Unveiling IPv6: The Next-Generation Internet Protocol

Table of Contents

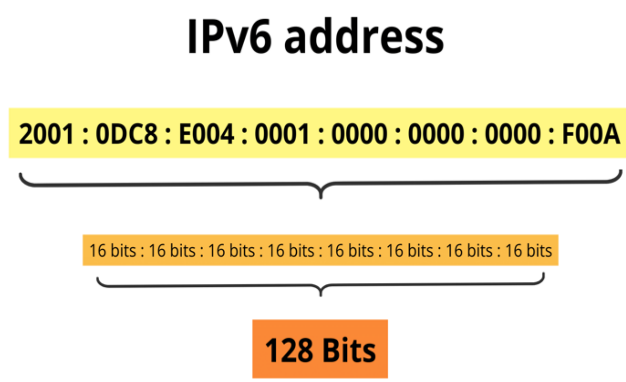
History

- 1992 – People realized IPv4 addresses would run out.
- 1994 – Work on IPv6 started.
- 1998 – First IPv6 standard (RFC 2460) came out.
- 2006 – IPv6 address structure was finalized (RFC 4291).
- 2008 – U.S. government required IPv6 support in federal networks.
- 2015 – ARIN ran out of IPv4 addresses.

Addressing System

Address Length and Representation Rules

- IPv6 addresses are 128 bits long.
- Colon ":" used to separate group of four-hex characters (a "word")
- 4 bits = 1 hex character
- Written as 8 groups of 4 hexadecimal digits
- Leading zeros in a group can be skipped.
    - Example: 2001:0db8::1 instead of 2001:0db8:0000:0000:0000:0000:0000:0001.
- One sequence of consecutive zeros can be replaced with ::.



IPv6 Host Address Structure

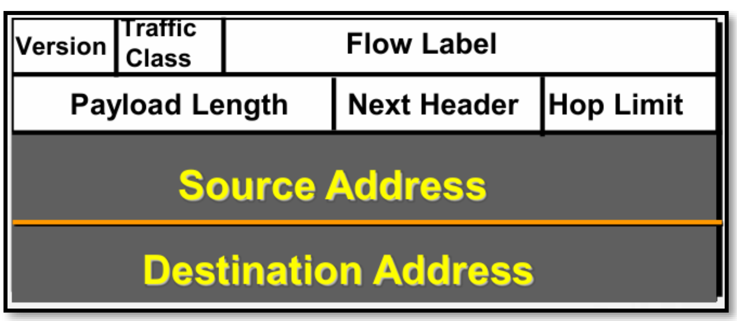
An IPv6 address has two main parts:

- Network/Subnet Portion (Prefix)
    - This shows the network where the host belongs.
    - Example: 2001:aabb:cc11::/64 → the first 64 bits are the network prefix.

- Interface Identifier (Host Part)
    - This identifies the specific device (host) inside that network.
    - Example: in 2001:aabb:cc11::3a/64, the last 64 bits (...3a) are the host part.
- Dynamic Interface ID (EUI-64)
    - IPv6 can auto-generate the host part using the device's MAC address.
    - This is called EUI-64 format
    - Example:
      Network Prefix = 2001:db8:abcd:1::/64
      MAC = AA:BB:CC:11:22:33
      EUI-64 = A8BB:CCFF:FE11:2233
      Full IPv6 Address = 2001:db8:abcd:1:A8BB:CCFF:FE11:2233

IPv6 Header

The IPv6 header has a fixed size of 40 bytes, making it simpler than the variable IPv4 header (20–60 bytes). It removes fields like checksum and fragmentation to reduce processing overhead. Instead, IPv6 uses extension headers for optional features. Routers no longer fragment packets. IPv6 also has built-in support for IPsec, providing better security. Overall, the IPv6 header is simpler, faster, and more efficient than IPv4.



- **Version** → Identifies IP version (IPv6 = 6).
- **Traffic Class** → Defines packet priority (like QoS).
- **Flow Label** → Used for special handling of packet flows.
- **Payload Length** → Size of the data carried after the header.
- **Next Header** → Tells which protocol comes next (TCP, UDP, ICMPv6, etc.).

- **Hop Limit** → Maximum hops (like TTL in IPv4).
- **Source Address** → IPv6 address of sender.
- **Destination Address** → IPv6 address of receiver.

Address Type

Link-local addresses

- IPv6 host gets it automatically when it connects.
- Works like IPv4's 169.254.x.x (APIPA).
- Always starts with FE80::/10 → means it can be FE80, FE90, FEA0, or FEB0.
- The last 64 bits are made from the **MAC address**, with **FFFE** added in the middle.

Global Unicast Addresses

- These IPv6 addresses start with the first 3 bits = 001, which means they belong to the range 2000::/3.
- They are called Global Unicast Addresses and work like public IPv4 addresses.
- The global routing prefix is usually 48 bits or less, which identifies the network or organization on the Internet.
- The remaining bits are used for subnetting and host IDs.

Unique Local Addresses

- These addresses are not globally routable (they only work inside a private network).
- They are like private IPv4 addresses (10.x.x.x, 192.168.x.x).
- They always start with FC00::/7, which includes FC00::/8 and FD00::/8.

Multicast Address

- Any address starting with FF00::/8 is an IPv6 multicast address.
- This is because the first 8 bits are always 11111111.

- Every IPv6 device automatically joins some multicast groups, like all-nodes and solicited-node multicast.
- Example: FF02::1 (all-nodes multicast).

Anycast Address

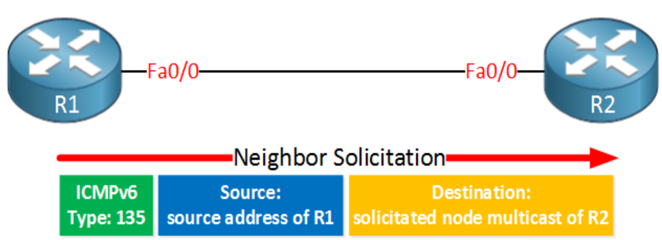
- Two or more nodes can share the same IPv6 address.
- This is used for load balancing purposes.
- Anycast addresses look the same as unicast addresses and cannot be distinguished by their format.

Solicited Node Multicast Address

- A special multicast address used in IPv6 for efficient local communication.
- Prefix is always FF02:0:0:0:0:1:FF/104 (first 104 bits fixed).
- The last 24 bits are copied from the device's Interface ID.
- Automatically created whenever a device configures an IPv6 address, just like Link-Local addresses.
- Every IPv6 unicast or anycast address has a matching solicited-node multicast address.
- Example:
  - IPv6 address = 2001:db8::1a2b:3c4d
  - Interface ID = 1a2b:3c4d
  - last 24 bits = 2b:3c:4d
  - Solicited-Node Multicast = FF02::1:FF2B:3C4D.

Role in NDP (Neighbor Discovery Protocol)

- NDP in IPv6 replaces ARP from IPv4.
- When a device wants to know the MAC address of another IPv6 host, it sends a Neighbor Solicitation (NS) message to that host's Solicited-Node Multicast Address instead of broadcasting to all.



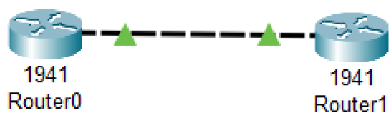
- The target device responds with a Neighbor Advertisement (NA), providing its Layer 2 (MAC) address.
- This makes address resolution more efficient because only the intended device processes the request, not every device on the link.
- NDP also handles other tasks: detecting duplicate addresses (DAD), router discovery, and neighbor reachability.

IPv6 Address Assignment

Static

In static assignment, the IPv6 address is configured manually on a device. The administrator specifies the IPv6 address. This method gives full control but requires more effort in large networks.

Cisco Packet Tracer:



Router0

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gig0/0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

Router(config-if)#ipv6 e
Router(config-if)#ipv6 en
Router(config-if)#ipv6 enable
Router(config-if)#ipv6 add 2001:db8:200:300::1/64
Router(config-if)#ipv6 add fe80::1 lin
Router(config-if)#ipv6 add fe80::1 link-local
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

Router1

```
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gig0/0
Router(config-if)#ipv6 en
Router(config-if)#ipv6 enable
Router(config-if)#ipv6 add 2001:db8:200:300::/64 e
Router(config-if)#ipv6 add 2001:db8:200:300::/64 eui-64
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gig0/0
Router(config-if)#no shut
```

```
Router(config)#do sho ipv6 int bri
GigabitEthernet0/0         [up/up]
    FE80::250:FFF:FEB6:BB01
    2001:DB8:200:300:250:FFF:FEB6:BB01
GigabitEthernet0/1         [administratively down/down]
    unassigned
Vlan1                      [administratively down/down]
    unassigned
Router(config)#
Router(config)#do sho int gig0/0 | i bia
  Hardware is CN Gigabit Ethernet, address is 0050.0fb6.bb01 (bia 0050.0fb6.bb01)
```

Ping (From Router0 to Router1)

```
Router#ping 2001:DB8:200:300:250:FFF:FEB6:BB01

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:200:300:250:FFF:FEB6:BB01, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/1 ms
```

Stateless Address Auto Configuration (SLAAC)

With SLAAC, a host automatically creates its IPv6 address without a DHCP server. The router sends Router Advertisements (RAs) that include the network prefix, and the host generates the interface ID (often using EUI-64). This makes it simple and automatic.

Cisco Packet Tracer:

Router0

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gig0/0
Router(config-if)#ipv6 add 2001:db8:200:300/64 eui-64
                                               ^
% Invalid input detected at '^' marker.

Router(config-if)#ipv6 add 2001:db8:200:300/64 e
Router(config-if)#ipv6 add 2001:db8:200:300::/64 e
Router(config-if)#ipv6 add 2001:db8:200:300::/64 eu
Router(config-if)#ipv6 add 2001:db8:200:300::/64 eui-64
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

Router(config-if)#do show ipv6 int bri
GigabitEthernet0/0         [up/down]
    FE80::206:2AFF:FE3E:4D01
    2001:DB8:200:300:206:2AFF:FE3E:4D01
GigabitEthernet0/1         [administratively down/down]
    unassigned
Vlan1                      [administratively down/down]
    unassigned
Router(config-if)#do show int gig0/0 | i bia
  Hardware is CN Gigabit Ethernet, address is 0006.2a3e.4d01 (bia 0006.2a3e.4d01)

Router(config)#ipv6 unicast-routing
```

Router1

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gig0/0
Router(config-if)#no shut
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 u
Router(config)#ipv6 unicast-routing
Router(config)#int gig0/0
Router(config-if)#ipv6 en
Router(config-if)#ipv6 enable
Router(config-if)#ipv6 add a
Router(config-if)#ipv6 add autoconfig
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#do show ipv6 int bri
GigabitEthernet0/0         [up/up]
    FE80::20B:BEFF:FE49:E201
    2001:DB8:200:300:20B:BEFF:FE49:E201
GigabitEthernet0/1         [administratively down/down]
    unassigned
Vlan1                      [administratively down/down]
    unassigned
Router(config)#do show int gig0/0 | i bia
  Hardware is CN Gigabit Ethernet, address is 000b.be49.e201 (bia 000b.be49.e201)
Router(config)#
```

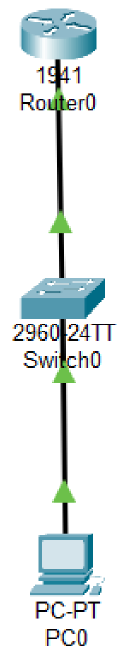
Ping (From Router0 to Router1)

```
Router#
Router#ping 2001:DB8:200:300:20B:BEFF:FE49:E201

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:DB8:200:300:20B:BEFF:FE49:E201, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 0/0/0 ms
```

Stateless DHCPv6

In Stateless DHCPv6, the router provides IPv6 addresses to hosts using **SLAAC (Stateless Address Auto Configuration)**, while the DHCPv6 server only gives **extra information** such as DNS servers and domain name. Unlike Stateful DHCPv6, the server does not assign or keep track of IPv6 addresses — hosts generate their own addresses based on the router's prefix and their interface ID (EUI-64). This makes Stateless DHCPv6 lightweight, combining SLAAC for addressing and DHCPv6 for additional parameters.

Cisco Packet Tracer:

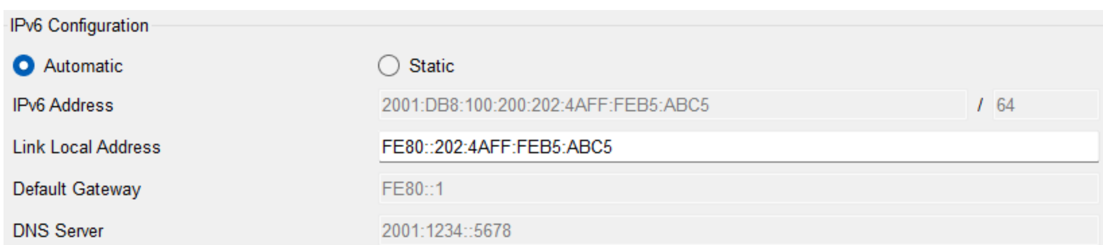


Router0

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 u
Router(config)#ipv6 unicast-routing
Router(config)#int gig0/0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Router(config-if)#ipv6 add fe80::1 l
Router(config-if)#ipv6 add fe80::1 link-local
Router(config-if)#ipv6 add 2001:db8:100:200::1/64
Router(config-if)#
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

```
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 dhcp pool mgk
Router(config-dhcpv6)#dns
Router(config-dhcpv6)#dns
Router(config-dhcpv6)#dns-server 2001:1234::5678
Router(config-dhcpv6)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gig0/0
Router(config-if)#ipv6 d
Router(config-if)#ipv6 dhcp s
Router(config-if)#ipv6 dhcp server m
Router(config-if)#ipv6 dhcp server mgk
Router(config-if)#ipv6 nd o
Router(config-if)#ipv6 nd other-config-flag
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

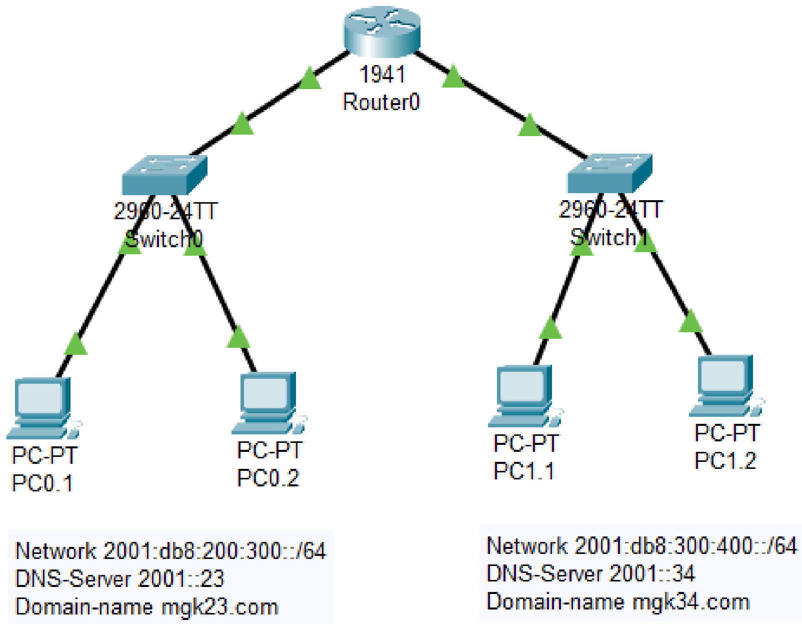
PC0



DHCPv6

In DHCPv6, a DHCP server assigns IPv6 addresses and other configuration details like DNS. Unlike SLAAC, which only provides address information, DHCPv6 can give extra parameters, making it useful for managed networks.

Cisco Packet Tracer:

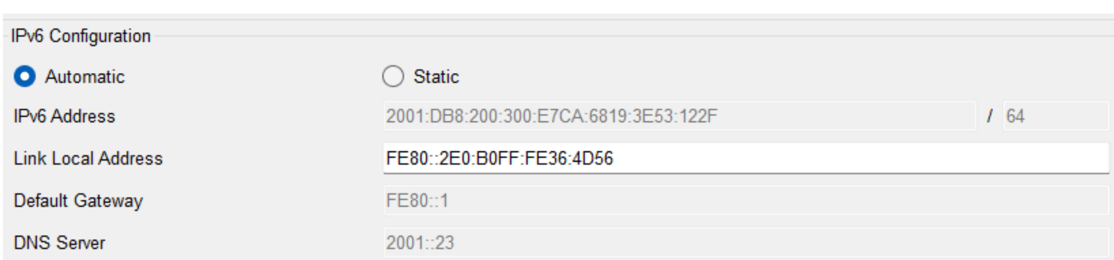


Router0

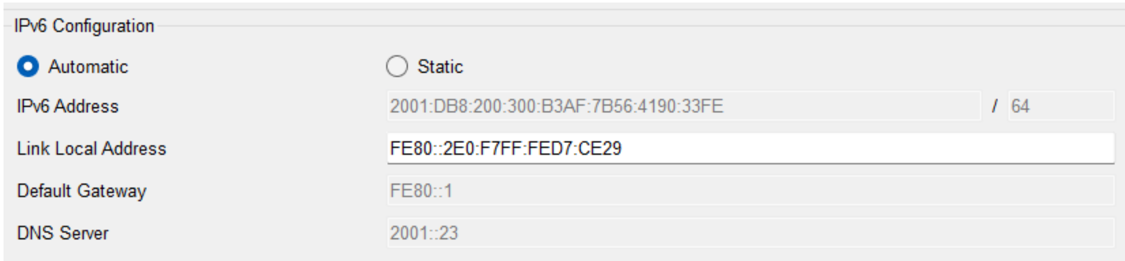
```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 un
Router(config)#ipv6 unicast-routing
Router(config)#ipv6 dhcp pool mgk
Router(config-dhcpv6)#dns
Router(config-dhcpv6)#dns-server 2001::23
Router(config-dhcpv6)#domain
Router(config-dhcpv6)#domain-name mgk23.com
Router(config-dhcpv6)#add
Router(config-dhcpv6)#address p
Router(config-dhcpv6)#address prefix 2001:db8:200:300::/64
Router(config-dhcpv6)#exit
Router(config)#ipv6 dhcp pool mgk1
Router(config-dhcpv6)#dns-server 2001::34
Router(config-dhcpv6)#domain-name mgk34.com
Router(config-dhcpv6)#address prefix 2001:db8:300:400::/64
Router(config-dhcpv6)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

```
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gig0/0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Router(config-if)#ipv6 fe80::1 lin
Router(config-if)#ipv6 add fe80::1 lin
Router(config-if)#ipv6 add fe80::1 link-local
Router(config-if)#ipv6 add 2001:db8:200:300::1/64
Router(config-if)#ipv6 dhcp s
Router(config-if)#ipv6 dhcp server mgk
Router(config-if)#int gig0/1
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

Router(config-if)#ipv6 add fe80::2 link-local
Router(config-if)#ipv6 add 2001:db8:300:400::2/64
Router(config-if)#ipv6 dhcp server mgk1
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

Router(config)#int gig0/0
Router(config-if)#ipv6 d
Router(config-if)#ipv6 nd m
Router(config-if)#ipv6 nd managed-config-flag
Router(config-if)#int gig0/1
Router(config-if)#ipv6 nd managed-config-flag
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

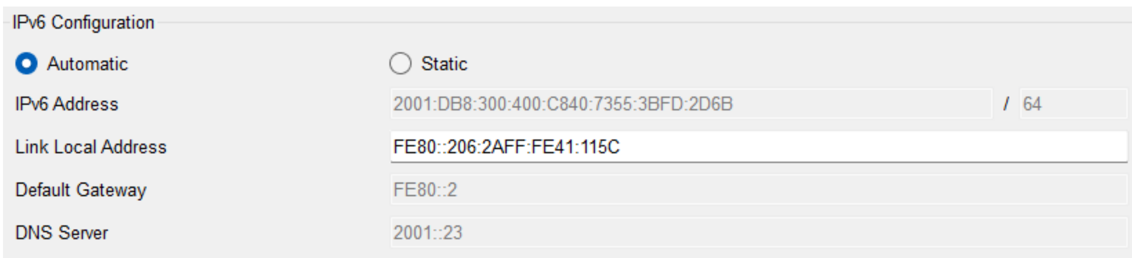
PC0.1



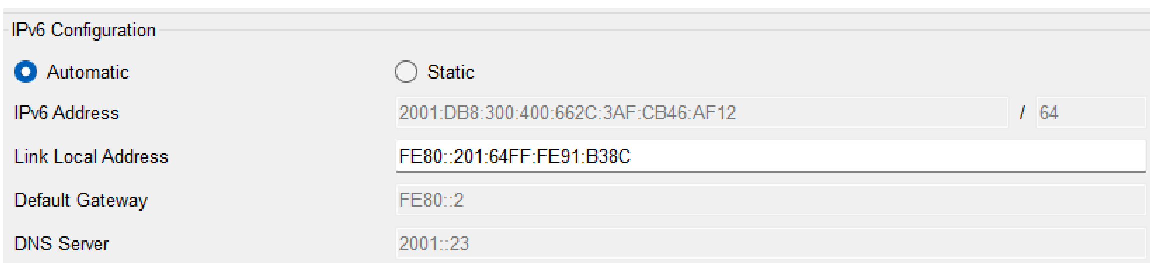
PC0.2



PC1.1



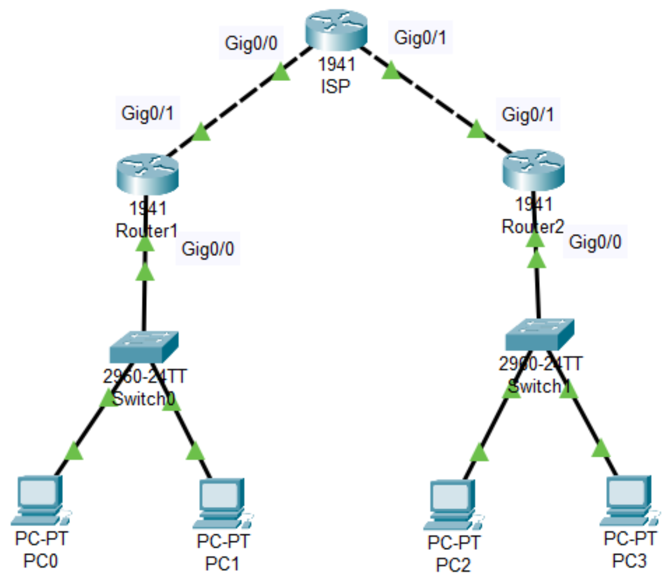
PC1.2



DHCPv6 prefix delegation

This is used by ISPs to provide entire IPv6 prefixes to customer routers. The customer router then uses this delegated prefix to assign addresses to its internal devices. It ensures hierarchical and scalable address distribution.

Cisco Packet Tracer:



ISP

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 u
Router(config)#ipv6 unicast-routing
Router(config)#ipv6 dhcp pool mgk
Router(config-dhcpv6)#dns
Router(config-dhcpv6)#dns-server 2001:db8::23
Router(config-dhcpv6)#pre
Router(config-dhcpv6)#prefix-delegation pool mgk-local
Router(config-dhcpv6)#exit
Router(config)#ipv6 d
Router(config)#ipv6 l
Router(config)#ipv6 local p
Router(config)#ipv6 local pool mgk-local 2001:db8:100::/40 48
Router(config)#ipv6 dhcp pool mgkl
Router(config-dhcpv6)#d
Router(config-dhcpv6)#dn
Router(config-dhcpv6)#dns-server 2001:db8::34
Router(config-dhcpv6)#p
Router(config-dhcpv6)#prefix-delegation pool mgkl-local
Router(config-dhcpv6)#exit
Router(config)#ipv6 local pool ?
  WORD  Create named local prefix pool
Router(config)#ipv6 local pool mgkl-local 2001:db8:200::/40 48
Router(config)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

```
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gig0/0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

Router(config-if)#ipv6 add 2001:db8::200:300::1/64
% Incomplete command.
Router(config-if)#ipv6 add 2001:db8:200:300::1/64
Router(config-if)#ipv6 dhcp ser
Router(config-if)#ipv6 dhcp server mgk
Router(config-if)#int g0/1
Router(config-if)#no shut

Router(config-if)#ipv6 add 2001:DB:300:400::1/64
Router(config-if)#ipv6 dhcp server mgk1
Router(config-if)#end
```

ipv6 local pool mgk-local 2001:db8:100::/40 48
Creates a pool of prefixes (from /40, delegating /48 prefixes to clients).

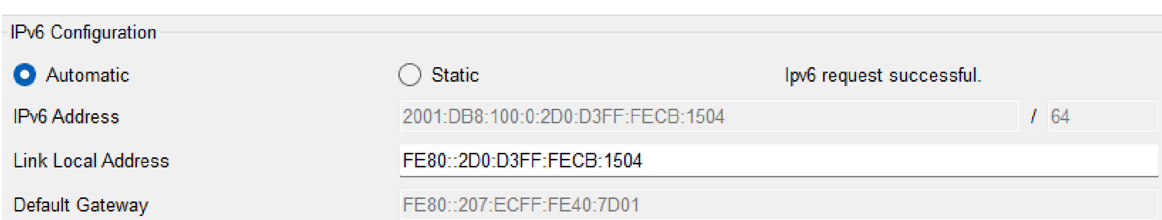
Router1

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 u
Router(config)#ipv6 unicast-routing
Router(config)#int gig0/1
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

Router(config-if)#ipv6 en
Router(config-if)#ipv6 enable
Router(config-if)#ipv6 d
Router(config-if)#ipv6 dhcp c
Router(config-if)#ipv6 dhcp client pd mgk11
Router(config-if)#ipv6 add a
Router(config-if)#ipv6 add autoconfig d
Router(config-if)#ipv6 add autoconfig default
                                      ^
% Invalid input detected at '^' marker.

Router(config-if)#ipv6 add autoconfig
Router(config-if)#int gig0/0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Router(config-if)#ipv6 add mgk11 ::1:0:0:1/64
% Warning: Bits overlapping with general prefix mgk11 will be truncated.
Router(config-if)#end

Router#show ipv6 int bri
GigabitEthernet0/0          [up/up]
    FE80::207:ECFF:FE40:7D01
    2001:DB8:100:0:1::1
GigabitEthernet0/1          [up/up]
    FE80::207:ECFF:FE40:7D02
    2001:DB8:200:300:207:ECFF:FE40:7D02
Vlan1                       [administratively down/down]
```

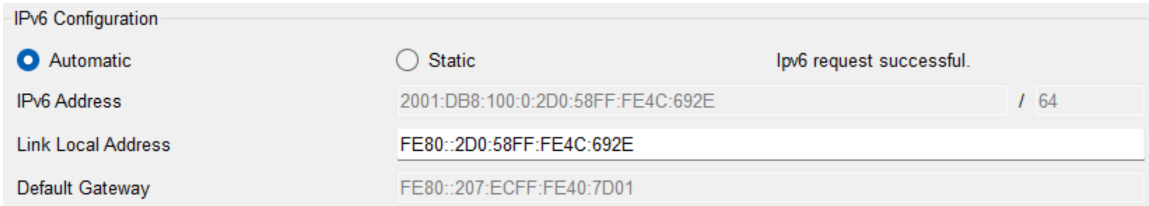
Router2

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#int gig 0/1
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/1, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/1, changed state to up

Router(config-if)#ipv6 d
Router(config-if)#ipv6 dhcp c
Router(config-if)#ipv6 dhcp client pd mgk22
Router(config-if)#ipv6 add aut
Router(config-if)#ipv6 add autoconfig
Router(config-if)#ipv6 en
Router(config-if)#ipv6 enable
Router(config-if)#exit
Router(config)#ipv6 u
Router(config)#ipv6 unicast-routing
Router(config)#int gig0/0
Router(config-if)#no shut

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0, changed state to up

%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/0, changed state to up

Router(config-if)#ipv6 add mgk22 ::2:0:0:2/64
% Warning: Bits overlapping with general prefix mgk22 will be truncated.
Router(config-if)#end
Router#
Router#sho ipv6 int bri
GigabitEthernet0/0         [up/up]
    FE80::2E0:B0FF:FEDE:8901
    2001:DB8:200:0:2::2
GigabitEthernet0/1         [up/up]
    FE80::2E0:B0FF:FEDE:8902
    2001:DB:300:400:2E0:B0FF:FEDE:8902
Vlan1                      [administratively down/down]
```

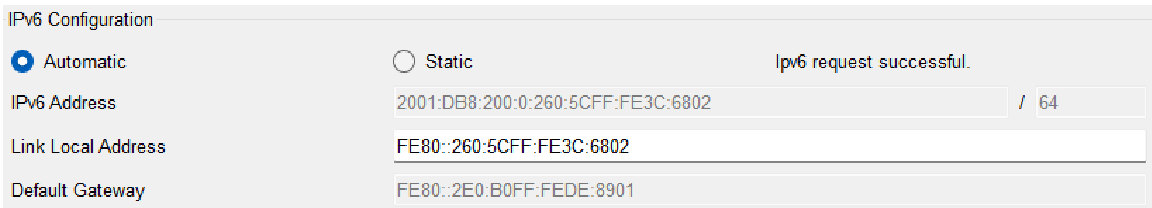
PC0



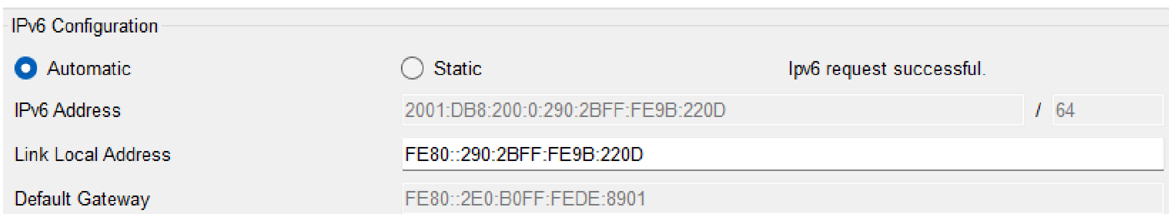
PC1



PC2



PC3

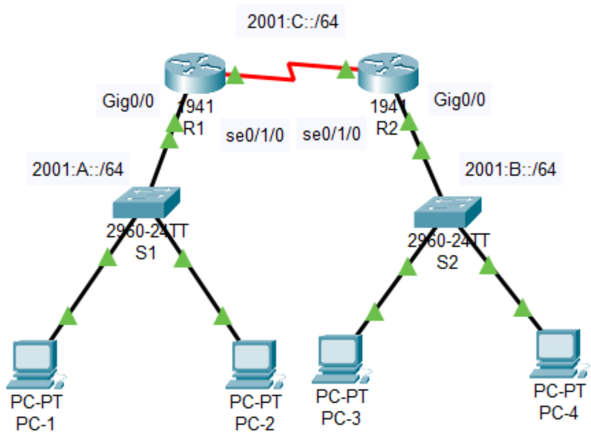


Routing Protocols

RIP Routing

RIP (Routing Information Protocol) is a distance-vector routing protocol that uses hop count as its metric. It is simple to configure but limited to a maximum of 15 hops, making it more suitable for small networks. RIP periodically exchanges full routing tables with neighbors every 30 seconds.

Cisco Packet Tracer:



Router1

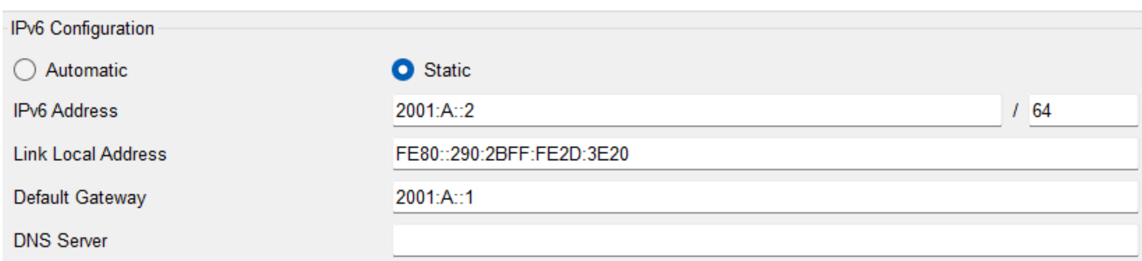
```
ipv6 unicast-routing
!
!
interface GigabitEthernet0/0
 no ip address
 duplex auto
 speed auto
 ipv6 address 2001:A::1/64
 ipv6 enable
!
interface GigabitEthernet0/1
 no ip address
 duplex auto
 speed auto
 shutdown
!
interface Serial0/1/0
 no ip address
 ipv6 address 2001:C::1/64
 ipv6 enable
 clock rate 2000000
!
interface Serial0/1/1
 no ip address
 clock rate 2000000
 shutdown
!
```

Router2

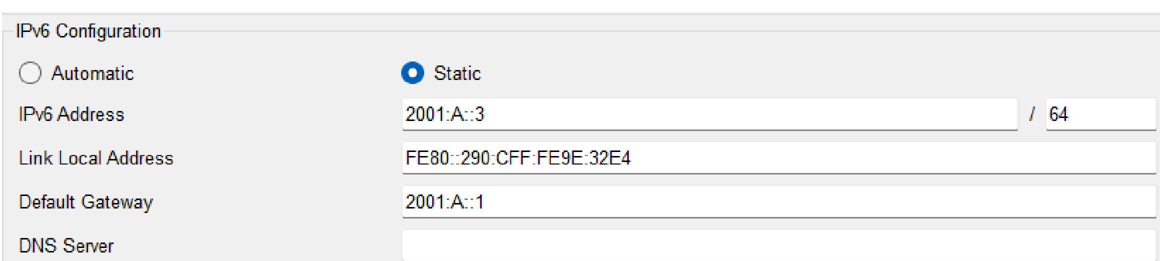
```
ipv6 unicast-routing
!
```

```
!
interface GigabitEthernet0/0
 no ip address
 duplex auto
 speed auto
 ipv6 address 2001:B::1/64
 ipv6 enable
!
interface GigabitEthernet0/1
 no ip address
 duplex auto
 speed auto
 shutdown
!
interface Serial0/1/0
 no ip address
 ipv6 address 2001:C::2/64
 ipv6 enable
!
interface Serial0/1/1
 no ip address
 clock rate 2000000
 shutdown
!
```

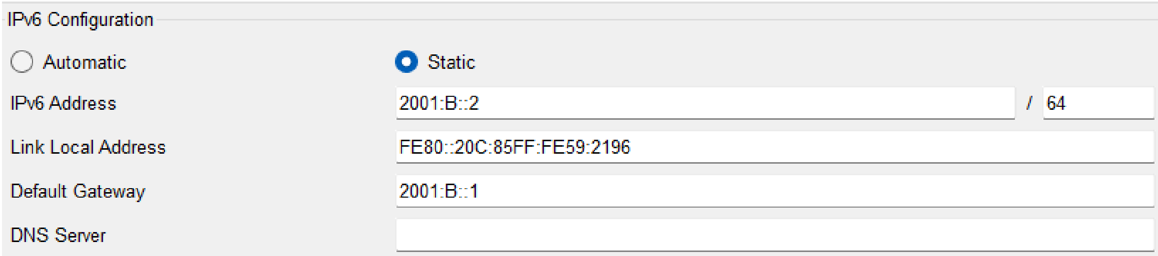
PC1



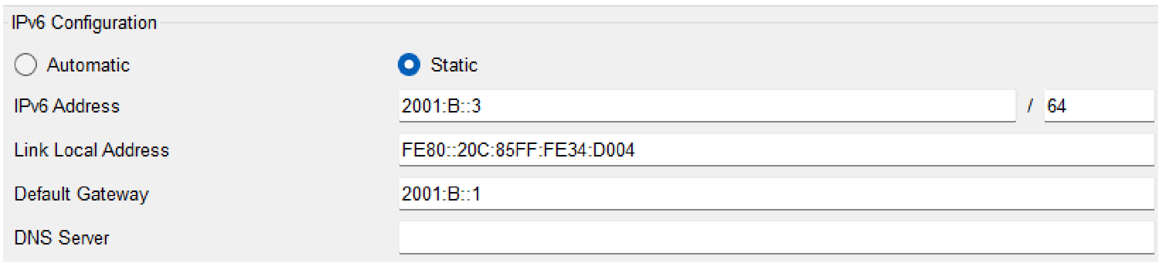
PC2



PC3



PC4



RIP Routing (R1)

```
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 router rip mgk
Router(config-rtr)#int se0/1/0
Router(config-if)#ipv6 rip mgk ena
Router(config-if)#int gig0/0
Router(config-if)#ipv6 rip mgk ena
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

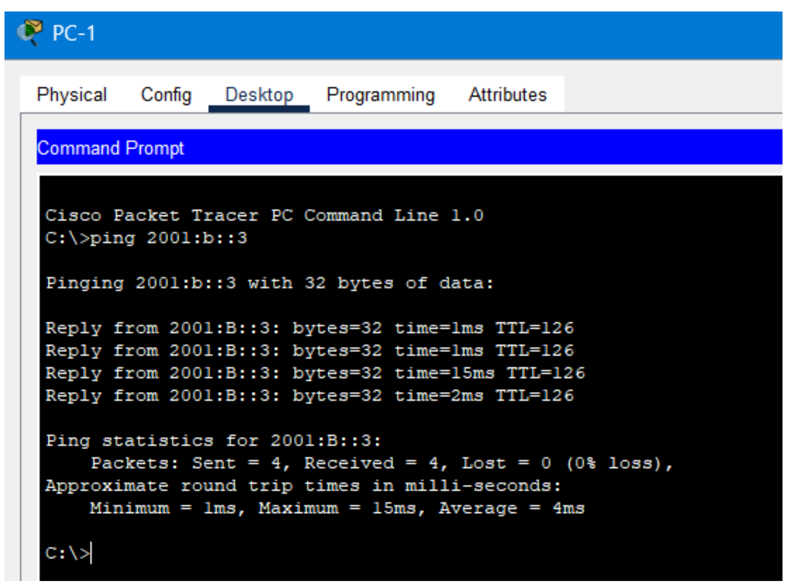
```
Router#show ipv6 rou
IPv6 Routing Table - 6 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr - Redirect
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
C   2001:A::/64 [0/0]
     via GigabitEthernet0/0, directly connected
L   2001:A::1/128 [0/0]
     via GigabitEthernet0/0, receive
R   2001:B::/64 [120/2]
     via FE80::240:BFF:FE6B:3501, Serial0/1/0
C   2001:C::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:C::1/128 [0/0]
     via Serial0/1/0, receive
L   FF00::/8 [0/0]
     via Null0, receive
```

RIP Routing (R2)

```
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 router rip mgk
Router(config-rtr)#int se0/1/0
Router(config-if)#ipv6 rip mgk enable
Router(config-if)#int gig0/0
Router(config-if)#ipv6 rip mgk enable
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

```
Router#show ipv6 route
IPv6 Routing Table - 6 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr - Redirect
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
R   2001:A::/64 [120/2]
     via FE80::203:E4FF:FE7E:2001, Serial0/1/0
C   2001:B::/64 [0/0]
     via GigabitEthernet0/0, directly connected
L   2001:B::1/128 [0/0]
     via GigabitEthernet0/0, receive
C   2001:C::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:C::2/128 [0/0]
     via Serial0/1/0, receive
L   FF00::/8 [0/0]
     via Null0, receive
```

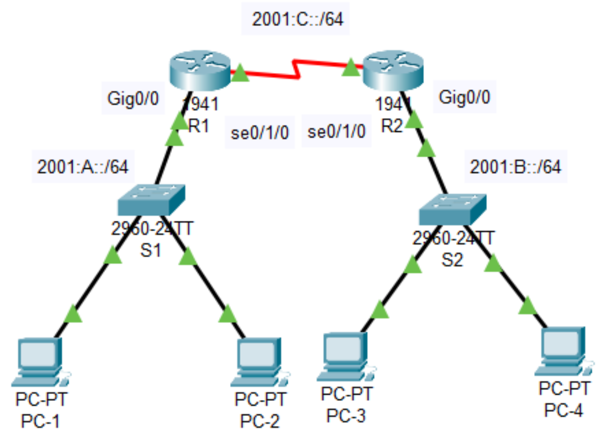
Pinging from PC1 to PC4



EIGRP Routing

EIGRP (Enhanced Interior Gateway Routing Protocol) is an advanced distance-vector protocol developed by Cisco. It uses the Diffusing Update Algorithm (DUAL) to find the best path and supports unequal cost load balancing. EIGRP is faster and more efficient than RIP, making it suitable for medium to large networks.

Cisco Packet Tracer:



!!! Same IP Addressing As In RIP Routing !!!

EIGRP Routing (R1)

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 router eigrp 10
Router(config-rtr)#eigrp r
Router(config-rtr)#eigrp router-id 1.1.1.1
Router(config-rtr)#no shut
Router(config-rtr)#int se0/1/0
Router(config-if)#ipv6 eigrp 10
Router(config-if)#int gig0/0
Router(config-if)#ipv6 eigrp 10
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

%DUAL-5-NBRCHANGE: IPv6-EIGRP 10: Neighbor FE80::2D0:D3FF:FE0C:D295 (Serial0/1/0) is up: new
adjacency
```

```
Router#show ipv6 ro
IPv6 Routing Table - 6 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr - Redirect
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
C   2001:A::/64 [0/0]
     via GigabitEthernet0/0, directly connected
L   2001:A::1/128 [0/0]
     via GigabitEthernet0/0, receive
D   2001:B::/64 [90/2172416]
     via FE80::2D0:D3FF:FE0C:D295, Serial0/1/0
C   2001:C::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:C::1/128 [0/0]
     via Serial0/1/0, receive
L   FF00::/8 [0/0]
     via Null0, receive
```

EIGRP Routing (R2)

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 router eigrp 10
Router(config-rtr)#eigrp r
Router(config-rtr)#eigrp router-id 2.2.2.2
Router(config-rtr)#no  shut
Router(config-rtr)#int se0/1/0
Router(config-if)#ipv6 eigrp 10
Router(config-if)#
%DUAL-5-NBRCHANGE: IPv6-EIGRP 10: Neighbor FE80::260:2FFF:FE6D:3912 (Serial0/1/0) is up: new
adjacency
ipv6 eigrp 10
Router(config-if)#int gig0/0
Router(config-if)#ipv6 eigrp 10
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console
```

```
Router#show ipv6 ro
IPv6 Routing Table - 6 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr - Redirect
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
D   2001:A::/64 [90/2172416]
     via FE80::260:2FFF:FE6D:3912, Serial0/1/0
C   2001:B::/64 [0/0]
     via GigabitEthernet0/0, directly connected
L   2001:B::1/128 [0/0]
     via GigabitEthernet0/0, receive
C   2001:C::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:C::2/128 [0/0]
     via Serial0/1/0, receive
L   FF00::/8 [0/0]
     via Null0, receive
```

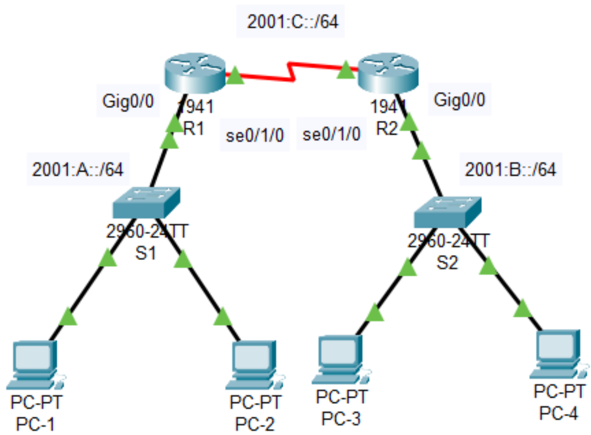
Pinging from PC1 to PC4

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 2001:b::3

Pinging 2001:b::3 with 32 bytes of data:

Reply from 2001:B::3: bytes=32 time=2ms TTL=126
Reply from 2001:B::3: bytes=32 time=2ms TTL=126
Reply from 2001:B::3: bytes=32 time=2ms TTL=126
Reply from 2001:B::3: bytes=32 time=15ms TTL=126

Ping statistics for 2001:B::3:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 15ms, Average = 5ms

C:\>
```

OSPF Routing

OSPF (Open Shortest Path First) is a link-state routing protocol that uses the Dijkstra Shortest Path First (SPF) algorithm to calculate the best path. It divides networks into areas to improve scalability and reduces routing overhead. OSPF converges quickly and is widely used in enterprise networks.

Cisco Packet Tracer:



!!! Same IPv6 Addressing As In RIP Routing !!!

OSPF Routing (R1)

```
Router#
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 router ospf 10
Router(config-rtr)#r
Router(config-rtr)#ro
Router(config-rtr)#router-id 1.1.1.1
Router(config-rtr)#int se0/1/0
Router(config-if)#ipv6 ospf 10 area 0
Router(config-if)#int gig0/0
Router(config-if)#ipv6 ospf 10 area 0
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

00:08:48: %OSPFv3-5-ADJCHG: Process 10, Nbr 2.2.2.2 on Serial0/1/0 from LOADING to FULL, Loading
Done
```

```
Router#show ipv6 ro
IPv6 Routing Table - 6 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr - Redirect
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
C   2001:A::/64 [0/0]
     via GigabitEthernet0/0, directly connected
L   2001:A::1/128 [0/0]
     via GigabitEthernet0/0, receive
O   2001:B::/64 [110/65]
     via FE80::260:70FF:FE2A:61, Serial0/1/0
C   2001:C::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:C::1/128 [0/0]
     via Serial0/1/0, receive
L   FF00::/8 [0/0]
     via Null0, receive
```

OSPF Routing (R2)

```
Router>en
Router#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#ipv6 router ospf 10
%OSPFv3-4-NORTRID: OSPFv3 process 10 could not pick a router-id,please configure manually
Router(config-rtr)#ro
Router(config-rtr)#router-id 2.2.2.2
Router(config-rtr)#int se0/1/0
Router(config-if)#ipv6 ospf 10 area 0
Router(config-if)#int gig0/0
Router(config-if)#ipv6 ospf 10 area 0
Router(config-if)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

00:08:48: %OSPFv3-5-ADJCHG: Process 10, Nbr 1.1.1.1 on Serial0/1/0 from LOADING to FULL, Loading
Done
```

```
Router#show ipv6 ro
IPv6 Routing Table - 6 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route, M - MIPv6
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       ND - ND Default, NDp - ND Prefix, DCE - Destination, NDr - Redirect
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       D - EIGRP, EX - EIGRP external
O   2001:A::/64 [110/65]
     via FE80::201:42FF:FE55:9A51, Serial0/1/0
C   2001:B::/64 [0/0]
     via GigabitEthernet0/0, directly connected
L   2001:B::1/128 [0/0]
     via GigabitEthernet0/0, receive
C   2001:C::/64 [0/0]
     via Serial0/1/0, directly connected
L   2001:C::2/128 [0/0]
     via Serial0/1/0, receive
L   FF00::/8 [0/0]
     via Null0, receive
```

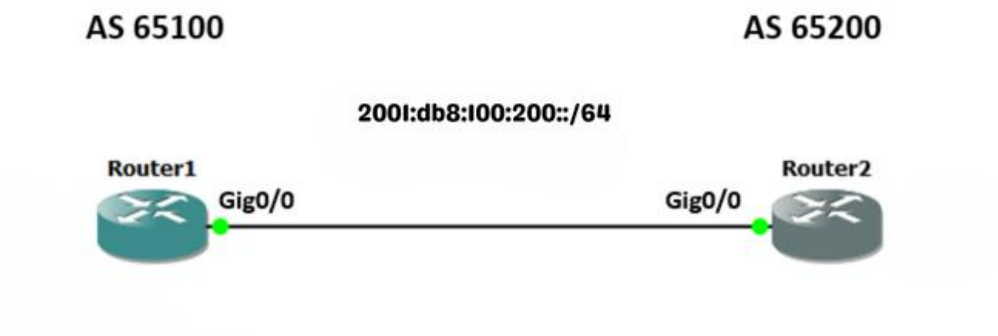
Pinging from PC1 to PC3

```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 2001:b::2

Pinging 2001:b::2 with 32 bytes of data:

Reply from 2001:B::2: bytes=32 time=2ms TTL=126
Reply from 2001:B::2: bytes=32 time=24ms TTL=126
Reply from 2001:B::2: bytes=32 time=22ms TTL=126
Reply from 2001:B::2: bytes=32 time=13ms TTL=126

Ping statistics for 2001:B::2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 2ms, Maximum = 24ms, Average = 15ms

C:\>
```

BGP Routing

Border Gateway Protocol (BGP) for IPv6, also called **BGP4+**, is the extension of BGP that supports IPv6 addressing and routing between autonomous systems (AS). Just like IPv4 BGP, it is used for exchanging routing information between different ISPs or large networks, ensuring scalability and policy-based routing. IPv6 BGP uses the same underlying mechanisms—such as neighbor establishment, path attributes, and route selection—but advertises IPv6 prefixes using Multiprotocol BGP (MP-BGP) extensions. It enables global IPv6 connectivity and efficient inter-domain routing by exchanging reachability information for IPv6 networks.

However, **Cisco Packet Tracer does not support IPv6 BGP routing**, therefore we will only discuss the configurations here.



Router1:
Ipv6 unicast-routing
Int Gig0/0
Ipv6 add 2001:db8:100:200::1/64
No shutdown
Router bgp 65100
Bgp router-id 1.1.1.1
Neighbor 2001:db8:100:200::2 remote-as 65200
No address-family ipv4 unicast
Address-family ipv6 unicast
Neighbor 2001:db8:100:200::2 activate
Show bgp ipv6 unicast summary
Ping 2001:db8:100:200::2 (Successful)

Router2:
Ipv6 unicast-routing
Int Gig0/0
Ipv6 add 2001:db8:100:200::2/64
No shutdown
Router bgp 65200
Bgp router-id 2.2.2.2
Neighbor 2001:db8:100:200::1 remote-as 65100
No address-family ipv4 unicast
Address-family ipv6 unicast
Neighbor 2001:db8:100:200::1 activate
Show bgp ipv6 unicast summary
Ping 2001:db8:100:200::1 (Successful)

If we connect PCs to the routers through switches, we only need to configure an additional LAN interface on each router, assign an IPv6 subnet to it, and advertise that network into BGP. For example:

- **Router1 (AS 65100):** LAN subnet 2001:db8:100:1::/64, interface Gig0/1 → 2001:db8:100:1::1/64, advertised in BGP with network 2001:db8:100:1::/64.
- **Router2 (AS 65200):** LAN subnet 2001:db8:100:2::/64, interface Gig0/1 → 2001:db8:100:2::1/64, advertised in BGP with network 2001:db8:100:2::/64.
- **PCs:** Assigned IPv6 addresses from the respective LAN subnets (e.g., 2001:db8:100:1::10/64 with gateway 2001:db8:100:1::1 on Router1 side, and 2001:db8:100:2::10/64 with gateway 2001:db8:100:2::1 on Router2 side).

This way, the LAN networks are learned via BGP on both routers, allowing PCs behind Router1 to communicate with PCs behind Router2 using IPv6.

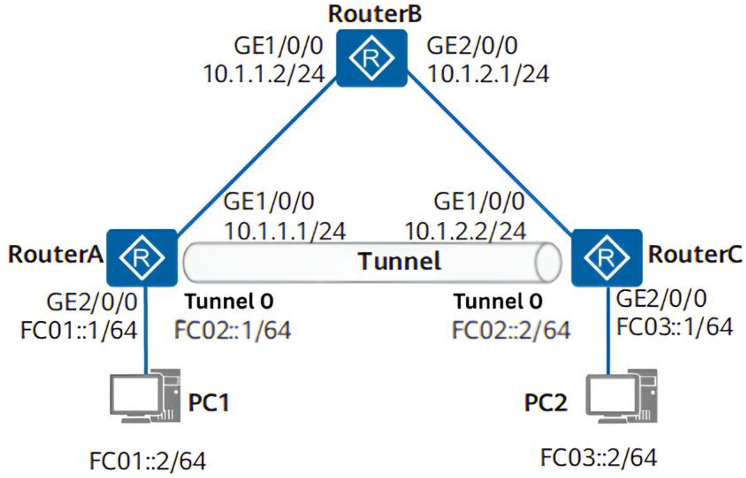
Transition Mechanisms

Transition mechanisms are methods that allow IPv4 and IPv6 networks to communicate with each other during the gradual shift from IPv4 to IPv6. Since IPv6 is not backward compatible with IPv4, these mechanisms provide smooth interoperability. The main categories include **dual-stack**, **translation**, and **tunneling**. Tunneling is especially important because it encapsulates IPv6 traffic inside IPv4 packets (or vice versa), enabling communication across incompatible networks.

GRE

(Generic Routing Encapsulation)

GRE is a tunneling protocol developed by Cisco that can encapsulate many types of network layer protocols inside virtual point-to-point links. For IPv6 transition, GRE can encapsulate IPv6 packets inside IPv4, allowing them to traverse IPv4-only networks. It is simple, flexible, and supports multiprotocol transport, but does not provide encryption by itself.



Configuration:

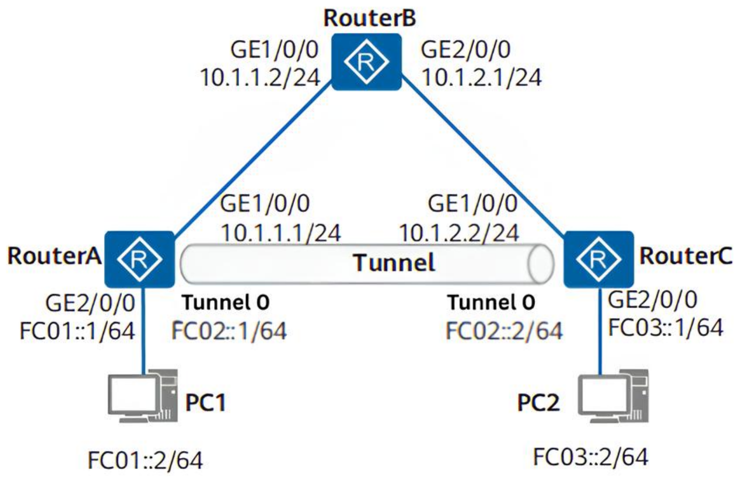
hostname RouterA ! ipv6 unicast-routing ! interface GigabitEthernet1/0/0 ip address 10.1.1.1 255.255.255.0 no shutdown ! interface GigabitEthernet2/0/0 ipv6 address FC01::1/64 no shutdown ! interface Tunnel0 ipv6 address FC02::1/64 tunnel mode gre ipv6 ipv6 tunnel source 10.1.1.1 tunnel destination 10.1.2.2 no shutdown ! ip route 10.1.2.0 255.255.255.0 10.1.1.2 ipv6 route FC03::/64 Tunnel0 ! end	hostname RouterB ! interface GigabitEthernet1/0/0 ip address 10.1.1.2 255.255.255.0 no shutdown ! interface GigabitEthernet2/0/0 ip address 10.1.2.1 255.255.255.0 no shutdown ! end
hostname RouterC !	PC1 and PC2 can successfully ping each other.

ipv6 unicast-routing ! interface GigabitEthernet1/0/0 ip address 10.1.2.2 255.255.255.0 no shutdown ! interface GigabitEthernet2/0/0 ipv6 address FC03::1/64 no shutdown ! interface Tunnel0 ipv6 address FC02::2/64 tunnel mode gre ipv6 ipv6 tunnel source 10.1.2.2 tunnel destination 10.1.1.1 no shutdown ! ip route 10.1.1.0 255.255.255.0 10.1.2.1 ipv6 route FC01::/64 Tunnel0 ! end	

IPv6IP tunnels

(Manual Tunnels)

IPv6-over-IPv4 tunnels are manually configured point-to-point links where IPv6 packets are encapsulated within IPv4 headers. This type of tunnel is often used when two IPv6 networks need to connect across an IPv4-only backbone. Configuration requires specifying both tunnel endpoints (source and destination), making it reliable but less scalable in large networks.

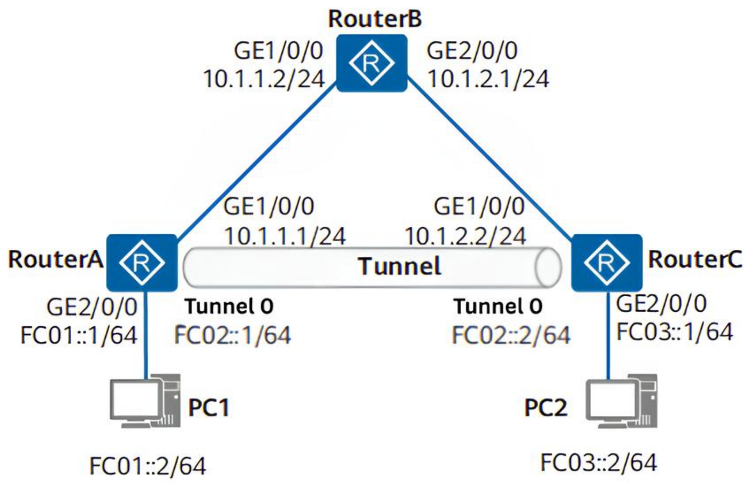


When using **tunnel mode gre ipv6**, an additional GRE header is added on top of IPv4, which reduces the effective MTU from 1500 to **1476 bytes**. If we switch the configuration to **tunnel mode ipv6ip**, only the IPv4 header is added without GRE overhead, so the MTU increases to **1480 bytes**. Functionally, GRE allows multiple protocols (IPv4, IPv6, multicast, routing updates), while IPv6IP only supports IPv6 traffic but with lower overhead.

- tunnel mode gre ipv6 → MTU = **1476** (IPv4 + GRE headers).
- tunnel mode ipv6ip → MTU = **1480** (only IPv4 header).
- GRE = more flexible, higher overhead.
- IPv6IP = only IPv6, simpler, more efficient.

6to4 tunnels

6to4 is an automatic tunneling mechanism that allows IPv6 packets to be transmitted over an IPv4 infrastructure without manual configuration of tunnel endpoints. It uses a special IPv6 prefix (2002::/16) where the IPv4 address of the gateway is embedded inside the IPv6 address. This makes it easier for IPv6 networks to connect over the IPv4 internet, but it depends heavily on relay routers and can face reliability issues.



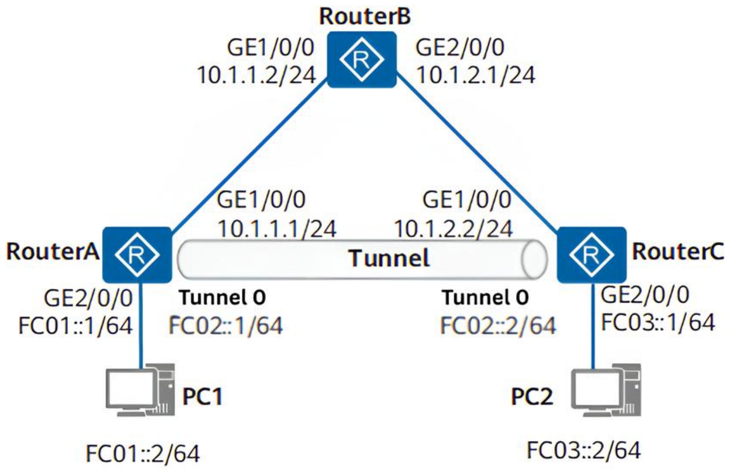
Configuration:

hostname RouterA ipv6 unicast-routing ! interface GigabitEthernet1/0/0 ip address 10.1.1.1 255.255.255.0 no shutdown ! interface GigabitEthernet2/0/0 ipv6 address FC01::1/64 no shutdown ! interface Tunnel0 ipv6 address 2002:0A01:0101::1/64 tunnel mode ipv6ip 6to4 tunnel source 10.1.1.1 no shutdown ! ip route 10.1.2.0 255.255.255.0 10.1.1.2 ipv6 route 2002::/16 Tunnel0 ipv6 route FC03::/64 Tunnel0 ! end	hostname RouterB ! interface GigabitEthernet1/0/0 ip address 10.1.1.2 255.255.255.0 no shutdown ! interface GigabitEthernet2/0/0 ip address 10.1.2.1 255.255.255.0 no shutdown end
hostname RouterC ipv6 unicast-routing !	PC1 and PC2 can successfully ping each other.

interface GigabitEthernet1/0/0 ip address 10.1.2.2 255.255.255.0 no shutdown ! interface GigabitEthernet2/0/0 ipv6 address FC03::1/64 no shutdown ! interface Tunnel0 ipv6 address 2002:0A01:0202::1/64 tunnel mode ipv6ip 6to4 tunnel source 10.1.2.2 no shutdown ! ip route 10.1.1.0 255.255.255.0 10.1.2.1 ipv6 route 2002::/16 Tunnel0 ipv6 route FC01::/64 Tunnel0 ! end	

ISATAP tunnels

ISATAP is designed for communication within an organization (intra-site). It treats the IPv4 network as a virtual IPv6 link layer and automatically assigns IPv6 addresses to hosts by embedding their IPv4 address into the IPv6 address. This is useful for enterprises gradually transitioning to IPv6 without changing their internal IPv4 infrastructure.



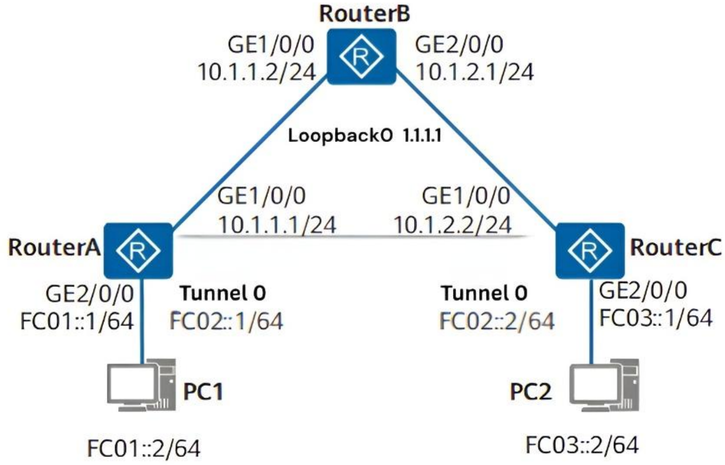
Configuration:

hostname RouterA ipv6 unicast-routing ! interface GigabitEthernet1/0/0 ip address 10.1.1.1 255.255.255.0 no shutdown ! interface GigabitEthernet2/0/0 ipv6 address FC01::1/64 no shutdown ! interface Tunnel0 ipv6 address 2001:db8:100:200::/64 eui-64 tunnel mode ipv6ip tunnel source 10.1.1.1 no shutdown ! ip route 10.1.2.0 255.255.255.0 10.1.1.2 ipv6 route FC03::/64 Tunnel0 ! end	hostname RouterB ! interface GigabitEthernet1/0/0 ip address 10.1.1.2 255.255.255.0 no shutdown ! interface GigabitEthernet2/0/0 ip address 10.1.2.1 255.255.255.0 no shutdown end
hostname RouterC ipv6 unicast-routing ! interface GigabitEthernet1/0/0 ip address 10.1.2.2 255.255.255.0 no shutdown ! interface GigabitEthernet2/0/0 ipv6 address FC03::1/64 no shutdown ! interface Tunnel0 ipv6 address 2001:db8:100:200::/64 eui-64	PC1 and PC2 can successfully ping each other.

tunnel mode ipv6ip tunnel source 10.1.2.2 no shutdown ! ip route 10.1.1.0 255.255.255.0 10.1.2.1 ipv6 route FC01::/64 Tunnel0 ! end	

GRE Multipoint (DMVPN)

Dynamic Multipoint VPN (DMVPN) builds on GRE by allowing multipoint tunnels instead of simple point-to-point links. In IPv6 transition, DMVPN allows IPv6 traffic to be encapsulated in IPv4 GRE tunnels dynamically, creating on-demand secure communication between multiple sites. It reduces the need for configuring multiple static tunnels and provides better scalability for large networks.



Configuration:

hostname RouterA ! ipv6 unicast-routing ! ipv6 router ospf 10 ! router ospf 1	hostname RouterB ! ipv6 unicast-routing ! ipv6 router ospf 10 ! router ospf 1

network 10.1.1.0 0.0.0.255 area 0 ! interface GigabitEthernet1/0/0 ip address 10.1.1.1 255.255.255.0 ipv6 ospf 10 area 0 no shutdown ! interface GigabitEthernet2/0/0 ipv6 address FC01::1/64 ipv6 ospf 10 area 0 no shutdown ! interface Tunnel0 ipv6 address FE80::1 link-local ipv6 address 2001:db8:100:200::1/64 tunnel source GigabitEthernet1/0/0 tunnel destination1.1.1.1 ipv6 nhrp network-id 1 ipv6 nhrp map multicast 1.1.1.1 ipv6 nhrp nhs 2001:db8:100:200::2 ipv6 nhrp map 2001:db8:100:200::2 1.1.1.1 ipv6 ospf 10 area 0 ipv6 ospf network broadcast ipv6 ospf priority 0 no shutdown ! end	network 1.1.1.1 0.0.0.0 area 0 network 10.1.1.0 0.0.0.255 area 0 network 10.1.2.0 0.0.0.255 area 0 ! interface Loopback0 ip address 1.1.1.1 255.255.255.255 no shutdown ! interface GigabitEthernet1/0/0 ip address 10.1.1.2 255.255.255.0 ipv6 ospf 10 area 0 no shutdown ! interface GigabitEthernet2/0/0 ip address 10.1.2.1 255.255.255.0 ipv6 ospf 10 area 0 no shutdown ! interface Tunnel0 ipv6 address FE80::2 link-local ipv6 address 2001:db8:100:200::2/64 tunnel source Loopback0 tunnel mode gre multipoint ipv6 nhrp network-id 1 ipv6 nhrp map multicast dynamic ipv6 ospf 10 area 0 ipv6 ospf network broadcast no shutdown ! end
hostname RouterC ! ipv6 unicast-routing ! ipv6 router ospf 10 ! router ospf 1 network 10.1.2.0 0.0.0.255 area 0 ! interface GigabitEthernet1/0/0	PC1 and PC2 can successfully ping each other.

ip address 10.1.2.2 255.255.255.0 ipv6 ospf 10 area 0 no shutdown ! interface GigabitEthernet2/0/0 ipv6 address FC03::1/64 ipv6 ospf 10 area 0 no shutdown ! interface Tunnel0 ipv6 address FE80::3 link-local ipv6 address 2001:db8:100:200::3/64 tunnel source GigabitEthernet1/0/0 tunnel destination 1.1.1.1 ipv6 nhrp network-id 1 ipv6 nhrp map multicast 1.1.1.1 ipv6 nhrp nhs 2001:db8:100:200::2 ipv6 nhrp map 2001:db8:100:200::2 1.1.1.1 ipv6 ospf 10 area 0 ipv6 ospf network broadcast ipv6 ospf priority 0 no shutdown ! end	

Phase 1:
Spokes (A & C) have a tunnel destination pointing to the hub (B).
All traffic flows Spoke → Hub → Spoke.
Direct spoke-to-spoke tunnels are not possible.
The hub has mGRE enabled and keeps dynamic NHRP mappings.
Routing: Hub is the central point.

Phase 2:
Goal: Allow spokes to form direct tunnels without passing through the hub for data traffic.
On spokes (A & C):
Remove the static tunnel destination.
Enable mGRE on spokes.

Configuration on RouterA & RouterC:

```
interface Tunnel0
shutdown
no tunnel destination 1.1.1.1
tunnel mode gre multipoint
no shutdown
```

After this, spokes register with the hub dynamically.
Spokes learn each other's mappings via NHRP and can create direct tunnels.
Supports both Spoke→Hub→Spoke and Spoke→Spoke communication.

Phase 3:
Goal: Make spoke-to-spoke traffic more efficient by skipping the hub in data forwarding.
The hub still acts as the control plane server, but data flows directly between spokes.

Configuration on RouterB (Hub):

```
interface Tunnel0
shutdown
ipv6 nhrp redirect
no shutdown
```

Configuration on RouterA & RouterC (Spokes):

```
interface Tunnel0
shutdown
ipv6 nhrp shortcut
no shutdown
```

Now, spokes immediately form direct tunnels after the first packet.
Hub is only used for signaling, not for data forwarding.

Author

Muhammad Gulraiz Khan
muhammadgulraiz047@gmail.com