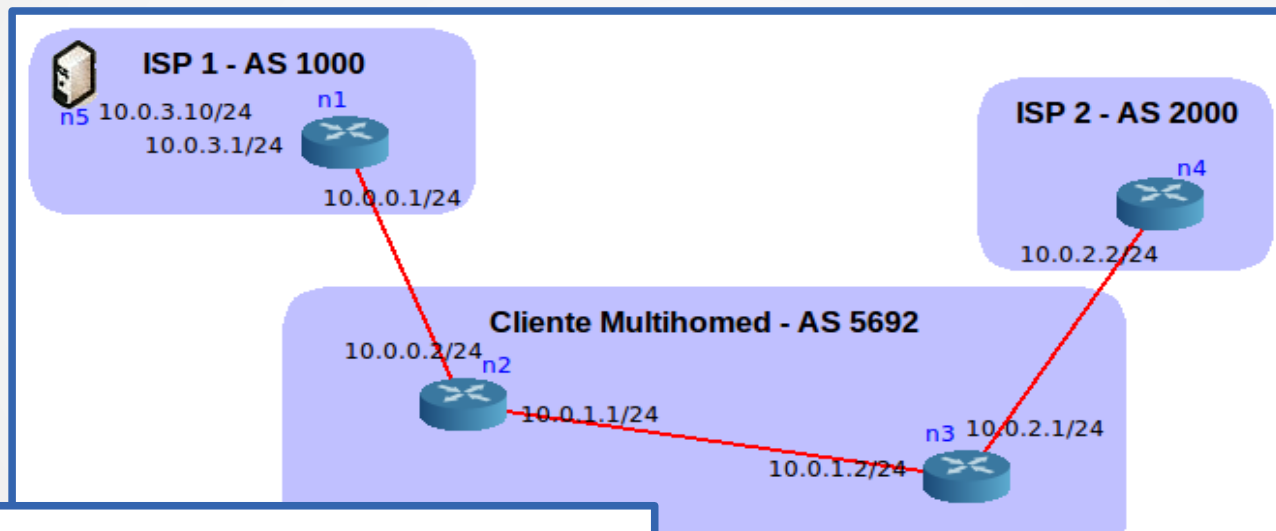
Weight

- Atributo LOCAL - No se propaga dentro ni fuera del AS
 - En “n2”, podemos indicar que queremos usar la ruta aprendida a la red 10.0.3.0/24 a través de “n1”



Local Preference

- Se propaga solamente dentro del AS
 - Si queremos que nuestro AS prefiera salir por el AS 2000, salvo la red propia del AS 1000



```
router bgp 5692
  neighbor 10.0.1.2 remote-as 5692
  neighbor 10.0.0.1 remote-as 1000
  neighbor 10.0.0.1 route-map setear_preferencia_local in
  network 10.0.1.0/24

route-map setear_preferencia_local permit 10
  match ip address 5
  set local-preference 110

route-map setear_preferencia_local permit 20

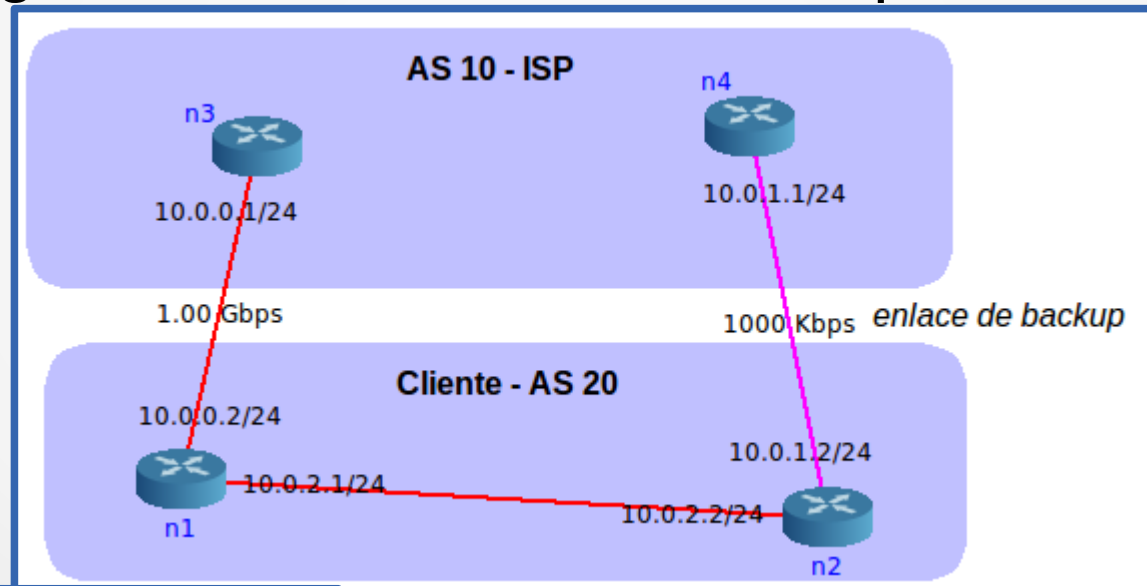
access-list 5 permit 10.0.3.0 0.0.0.255
access-list 5 deny any
```

```
router bgp 5692
  neighbor 10.0.1.1 remote-as 5692
  neighbor 10.0.2.2 remote-as 2000
  neighbor 10.0.2.2 route-map setear_preferencia_local in
  network 10.0.1.0/24

route-map setear_preferencia_local permit 10
  set local-preference 105
```

Multi-exit discriminator (MED)

- Sugerencia a un AS externo sobre el lugar por donde se prefiere que entre el tráfico a nuestro AS
- Se propaga dentro del AS vecino. Se prefiere el $< \text{MED}$



```
router bgp 20
  neighbor 10.0.2.2 remote-as 20
  neighbor 10.0.0.1 remote-as 10
  neighbor 10.0.0.1 route-map miMED out
  network 10.0.2.0/24

route-map miMED permit 10
  set metric 100
```

```
router bgp 20
  neighbor 10.0.2.1 remote-as 20
  neighbor 10.0.1.1 remote-as 10
  neighbor 10.0.1.1 route-map miMED-backup out
  network 10.0.2.0/24

route-map miMED-backup permit 10
  set metric 700
```

Origin

- Indica el origen de la ruta BGP

- IGP: La ruta es BGP

- `network 163.10.0.0/26`

- EGP: La ruta es del protocolo EGP (antecesor de BGP)

- Incomplete: La ruta es producto de una redistribución. El origen de la ruta es desconocido

- `router bgp 10`

- `redistribute static`

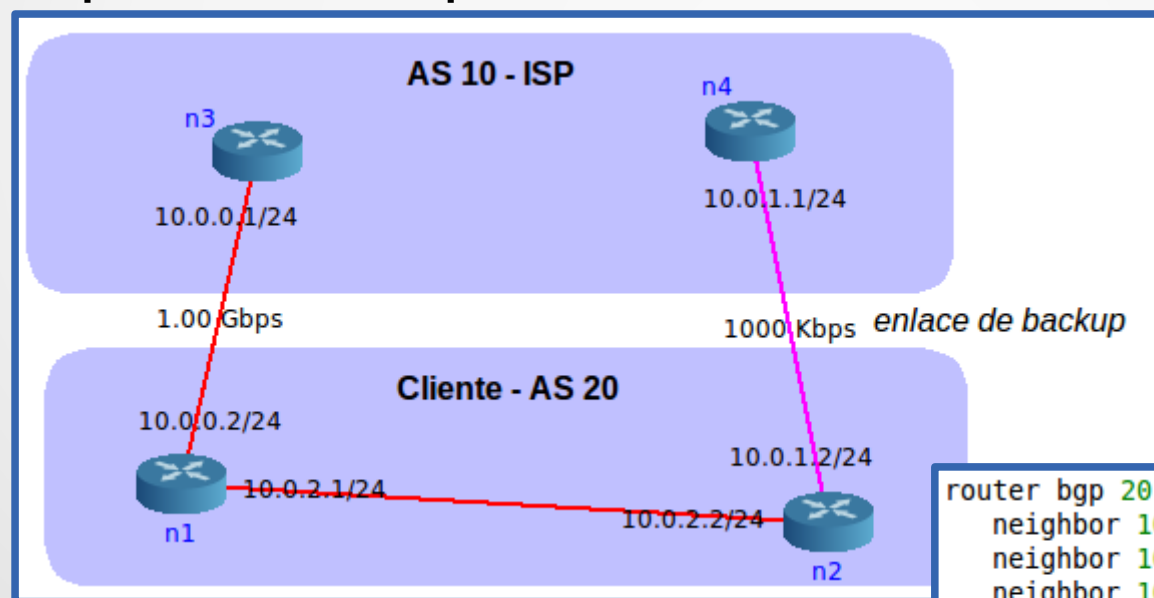
- `redistribute ospf`

- `redistribute rip`

```
*> 60.60.60.0/24    220.20.20.10      0          0 40 ?  
*>i110.10.10.0/24  192.168.0.5       0      100    0 10 i
```

AS_PATH / AS_PREPEND

- Cuando un AS enseña una ruta, se le agrega al principio del AS_PATH su número de AS
- Se puede manipular mediante la técnica AS_PREPEND



```
router bgp 20
  neighbor 10.0.2.1 remote-as 20
  neighbor 10.0.1.1 remote-as 10
  neighbor 10.0.1.1 route-map PREPENDEANDO out
  network 10.0.2.0/24

route-map PREPENDEANDO permit 10
  set as-path prepend 20 20 20 20 20
```

Next hop

Indica la dirección IP para llegar al próximo AS de una ruta recibida

- Se puede manipular manualmente, por ejemplo:

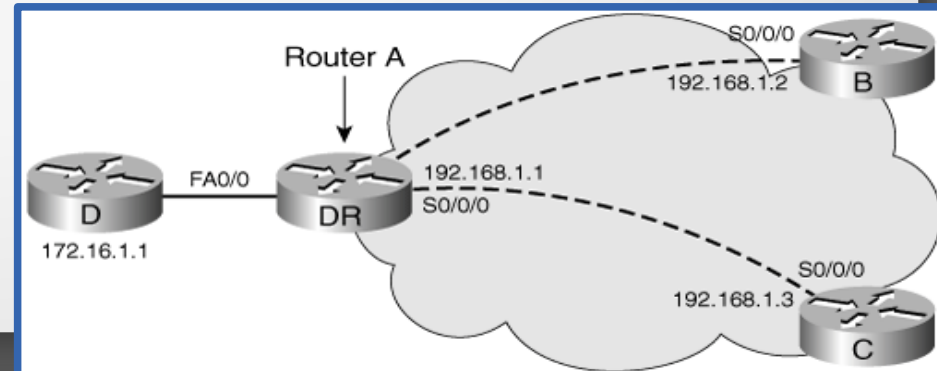
- Para que un peer iBGP sea el next_hop

- neighbor x.x.x.x next-hop-self**

- En redes multiacceso, se hace automáticamente

- En redes NBMA (NoBroadcast MultiAccess) es necesario usar el comando

- neighbor x.x.x.x next-hop-self**



Atributo Community

- Atributo definido en la RFC 1997
- Es opcional y transitivo
- No hay ninguna RFC ni otro estándar que especifiquen cómo las comunidades deben ser aplicadas en políticas de ruteo en Internet.

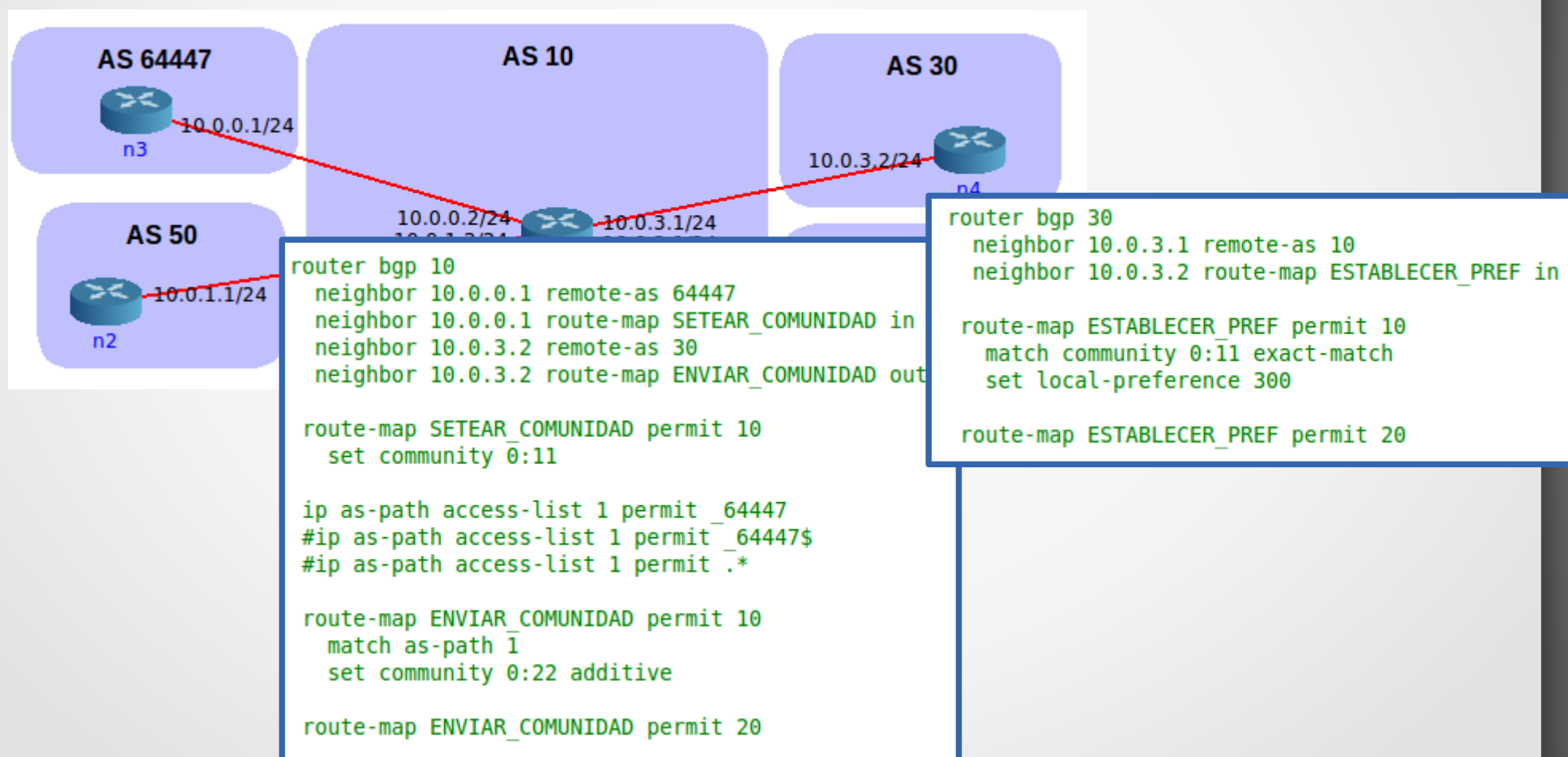
Comunidades Well Known

Existen algunas comunidades predefinidas:

- **no-export**
Do not advertise this route to eBGP peers
- **no-advertise**
Do not advertise this route to any peer (internal or external)
- **internet**
Advertise this route to the Internet community. All routers belong to it

Comunidades – Ejemplo de uso

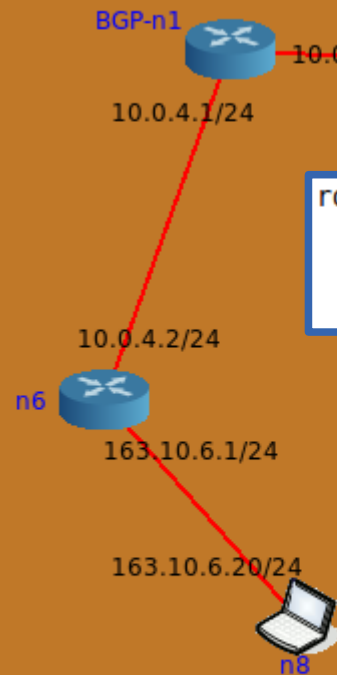
- Se pueden hacer agrupaciones de rutas a las cuales se les aplican decisiones de ruteo:



Peerings / Transit AS

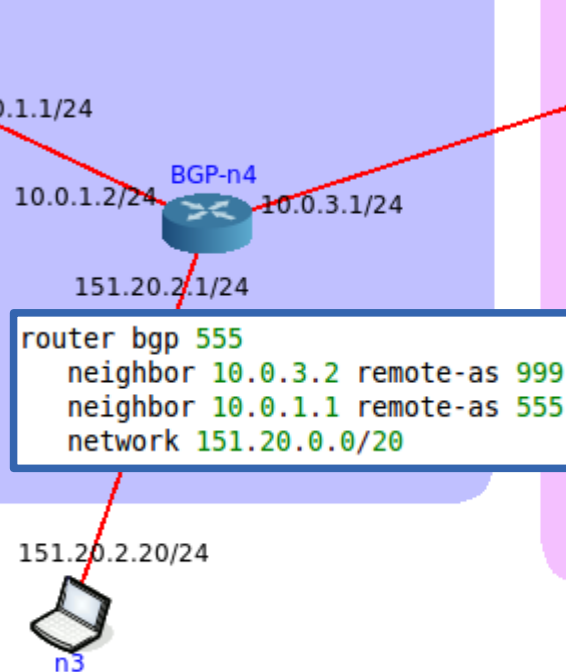
```
router bgp 1
  neighbor 10.0.0.2 remote-as 555
  network 163.10.0.0/16
```

AS 1 - Bloque IP 163.10.0.0/16



```
router bgp 555
  neighbor 10.0.0.1 remote-as 1
  neighbor 10.0.1.2 remote-as 555
  network 151.20.0.0/20
```

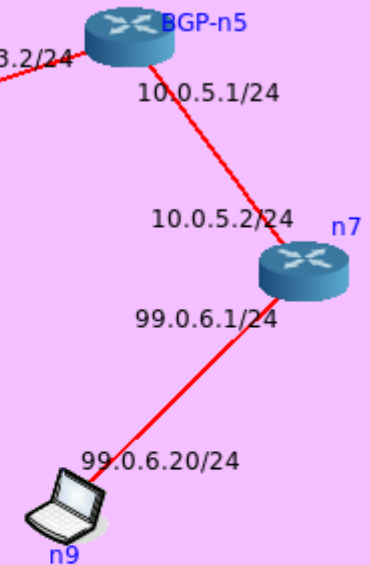
AS 555 - Bloque IP propio 151.20.0.0/20



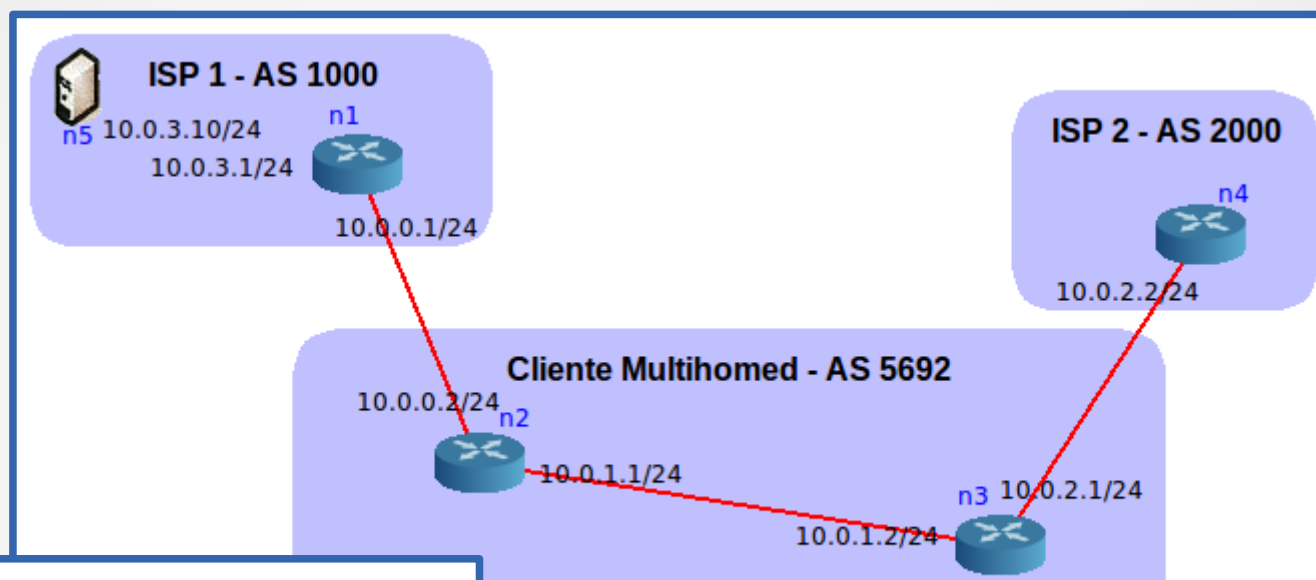
```
router bgp 555
  neighbor 10.0.3.2 remote-as 999
  neighbor 10.0.1.1 remote-as 555
  network 151.20.0.0/20
```

```
router bgp 999
  neighbor 10.0.3.1 remote-as 555
  network 99.0.0.0/8
```

AS 999 - Bloque IP 99.0.0.0/8



Multihome non Transit AS



```
router bgp 5692
  neighbor 10.0.1.2 remote-as 5692
  neighbor 10.0.1.2 next-hop-self
  neighbor 10.0.0.1 remote-as 1000
  neighbor 10.0.0.1 route-map localonly out
  network 10.0.1.0/24
```

```
ip as-path access-list 3 permit ^$
ip as-path access-list 3 deny .*
```

```
route-map localonly permit 10
  match as-path 3
```

!

```
router bgp 5692
  neighbor 10.0.1.1 remote-as 5692
  neighbor 10.0.1.1 next-hop-self
  neighbor 10.0.2.2 remote-as 2000
  neighbor 10.0.2.2 route-map localonly out
  network 10.0.1.0/24
```

```
ip as-path access-list 3 permit ^$
ip as-path access-list 3 deny .*
```

```
route-map localonly permit 10
  match as-path 3
```

!

Proceso de elección de la mejor ruta

1. Si el **NextHop** es **inalcanzable**, descartar el update
2. Prefiere el **mayor Weight**
3. Si tienen el mismo weight prefiere el camino con **mayor Local Preference**
4. Si tienen la misma Local Preference prefiere rutas **originadas por el propio router**
5. Si no hay desempate, prefiere la que tenga el **AS_PATH** más **corto**

...

...

Proceso de elección de la mejor ruta (cont)

6. Si los AS_PATH son de la misma longitud, prefiere la ruta con el **origin code mas bajo (IGP < EGP < INCOMPLETE)**
7. Si los origin codes son iguales, prefiere el camino con **menor MED (MULTI_EXIT_DISC)**
8. Si tienen el mismo MED, se da preferencia a caminos **externos (eBGP) sobre caminos internos (iBGP)**
9. Prefiere la ruta a través del **vecino IGP más cercano**
10. **Prefiere la ruta con el valor de BGP router ID más bajo**