



# Enrutamiento en IPv6

Material preparado por Mariela Rocha y Guillermo Cicileo

# OSPF e IS-IS

---

- IGP
- Utilizan una base de datos link-state (OSPF usa LSAs e IS-IS LSPs)
- Realizan sus cálculos en base a algoritmos SPF
- Utilizan areas para conformar dos niveles de jerarquía
- Son classless y pueden sumarizar redes dentro de las areas



# OSPF IPv6

---

## OSPF para IPv6

- ▶ OSPF para IPv6 – RFC 5340 (OSPFv3)
- ▶ OSPFv3: La version 3 de OSPF fue creada para que, a diferencia de la versión 2, pueda soportar direccionamiento IPv6.
- ▶ La mayoría de las características son las mismas en una versión que en la otra
- ▶ En OSPF para IPv6, el “routing process” no necesita ser explícitamente creado
- ▶ Habilitando OSPF para IPv6 en la interfaz, el proceso será creado.

---

## OSPF para IPv6

- ▶ En OSPF para IPv6, cada interfaz debe ser habilitada con un comando en modo de configuración de interfaz.
- ▶ Esto lo diferencia de OSPFv2, donde las interfaces quedan automáticamente habilitadas con un comando de configuración global.
- ▶ Al mismo tiempo, se pueden configurar varios prefijos en una única interfaz.

---

## OSPF para IPv6

- ▶ Cuando hacemos esto, en OSPF para IPv6, todos los prefijos de la interfaz serán anunciados por OSPF.
- ▶ No se podrá elegir qué prefijos serán importados dentro del OSPF (todos o ninguno)

---

## OSPF para IPv6

- ▶ Tener en cuenta antes de habilitar OSPF para IPv6 en una interfaz:
  - ▶ Planifique su red IPv6 y la estrategia del OSPF
  - ▶ Habilite el ruteo IPv6 para unicast
  - ▶ Habilite OSPF para IPv6 en las interfaces comprometidas
  - ▶ Defina el rango de prefijos que utilizará en las distintas areas, y si estos pueden ser sumarizados

---

## OSPF para IPv6

- ▶ Habilitando el ruteo para IPv6:
  - Router> enable
  - Router# configure terminal
  - Router(config)# ipv6 unicast-routing
- ▶ Habilitando OSPF para IPv6
  - Router> enable
  - Router# configure terminal
  - Router(config)# interface <type> <number>
  - Router(config-if)# ipv6 ospf <process-id> area <area-id>



---

## OSPF IPv6

### ► Definiendo Area Range

- Router> enable
- Router# configure terminal
- Router(config)# ipv6 router ospf <process-id>
- Router(config-rtr)# area <area-id> range <ipv6-prefix/prefix-length>

---

## Ejemplo:

```
interface Ethernet7/0
  ipv6 address 2001:DB8:0:7::1/64
  ipv6 ospf 1 area 1
!
interface Ethernet8/0
  ipv6 address 2001:DB8:0:8::1/64
  ipv6 ospf 1 area 1
!
interface Ethernet9/0
  ipv6 address 2001:DB8:0:9::9/64
  ipv6 ospf 1 area 1
!
ipv6 router ospf 1
  router-id 10.11.11.1
  area 1 range 2001:DB8::/48
```



# IS-IS IPv6

---

## IS-IS IPv6

- ▶ Las características de IPv6 para IS-IS permiten que se sumen a las rutas IPv4, los prefijos IPv6.
- ▶ Se crea un nuevo address family para incluir IPv6
- ▶ IS-IS IPv6 soporta tanto single-topology como multiple-topology.

---

## IS-IS Single Topology

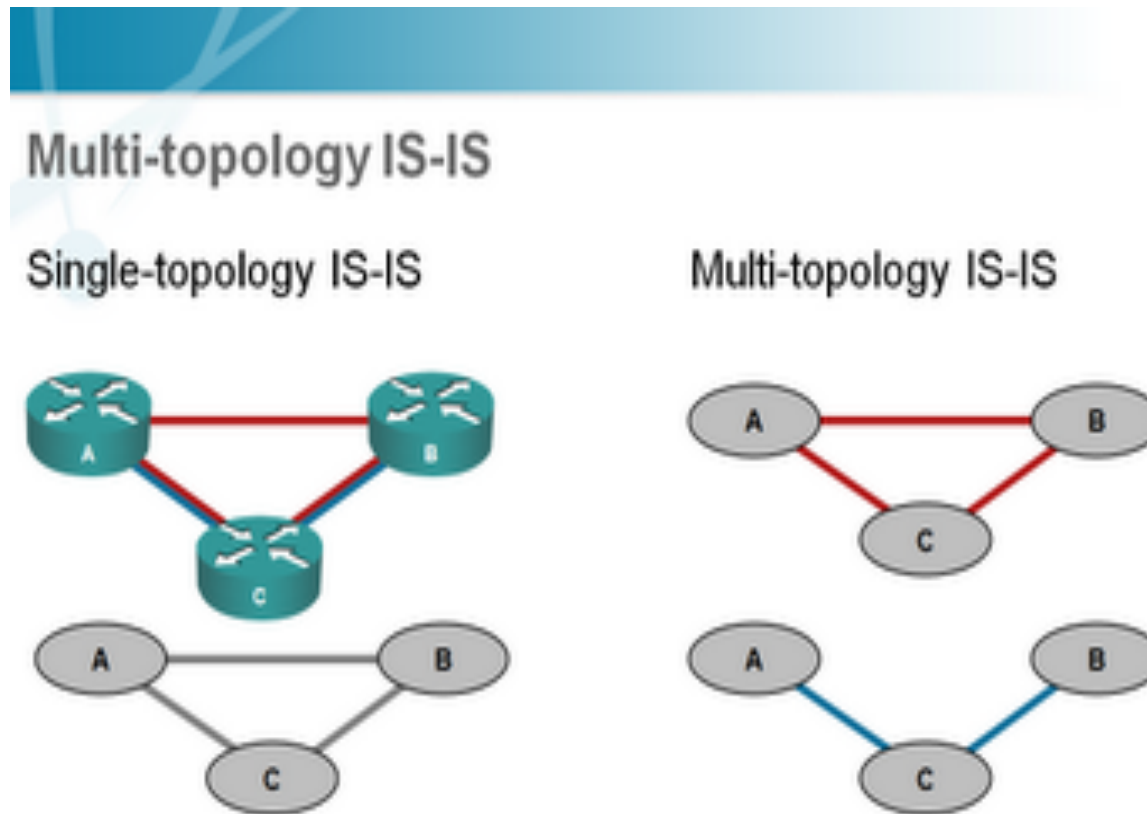
- ▶ IS-IS tiene la particularidad de soportar múltiples protocolos de capa 3.
- ▶ Si tenemos IS-IS con otro protocolo (por ej: IPv4) configurado en una interfaz, podemos configurar también IS-IS para IPv6.
- ▶ Todas las interfaces deben ser configuradas en forma idéntica en cada address family (misma topología), tanto para los routers L1 como los L2.

---

## IS-IS Multi-Topology

- ▶ IS-IS multi-topology permite mantener topologías independientes dentro de un área.
- ▶ Elimina la restricción para todas las interfaces de tener idénticas topologías por cada address family
- ▶ Los routers construyen una topología por cada protocolo de capa 3, por lo que, pueden encontrar el camino óptimo (SPF) aun si algún link soporta solo uno de estos protocolos.

Supongamos un camino de A hasta B:



---

## Configurando IS-IS IPv6

- ▶ Comprende 2 pasos:
  - ▶ Antes que nada, crear un proceso IS-IS routing process, esto independiente del protocolo.
  - ▶ La segunda actividad es configurar el protocolo IS-IS en la interfaz
- ▶ Pre-requisito:
  - ▶ Tener IPv6 unicast-routing habilitado



---

## Configurando IS-IS IPv6

- ▶ Configurando el proceso IS-IS
  - Router> enable
  - Router# configure terminal
  - Router(config)# router isis <area-name>
  - Router(config-router)# net <network-entity-title>

---

## Configurando IS-IS IPv6

- ▶ Configurando las interfaces:
  - Router> enable
  - Router# configure terminal
  - Router(config)# interface <type> <number>
  - Router(config-if)# ipv6 address <ipv6-prefix/prefix-length>
  - Router(config-if)# ipv6 router isis <area-name>

---

## Configurando IS-IS IPv6

### ► IS-IS Multi-topology

- Router> enable
- Router# configure terminal
- Router(config)# router isis <area-name>
- Router(config-router)# metric-style wide [level-1 | level-2 | level-1-2]
- Router(config-router)# address-family ipv6 [unicast | multicast]
- Router(config-router-af)# multi-topology

# Ejemplo

---

```
interface Serial0/1
description link RT3
ipv6 address 2001:DB8:0:3::1/64
ipv6 router isis BB
```

```
interface Serial0/2
description link to RT4
ipv6 address 2001:DB8:0:1::1/64
ipv6 router isis A1
```

```
router isis BB
net 49.0001.0000.0000.0001.00
!
router isis A1
net 49.0001.0000.0000.0001.00
is-type level-1
```

---



**BGP-4**

# BGP versión 4

---

- BGP lleva sólo 3 tipos de información que es específica de IPv4:
  - Los NLRI en los mensajes UPDATES contienen un prefijo IPv4
  - El atributo NEXT\_HOP en un mensaje UPDATE contiene una dirección IPv4
  - El BGP ID en el atributo AGGREGATOR

# BGP Multiprotocolo

---

- La RFC 4760 define extensiones para BGP para soportar múltiples protocolos:
  - De esta forma se puede llevar información en BGP para diferentes protocolos de red
  - Tenemos distintas “address families”: IPv4 unicast, IPv6 unicast, IPv6 multicast, VPNv4...
  - Nuevos atributos BGP:
    - MP\_REACH\_NLRI
    - MP\_UNREACH\_NLRI
  - NEXT\_HOP y NLRI independientes del protocolo

## BGP con IPv6

---

- BGP utiliza un *router ID* para identificar los peers BGP
- Ese router-ID es un valor de 32 bits que normalmente es una dirección IPv4
  - Por defecto, en Cisco se utiliza la dirección IPv4 de una interfaz loopback
- Si el proceso BGP utilizará sólo IPv6, entonces es necesario configurar manualmente el Router ID



# AFI y SAFI

---

- AFI – Address Family Identifier

- 1 -> IPv4

- 2 -> IPv6

- SAFI – Susequent Address Family Identifier

- 1 -> Network Layer Reachability Information usada para unicast forwarding

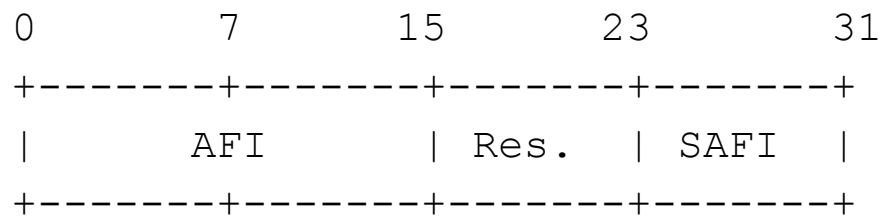
- 2 -> Network Layer Reachability Information usada para multicast forwarding

- 3 -> Network Layer Reachability Information usada tanto para unicast como multicast forwarding

# BGP Capability Advertisement

---

- Un peer BGP que usa extensiones multiprotocolo debe anunciarlo al establecer la sesión (“capabilities advertisement”)



- ☐ AFI - Address Family Identifier (16 bit)
  - ☐ Res. - Reserved (8 bit). Seteado en 0
  - ☐ SAFI - Subsequent Address Family Identifier (8 bit)
- Se deben incluir múltiples pares <AFI, SAFI> en caso de utilizar distintas address families

# Configuración de BGP en IPv6

---

- Fuera de lo mencionado, no tiene grandes diferencias con IPv4
- Es necesario configurar dentro de cada address family las redes que se van a publicar
- Los neighbors se deben activar en cada address family
- Se pueden aplicar filtros y políticas similares a las que existen en IPv4

---

## BGP con IPv6: configuración

### ■ Configuración de un peer:

enable

configure terminal

router bgp *xxxx*

neighbor *ipv6-address* remote-as *autonomous-system-number*

address-family ipv6 [unicast | multicast]

neighbor *ipv6-address activate*

---

## BGP con IPv6: prefix-list

- prefix-list para Bogus Routes básica

```
ipv6 prefix-list IPv6-BOGUS deny 2001:db8::/32 le 128
```

```
ipv6 prefix-list IPv6-BOGUS permit 2002::/16
```

```
ipv6 prefix-list IPv6-BOGUS deny 2002::/16 le 128
```

```
ipv6 prefix-list IPv6-BOGUS deny 0000::/8 le 128
```

```
ipv6 prefix-list IPv6-BOGUS deny fe00::/9 le 128
```

```
ipv6 prefix-list IPv6-BOGUS deny ff00::/8 le 128
```

```
ipv6 prefix-list IPv6-BOGUS permit 0::/0 le 48
```

```
ipv6 prefix-list IPv6-BOGUS deny 0::/0 le 128
```

# BGP con IPv6: prefix-list

---

## ■ Prefix-list rationale

- 2001:db8::/32 – IPv6 documentation prefix (RFC3849)
- 2002::/16 – only permits the /16 - no more-specifics
- 0000::/8 – is denied (loopback, unspecified, v4-mapped)
- FE00::/9 and FF00::/8 – multicast ranges are denied (RFC3513)
- 0::0/0 – all the rest of the IPv6 unicast address space is permitted
- 3FFE::/16 (6bone) has special treatment according to the 6bone rules

# Ejemplo

---

- ▶ router bgp 100
- ▶ bgp log-neighbor-changes
- ▶ no bgp default ipv4-unicast
- ▶ bgp router-id 192.168.99.70
- ▶ !
- ▶ neighbor 2001:db8:1::10 remote-as 100
- ▶ neighbor 2001:db8:1::10 update-source Loopback0
- ▶ !
- ▶ address-family ipv6
- ▶ neighbor 2001:db8:1::10 activate
- ▶ no synchronization
- ▶ network 2001:db8:16:1::/64
- ▶ network 2001:db8:16:100::/64
- ▶ exit-address-family

# Comandos show

---

- La sintaxis es ligeramente diferente en IPv6:
  - Show bgp ipv6 [unicast | multicast] summary
  - Show bgp ipv6 [unicast | multicast] <prefijo>
  - Show bgp ipv6 [unicast | multicast] neighbor <peer>
  - Show bgp ipv6 [unicast | multicast] neighbor <peer> advertised-routes
  - Show bgp ipv6 [unicast | multicast] neighbor <peer> routes
  - Show bgp ipv6 [unicast | multicast] neighbor <peer> received-routes



# Referencias

---

- Proyecto 6deploy: [www.6deploy.org](http://www.6deploy.org)
- Curso IPv6 NIC.Br: [www.ipv6.br](http://www.ipv6.br)
- Curso de enrutamiento avanzado de RedCLARA
- RFC 4760
- RFC 2842