

Transitioning to IPv6



Address Space Management

IPv4 and IPv6

IPv4:	4 octets
11000000.10101000.11001001.0111000	
192.168.201.113	
4,294,467,295 IP addresses	

- Currently, there are approximately 1.3 billion usable IPv4 addresses available.

IPv6:	16 octets
11010001.11011100.11001001.01110001.11010001.11011100. 11001100.01110001.11010001.11011100.11001001.01110001. 11010001.11011100.11001001.01110001	
A524:72D3:2C80:DD02:0029:EC7A:002B:EA73	
3.4 x 10 ³⁸ IP addresses	

Why Do We Need a Larger Address Space?

- Internet population
 - Approximately 973 million users in November 2005
 - Emerging population and geopolitical address space
- Mobile users
 - PDA, pen tablet, notepad, and so on
 - Approximately 20 million in 2004
- Mobile phones
 - Already 1 billion mobile phones delivered by the industry
- Transportation
 - 1 billion automobiles forecast for 2008
 - Internet access in planes, for example, Lufthansa
- Consumer devices
 - Sony mandated that all its products be IPv6-enabled by 2005
 - Billions of home and industrial appliances

IPv6 Advanced Features

Larger address space:

- Global reachability and flexibility
- Aggregation
- Multihoming
- Autoconfiguration
- Plug-and-play
- End-to-end without NAT
- Renumbering

Mobility and security:

- Mobile IP RFC-compliant
- IPsec mandatory (or native) for IPv6

Simpler header:

- Routing efficiency
- Performance and forwarding rate scalability
- No broadcasts
- No checksums
- Extension headers
- Flow labels

Transition richness:

- Dual stack
- 6to4 and manual tunnels
- Translation

IPv6 Address Representation

Format:

- x:x:x:x:x:x:x:x, where x is a 16-bit hexadecimal field
 - Case-insensitive for hexadecimal A, B, C, D, E, and F
- Leading zeros in a field are optional
- Successive fields of zeros can be represented as :: only once per address

Examples:

- 2031:0000:130F:0000:0000:09C0:876A:130B
 - Can be represented as 2031:0:130f::9c0:876a:130b
 - Cannot be represented as 2031::130f::9c0:876a:130b
- FF01:0:0:0:0:0:0:1 → FF01::1
- 0:0:0:0:0:0:0:1 → ::1
- 0:0:0:0:0:0:0:0 → ::

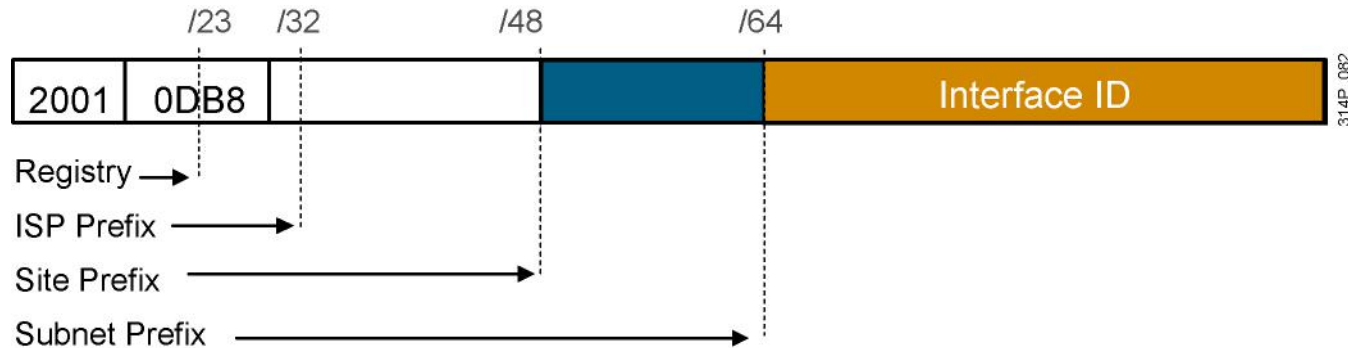
IPv6 Address Types

- Unicast:
 - Address is for a single interface
 - IPv6 has several types (for example, global, reserved, link-local, and site-local)
- Multicast:
 - One-to-many
 - Enables more efficient use of the network
 - Uses a larger address range
- Anycast:
 - One-to-nearest (allocated from unicast address space)
 - Multiple devices share the same address
 - All anycast nodes should provide uniform service
 - Source devices send packets to anycast address
 - Routers decide on closest device to reach that destination
 - Suitable for load balancing and content delivery services

IPv6 Unicast Addressing

- Types of IPv6 unicast addresses:
 - Global: Starts with 2000::/3 and assigned by IANA
 - Reserved: Used by the IETF
 - Private: Link local (starts with FE80::/10)
 - Loopback (::1)
 - Unspecified (::)
- A single interface may be assigned multiple IPv6 addresses of any type: unicast, anycast, or multicast.
- IPv6 addressing rules are covered by multiple RFCs.
 - Architecture defined by RFC 4291

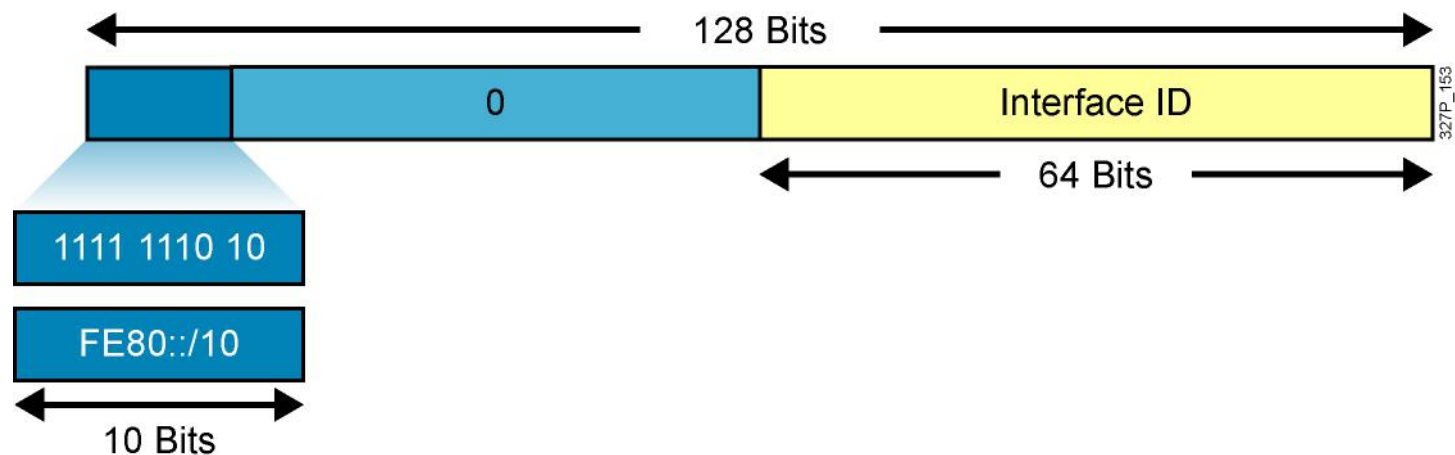
IPv6 Global Unicast (and Anycast) Addresses



IPv6 has the same address format for global unicast and for anycast addresses.

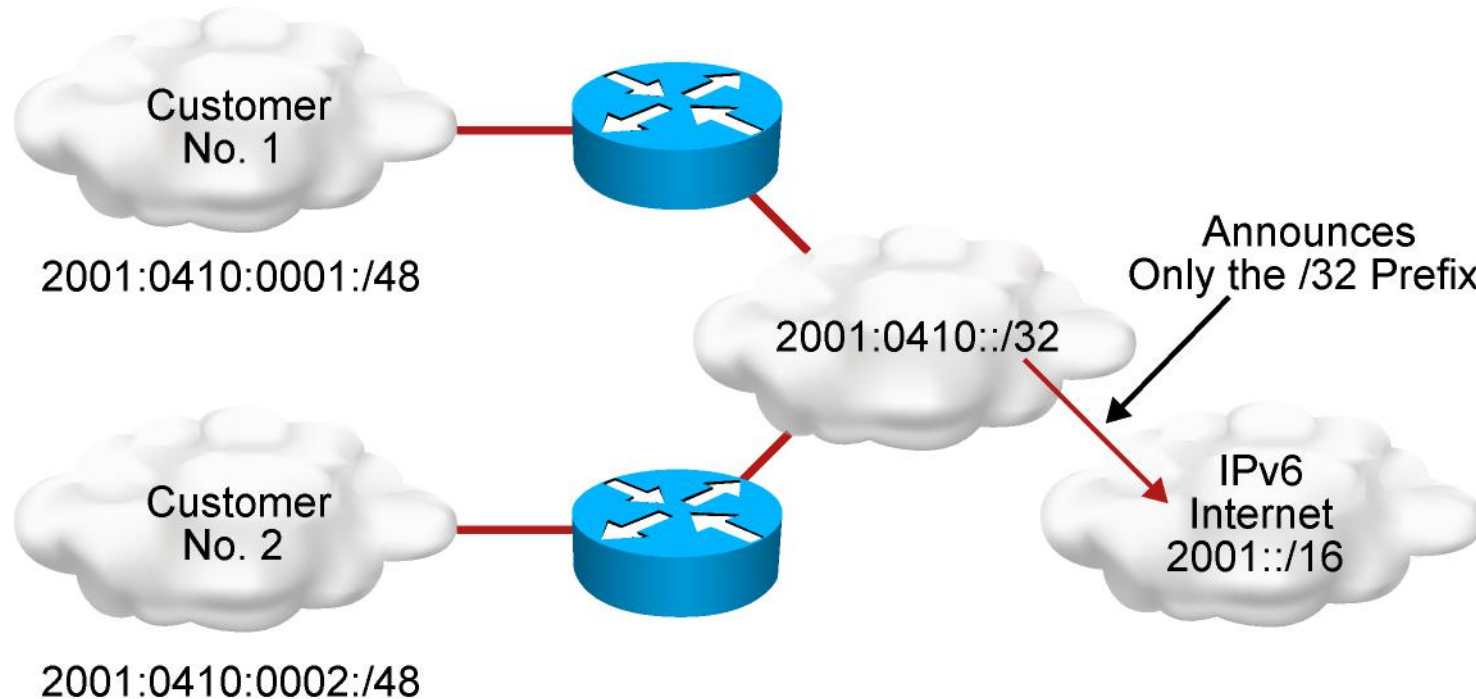
- Uses a global routing prefix—a structure that enables aggregation upward, eventually to the ISP.
- A single interface may be assigned multiple addresses of any type (unicast, anycast, multicast).
- Every IPv6-enabled interface contains at least one loopback (::1/128) and one link-local address.
- Optionally, every interface can have multiple unique local and global addresses.

Link-Local Addresses



- Link-local addresses have a scope limited to the link and are dynamically created on all IPv6 interfaces by using a specific link-local prefix `FE80::/10` and a 64-bit interface identifier.
- Link-local addresses are used for automatic address configuration, neighbor discovery, and router discovery. Link-local addresses are also used by many routing protocols.
- Link-local addresses can serve as a way to connect devices on the same local network without needing global addresses.
- When communicating with a link-local address, you must specify the outgoing interface because every interface is connected to `FE80::/10`.

Larger Address Space Enables Address Aggregation

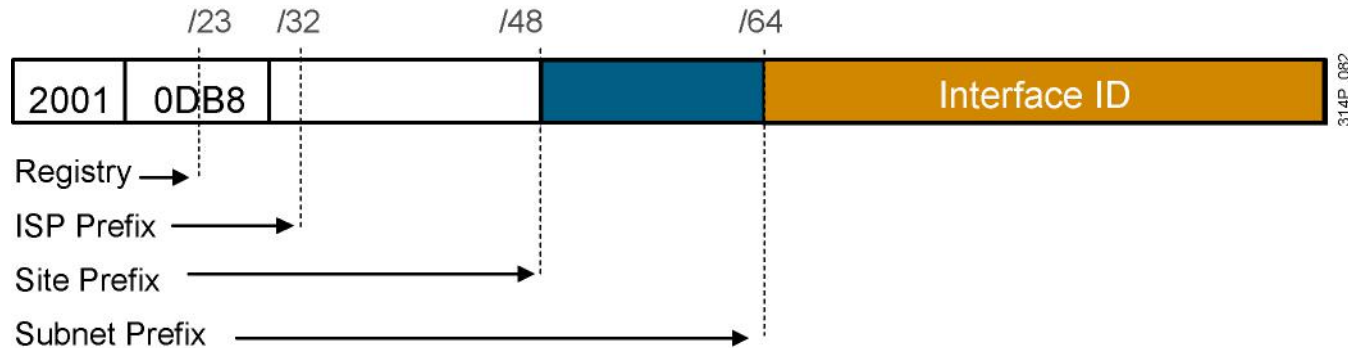


314P_078

Address aggregation provides the following benefits:

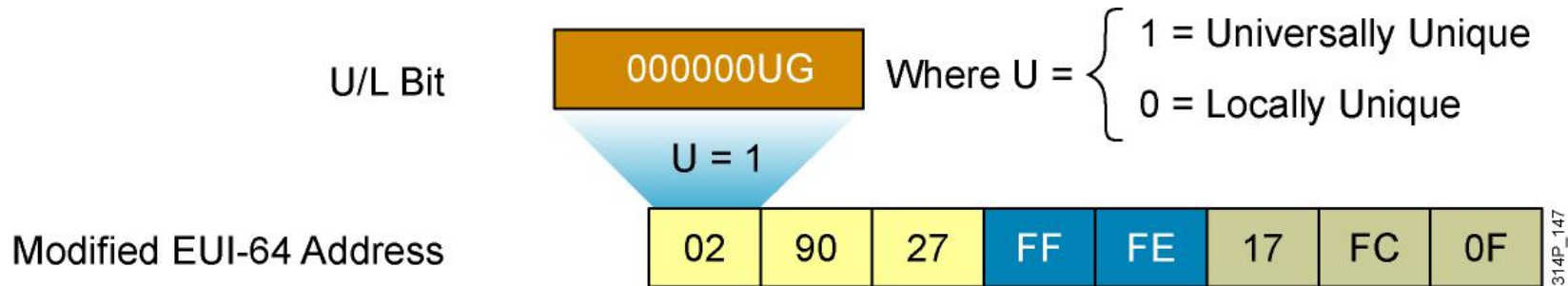
- Aggregation of prefixes announced in the global routing table
- Efficient and scalable routing
- Improved bandwidth and functionality for user traffic

Assigning IPv6 Global Unicast Addresses



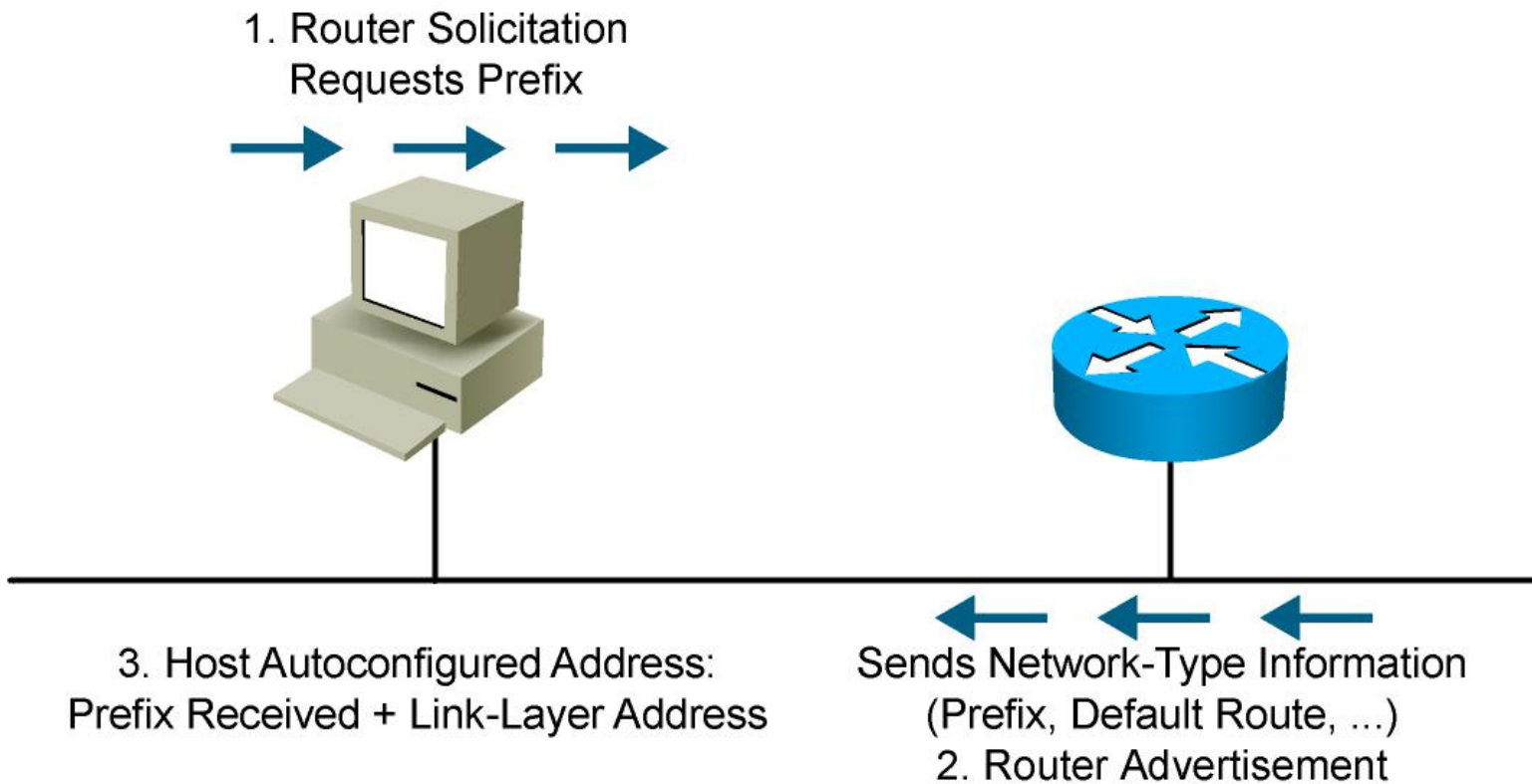
- Static assignment
 - Manual interface ID assignment
 - EUI-64 interface ID assignment
- Dynamic assignment
 - Stateless autoconfiguration
 - DHCPv6 (stateful)

IPv6 EUI-64 Interface Identifier



- Cisco can use the EUI-64 format for interface identifiers.
- This format expands the 48-bit MAC address to 64 bits by inserting “FFFE” into the middle 16 bits.
- To make sure that the chosen address is from a unique Ethernet MAC address, the U/L bit is set to 1 for global scope (0 for local scope).

Stateless Autoconfiguration



DHCPv6 (Stateful)

DHCPv6 is an updated version of DHCP for IPv4:

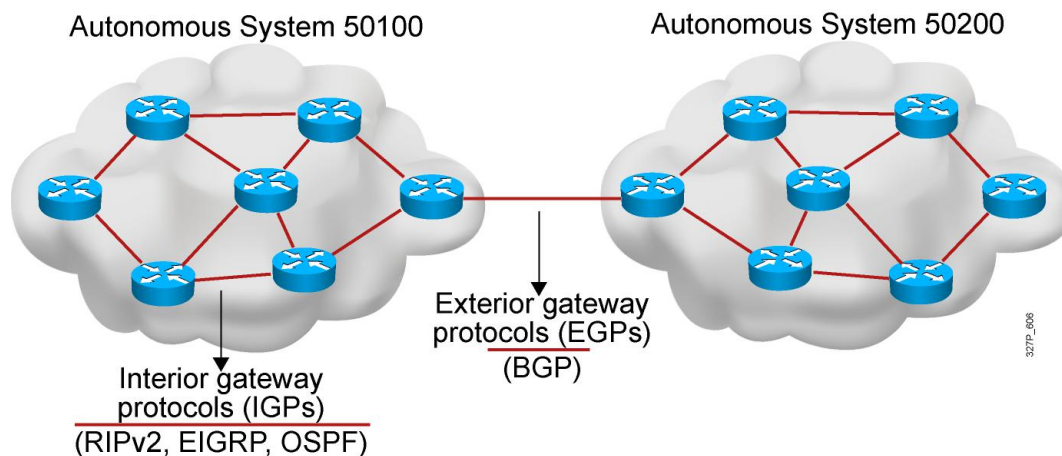
- Supports new addressing
- Enables more control than stateless autoconfiguration
- Can be used for renumbering
- Can be used for automatic domain name registration of hosts using dynamic DNS

DHCPv6 Operation

DHCPv6 operates in a way that is similar to DHCPv4, except:

- Client first detects the presence of routers on the link.
- If a router is found, the router advertisement is examined to determine if DHCP can be used.
- If no router is found, or if the router says DHCP can be used, then:
 - A DHCP solicit message is sent to the all-DHCP-agents multicast address.
 - The client uses the link-local address as the source address.

IPv6 Routing Protocols



- IPv6 routing types:
 - Static
 - RIPng (RFC 2080)
 - OSPFv3 (RFC 2740)
 - IS-IS for IPv6
 - MP-BGP4 (RFC 2545/2858)
 - EIGRP for IPv6
- The **ipv6 unicast-routing** command is required to enable IPv6 before any routing protocol is configured.

RIPng (RFC 2080)

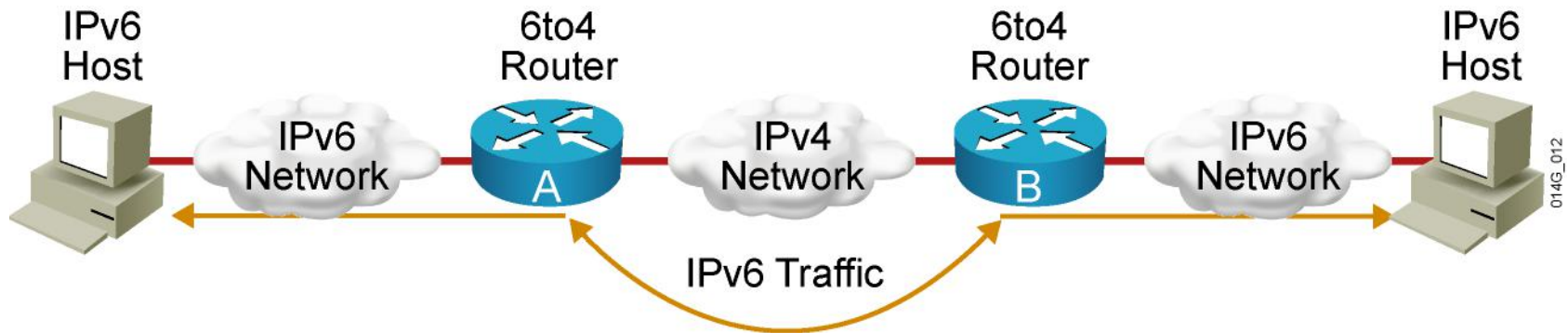
Similar IPv4 features:

- Distance vector, radius of 15 hops, split horizon, and poison reverse
- Based on RIPv2

Updated features for IPv6:

- IPv6 prefix, next-hop IPv6 address
- Uses the multicast group FF02::9, the all-rip-routers multicast group, as the destination address for RIP updates
- Uses IPv6 for transport
- Named RIPng

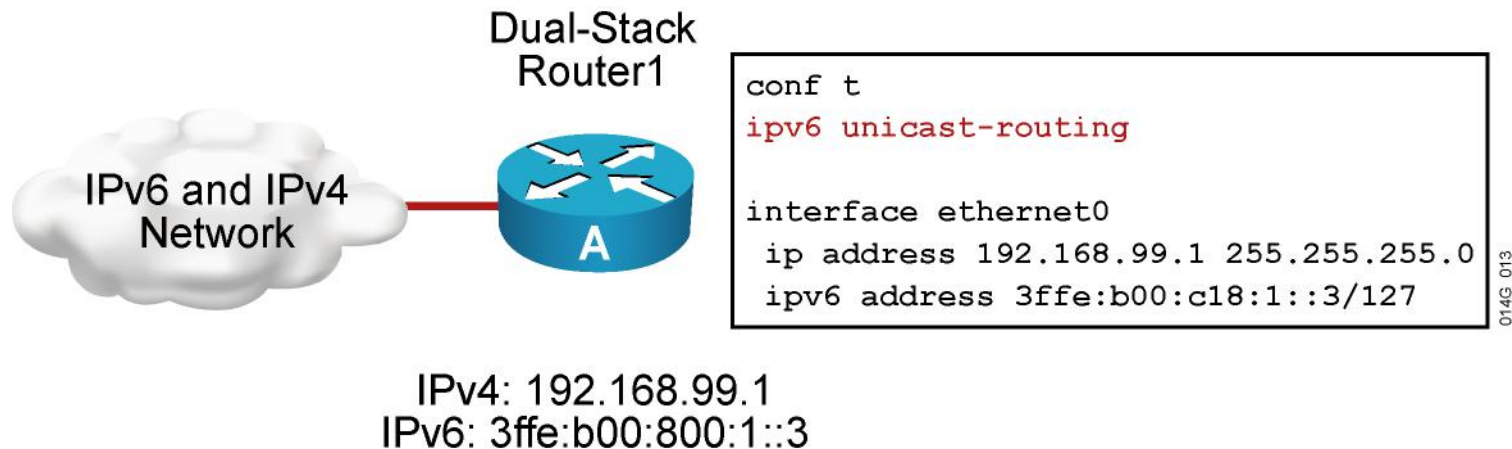
IPv4-to-IPv6 Transition



Transition richness means:

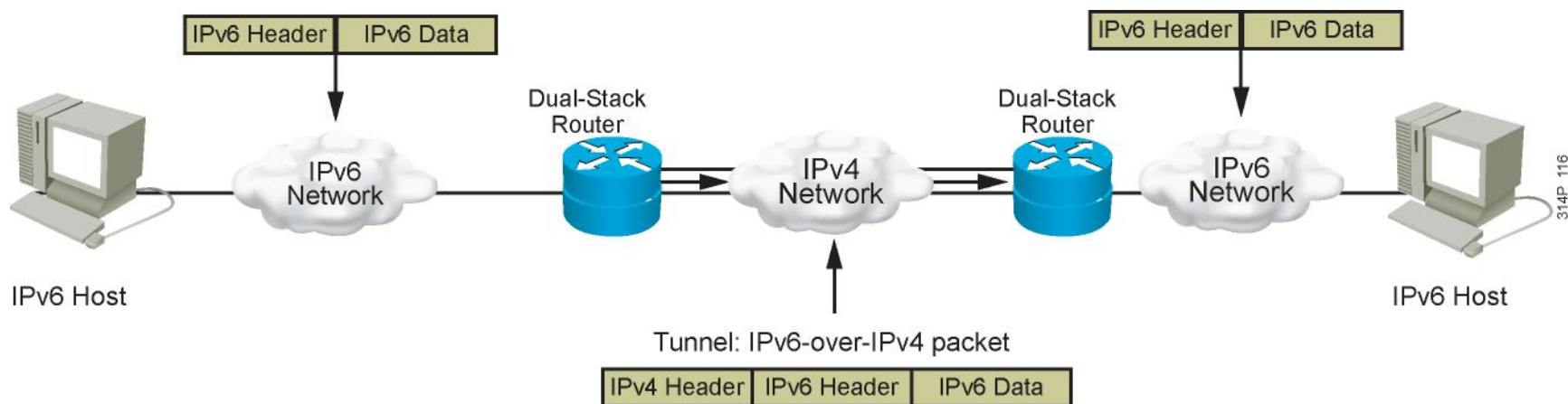
- No fixed day to convert; no need to convert all at once
- Different transition mechanisms are available:
 - Dual stack
 - Manual tunnel
 - 6to4 tunnel
 - ISATAP tunnel
 - Teredo tunnel
- Different compatibility mechanisms:
 - Proxying and translation (NAT-PT)

Cisco IOS Dual Stack (Cont.)



- When both IPv4 and IPv6 are configured on an interface, the interface is considered dual-stacked.

IPv6 Tunneling



Tunneling is an integration method in which an IPv6 packet is encapsulated within another protocol, such as IPv4. This method of encapsulation is IPv4.

- Includes a 20-byte IPv4 header with no options and an IPv6 header and payload
- Requires dual-stack routers

IPv6 Address Configuration Example

LAN: 2001:db8:c18:1::/64

Ethernet 0



```
ipv6 unicast-routing
interface Ethernet0
  ipv6 address 2001:db8:c18:1::/64 eui-64
```

MAC address: 0260.3e47.1530

```
RouterX# show ipv6 interface Ethernet0
Ethernet0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::260:3EFF:FE47:1530
Global unicast address(es):
  2001:DB8:C18:1:260:3EFF:FE47:1530, subnet is 2001:DB8:C18:1::/64
Joined group address(es):
  FF02::1:FF47:1530
  FF02::1
  FF02::2
MTU is 1500 bytes
```

327P_106

Cisco IOS IPv6 Name Resolution

Two ways to perform Cisco IOS name resolution for IPv6:

- Define a static name for IPv6 addresses

RouterX(config)#

```
ipv6 host name [port] ipv6addr [{ipv6addr} ...]
```

```
RouterX(config)# ipv6 host router1 3ffe:b00:ffff:b::1
```

- Configure a DNS server or servers to query

RouterX(config)#

```
ip name-server address
```

```
RouterX(config)#ip name-server 3ffe:b00:ffff:1::10
```

Configuring and Verifying RIPng for IPv6

RouterX(config)#

```
ipv6 router rip tag
```

- Creates and enters RIP router configuration mode

RouterX(config-if)#

```
ipv6 rip tag enable
```

- Configures RIP on an interface

```
show ipv6 rip
```

- Displays the status of the various RIP processes

```
show ipv6 route rip
```

- Shows RIP routes in the IPv6 route table

RIPng for IPv6 Configuration Example

