Práctica

Configuración Routing / resumen comandos

Comandos Cisco (v1.7)

Para entrar en modo configuration:

```
#conf t
(config)#
```

Para guardar configuración:

```
# write mem
o
# copy running-config startup-config
```

1. Habilitar IPv6 en una interfaz

Nota: Al configurar una dirección IPv6 en una interfaz, se habilita automáticamente IPv6 en esa interfaz. Lleva implícito el "ipv6 enable"

```
interface xxxxx
   ipv6 enable
```

2. Configurar una dirección (varias formas)

```
interface xxxxx
   ! Direccion General
   ipv6 address X:X:X:X::X/<0-128>
   ! Dirección link-local
   ipv6 address X:X:X:X::X
   ! Autoconfiguración
   ipv6 address autoconfig
```

Ejemplo (Interfaz LAN doble-pila)

```
interface Ethernet0/0
   ip address 192.168.1.254 255.255.255.0
   ipv6 address 2001:db8:123:1::2/64
```

Habilitar routing IPv6 en IOS

```
! activate IPv6 unicast routing
ipv6 unicast-routing
! activate IPv6 Cisco Express forwarding
ipv6 cef
```

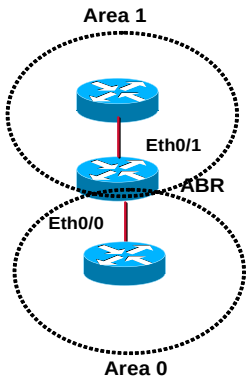
Configurar rutas estáticas

```
! sintaxis
! ipv6 route prefix/prefixlen {next-hop-ipv6-
! address | interface-type interface-number}[admin-distance]
! use ::/0 for default route
ipv6 route ::/0 2001:db8:10a:1001::1
```

Routing (OSPFv3)

```
! "1" is the process ID
 ipv6 router ospf 1
    router-id 2.2.2.2

interface Ethernet0/0
      ipv6 address 2001:db8:1:1::1/64
      ipv6 ospf 1 area 0
      !
interface Ethernet0/1
      ipv6 address 2001:db8:1:2::2/64
      ipv6 ospf 1 area 1
      !
```



Redistribución (OSPFv3) direcciones Loopback

Hay varias maneras de conseguir esto:

1. Redistribución

```
Router(config)# ipv6 router ospf 1
Router(config-rtr)# redistribute connected --> LOOPBACK
Router(config-rtr)# redistribute static ---> Configuradas Estaticas
```

Nota: Las rutas de una interfaz solo se anunciarán si la interfaz está levantada, o si se añade su dirección a la table de routing, por ejemplo introduciendo una ruta estática.:

```
Router1(config)# ipv6 route 2001:DB8:CAFE:A::/64 null 0
```

2. Incluyéndola en OSPFv3 con passive:

```
Router(config)# interface loopback0
Router(config-if)# ipv6 ospf 1 area 0
Router(config-if)# exit
Router(config)# ipv6 router ospf 1
Router(config-rtr)# passive-interface loopback 0
```

Routing (EIGRP)

```
!
ipv6 router eigrp 100
      no shutdown
      eigrp router-id a.b.c.d
!
interface Ethernet0/0
      ipv6 address 2001:db8:c18:1::1/64
      ipv6 eigrp 100
```

Redistribución (EIGRP) direcciones Loopback

```
Router(config)# ipv6 router eigrp 100
Router(config-rtr)# redistribute connected --> LOOPBACK
Router(config-rtr)# redistribute static ---> Configuradas Estáticas
```

Nota: Las rutas de una interfaz solo se anunciarán si la interfaz está levantada, o si se añade su dirección a la table de routing, por ejemplo introduciendo una ruta estática.:

```
Router1(config)# ipv6 route 2001:DB8:CAFE:A::/64 null 0
```

Routing (BGP): eMBGP Peering

```
router bgp <my-as-number>
      no bgp default ipv4-unicast
      bgp log-neighbor-changes
      bgp router-id a.b.d.f
      neighbor X:X:X:X::X remote-as <neighbor-as>
      address-family ipv6 unicast
        neighbor X:X:X:X::X activate
        network 2001:db8::/32
        no synchronization
      exit
```

Routing (BGP): iMBGP Peering

Nota: Para los peerings iMBGP, debes especificar la dirección IPv6 utilizada para las actualizaciones de routing BGP

```
router bgp <my-as-number>
      no bgp default ipv4-unicast
      bgp log-neighbor-changes
      bgp router-id a.b.d.f
      neighbor X:X:X:X::X remote-as <my-as-number>
      neighbor X:X:X:X::X update-source Loopback 0
```

```
address-family ipv6 unicast
  neighbor X:X:X:X::X next-hop-self
  neighbor X:X:X:X::X activate
  no synchronization
exit
```

Routing (BGP): “Inyectar” prefijos IPv6 en BGP

```
Router(config)# ipv6 route 2001:DB8:CAFE:1::/64 Null0
```

Nota: Recordar que solo se pueden anunciar las rutas que existan en la tabla de ruteo propia. Si una ruta no existe hay que “instalarla” en la tabla de rutas.

```
Router(config)# router bgp 65152
Router(config-router)# address-family ipv6 unicast
Router(config-router-af)# network 2001:DB8:CAFE:1::/64
```

Políticas de filtrado de routing

```
! Ejemplo sencillo
! SALIDA: Permito solo mi prefijo asignado 2001:db8::/32
! ENTRADA:
router bgp <my-as-number>
 neighbor X:X:X:X::X remote-as <neighbor-as>
 !
 address-family ipv6
  neighbor X:X:X:X::X activate
  neighbor X:X:X:X::X route-map TRANSITO1_in in
  neighbor X:X:X:X::X route-map TRANSITO1_out out
 exit-address-family
!
ipv6 prefix-list MI_prefijo_1 seq 5 permit 2001:db8::/32
!
!Filtros típicos de entrada, se puede usar otra lista más restrictiva
ipv6 prefix-list global_ipv6 seq 3 deny 3FFE::/16 le 128
ipv6 prefix-list global_ipv6 seq 5 deny 2001:DB8::/32 le 128
ipv6 prefix-list global_ipv6 seq 10 permit 2002::/16
ipv6 prefix-list global_ipv6 seq 15 deny 2002::/16 le 128
ipv6 prefix-list global_ipv6 seq 20 deny ::/8 le 128
ipv6 prefix-list global_ipv6 seq 25 deny FE00::/9 le 128
ipv6 prefix-list global_ipv6 seq 30 permit FF00::/8 le 128
ipv6 prefix-list global_ipv6 seq 35 permit ::/0 le 48
ipv6 prefix-list global_ipv6 seq 40 deny ::/0 le 128
!
route-map TRANSITO1_out permit 10
 match ipv6 address prefix-list MI_prefijo_1
!
route-map TRANSITO1_in permit 10
 match ipv6 address prefix-list global_ipv6
```

Aplicar reglas filtrado a la sesión BGP

Se puede hacer de manera "suave" de una de las dos siguientes maneras:

```
clear bgp ipv6 unicast soft {in|out}
clear bgp ipv6 unicast external {in|out}
```

De una manera más brusca, echando abajo la sesión BGP y volviendo a levantarla:

```
neighbor shutdown
no neighbor shutdown
```

Comandos show

Interfaces IPv6:

```
show ipv6 interface
```

Rutas estáticas IPv6:

```
show ipv6 route
```

Comandos OSPF IPv6:

```
show ipv6 ospf
show ipv6 ospf neighbor
show ipv6 ospf interface
show ipv6 ospf database

show ipv6 route
show ipv6 route ospf
```

Comandos EIGRP IPv6:

```
show ipv6 eigrp neighbor
show ipv6 protocols
show ipv6 eigrp topology all-links
show ipv6 eigrp interfaces detail
show ipv6 route eigrp
```

Comandos BGP IPv6:

```
show bgp ipv6
show bgp ipv6 unicast summary
show bgp ipv6 unicast neighbors
show ipv6 route summary
show bgp ipv6 unicast neighbors <neigh_ipv6_addr> advertised-routes
show bgp ipv6 unicast neighbors <neigh_ipv6_addr> routes

! El siguiente comando necesita inbound soft-reconfiguration
! Aumenta el uso de memoria porque se almacenan los updates recibidos
! R(config-router-af)#neighbor <neigh_ipv6_addr> soft-reconfiguration in
show bgp ipv6 unicast neighbors <neigh_ipv6_addr> received-routes
```

6RD:

```
show tunnel 6rd prefix <ipv6-address-used-in-my-side> tunnel zz
show tunnel 6rd tunnel zz
show interfaces tunnel zz
```

```
show ipv6 interface tunnel zz

show ipv6 general-prefix
```

Configurar un túnel

Configurar un túnel IPv6 in IPv4 (puede ser ipv6ip o GRE, hay que elegir uno)

```
interface tunnel x
   tunnel source (interface_type number | ipv4_address)
   tunnel destination <ipv4_address>
   ipv6 address X:X:X:X::X/<0-128>
   ! ipv6ip for direct tunneling
   tunnel mode ipv6ip
   ! gre for GRE encapsulation
   tunnel mode gre ip
```

Configurar un túnel IPv6 in IPv6 (puede ser ipv6ip o GRE, hay que elegir uno)

```
interface tunnel x
   tunnel source (interface_type number | ipv6_address)
   tunnel destination <ipv6_address>
   ipv6 address X:X:X:X::X/<0-128>
   ! ipv6ip for direct tunneling
   tunnel mode ipv6
   ! gre for GRE encapsulation
   tunnel mode gre ipv6
```

Configure 6RD tunnel (Border Relay)[1]

```
interface tunnel x
      no ip address
      ipv6 enable
      ! "internal" IPv4 interface
      tunnel source (interface_type number | ipv4_address)
      ! Optional, you can use link-local
      ! ipv6 address X:X:X:X::X/<0-128>
      ! specify 6rd IPv6 in IPv4 encapsulation
      tunnel mode ipv6ip 6rd
      ! 6rd specific parameters
      tunnel 6rd ipv4 suffix-len <ipv4-suffix-length>
      tunnel 6rd ipv4 prefix-len <ipv4-prefix-length>
      tunnel 6rd prefix <6rd-ipv6-prefix-used>

! To add a route for all the 6RD prefix
! through the 6RD virtual interface
ipv6 route <6rd-ipv6-prefix-used> tunnelx
```

Configure 6RD tunnel (CPE) [2]

[1] Check Annex A about 6RD configuration
[2] Check Annex A about 6RD configuration

```
interface tunnel x
      no ip address
      ipv6 enable
      ! The WAN IPv4 interface
      tunnel source (interface_type number | ipv4_address)
   ! specify 6rd IPv6 in IPv4 encapsulation
      tunnel mode ipv6ip 6rd
   ! 6rd specific parameters
      tunnel 6rd ipv4 suffix-len <ipv4-suffix-length>
      tunnel 6rd ipv4 prefix-len <ipv4-prefix-length>
      tunnel 6rd prefix <6rd-ipv6-prefix-used>
      tunnel 6rd br <reachable-BR-IPv4-address>

! Define a name for the prefix received by 6RD
ipv6 general-prefix 6rdDelegatedPrefix 6rd Tunnelx

! You can use the defines name for the prefix
interface Loopback 100
      ipv6  address  6rdDelegatedPrefix  ::100:A/128

! null route for the 6rd delegated prefix on this CPE
ipv6 route <6rd-delegated-prefix> Null0
! static route for 6rd prefix to use tunnel as egress interface
ipv6 route <6rd-ipv6-prefix-used> Tunnel x
! default route pointing to subnet anycast address of BR's
! 6rd delegated prefix
ipv6 route ::/0 <BRs-6rd-delegated-prefix>::
```

Access Control Lists (ACL)

```
ipv6 access-list vty-ipv6
      permit tcp 2001:db8:0:401::/64 any eq telnet
      deny ipv6 any any log-input
```

Aplicar una ACL a una interfaz

```
ipv6 traffic-filter <acl_name> in | out
```

Restringiendo el acceso al router

```
ipv6 access-class <acl_name> in | out
```

Aplicar una ACL para filtrar tráfico de debug

```
debug ipv6 packet [access-list <acl_name>] [detail]
```

ANNEX A: 6RD Configuration Details

6RD tunnels are a new feature in IOS 15.1(3)T. You must be running at least this version to implement 6RD. From Cisco's point of view [2]:

IPv6 Rapid Deployment Tunnels

The 6RD feature is an extension of the 6to4 feature. The 6RD feature allows a service provider (SP) to provide a unicast IPv6 service to customers over its IPv4 network by using encapsulation of IPv6 in IPv4.

The main differences between 6RD and 6to4 tunneling are as follows:

- 6RD does not require addresses to have a 2002::/16 prefix; therefore, the prefix can be from the SP's own address block. This function allows the 6RD operational domain to be within the SP network. From the perspective of customer sites and the general IPv6 internet connected to a 6RD-enabled SP network, the IPv6 service provided is equivalent to native IPv6.
- All 32 bits of the IPv4 destination need not be carried in the IPv6 payload header. The IPv4 destination is obtained from a combination of bits in the payload header and information on the router. Furthermore, the IPv4 address is not at a fixed location in the IPv6 header as it is in 6to4.

The 6RD SP prefix was selected by the SP for the IPv6 deployment shown in Figure 2. The **6RD delegated prefix is derived from the SP prefix and the IPv4 address bits**, and is used by the CE for hosts within its site.

Figure 2 6RD Deployment

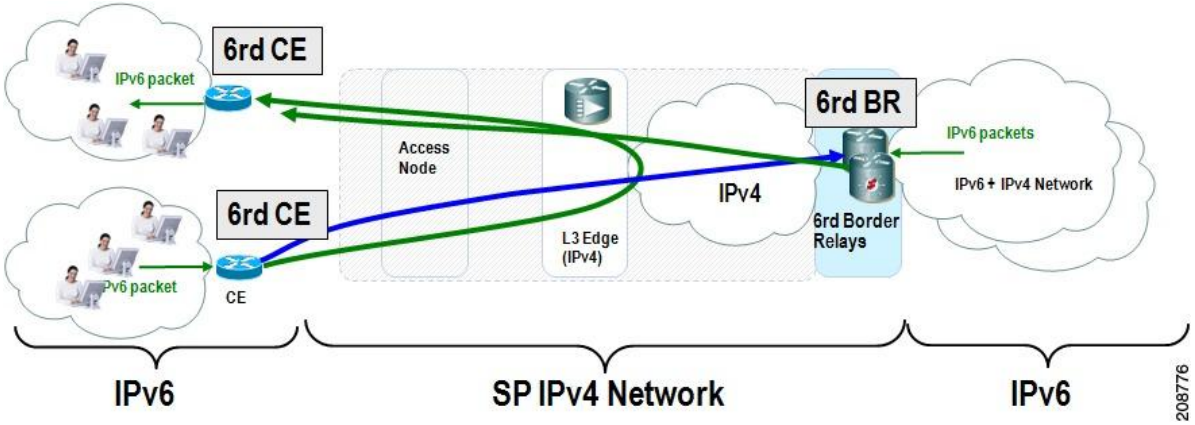


Figure 3 shows how 6RD prefix delegation works.

Figure 3 6RD Prefix Delegation Explanation

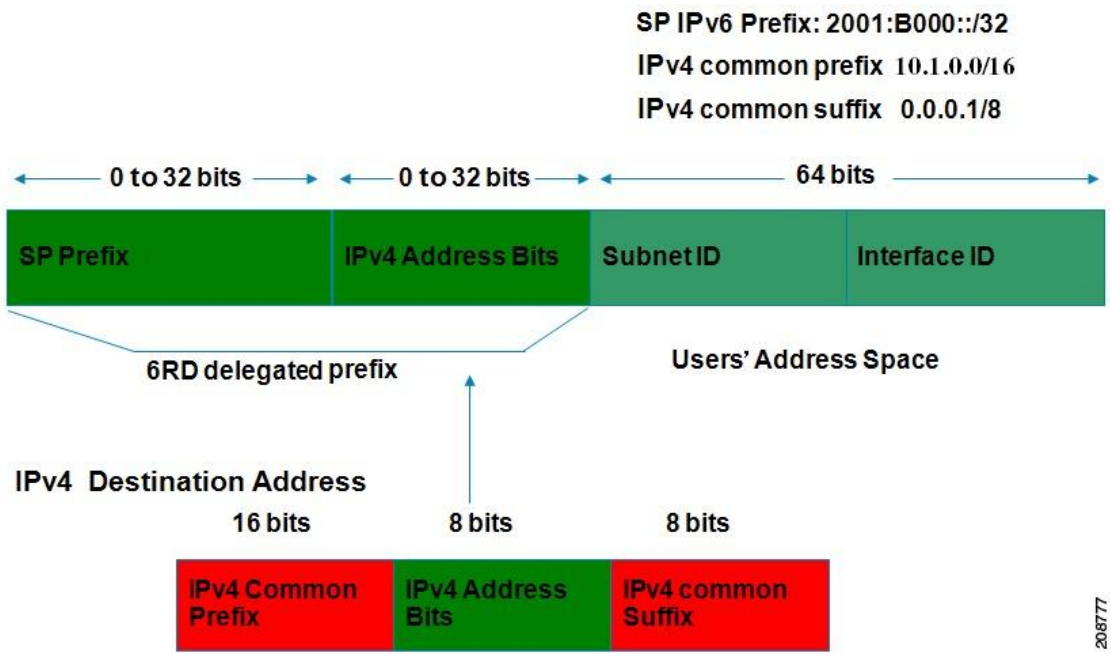
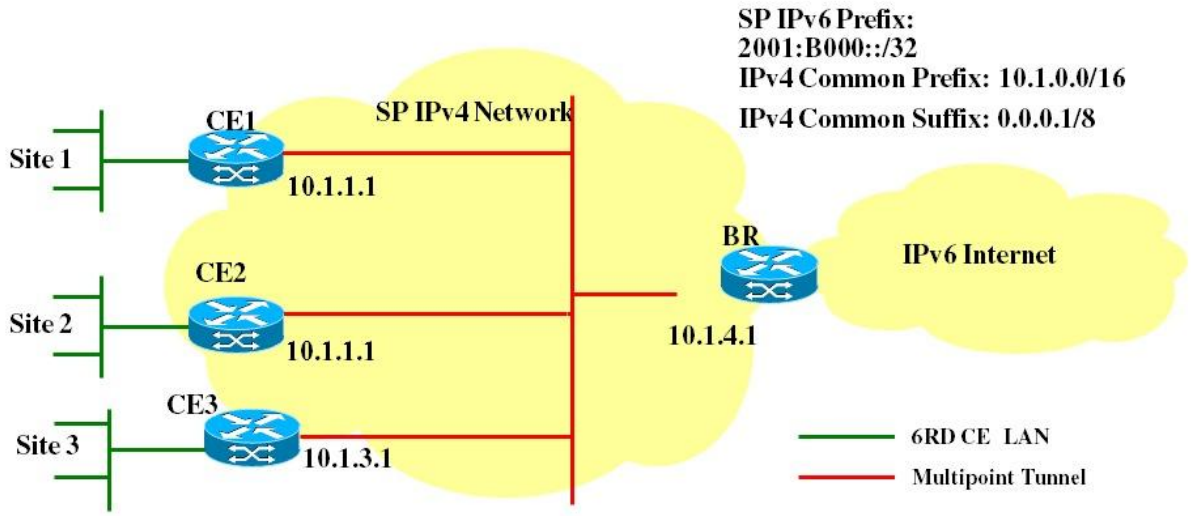


Figure 4 shows a 6RD prefix delegation topology.

Figure 4 6RD Prefix Delegation and Explanation



SP Prefix	2001:B000::/32
IPv4 Common Prefix	10.1.0.0/16
IPv4 Common Suffix	0.0.0.1/8
CE1: Delegated 6RD prefix	2001:B000:0100::/40
CE2: Delegated 6RD prefix	2001:B000:0200::/40
BR: Delegated 6RD prefix	2001:B000:0400::/40
CE1 (IPv4) tunnel transport source	10.1.1.1
CE2 (IPv4) tunnel transport source	10.1.2.1
BR (IPv4) tunnel transport source	10.1.4.1

208778

Following this example the configuration for Cisco IOS in BR and CE1 would be (just the variable part):

Configure 6RD tunnel (BR)

```
interface tunnel x
      ! we can use the IP or interface type/number
      tunnel source 10.1.4.1

      ! 6rd specific parameters
      tunnel 6rd ipv4 suffix-len 8
      tunnel 6rd ipv4 prefix-len 16
      tunnel 6rd prefix 2001:B000::/32
```

Configure 6RD tunnel (CPE)

```
interface tunnel x
      ! we can use the IP or interface type/number
      tunnel source 10.1.1.1

      ! 6rd specific parameters
      tunnel 6rd ipv4 suffix-len 8
      tunnel 6rd ipv4 prefix-len 16
      tunnel 6rd prefix 2001:B000::/32
      tunnel 6rd br 10.1.4.1
```

[1] Enterprise IPv6 Solution: IPv6 Rapid Deployment: Provide IPv6 Access to Customers over an IPv4-Only Network
http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6553/whitepaper_c11-665758.html
[2] Networking Software (IOS & NX-OS): Implementing Tunneling for IPv6
http://www.cisco.com/en/US/docs/ios/ios_xe/ipv6/configuration/guide/ip6-tunnel_xe.html#wp1066434
[3] Comcast 6RD Configuration Example - Home Router:
https://supportforums.cisco.com/docs/DOC-15565
[4] IPv6 Configuration Guide, Cisco IOS Release 15.1M&T: Implementing Tunneling for IPv6
http://www.cisco.com/en/US/docs/ios-xml/ios/ipv6/configuration/15-1mt/ip6-tunnel.html
[5] CCIE Blog: IPv6 Rapid Deployment – 6RD
http://blog.initialdraft.com/archives/561/