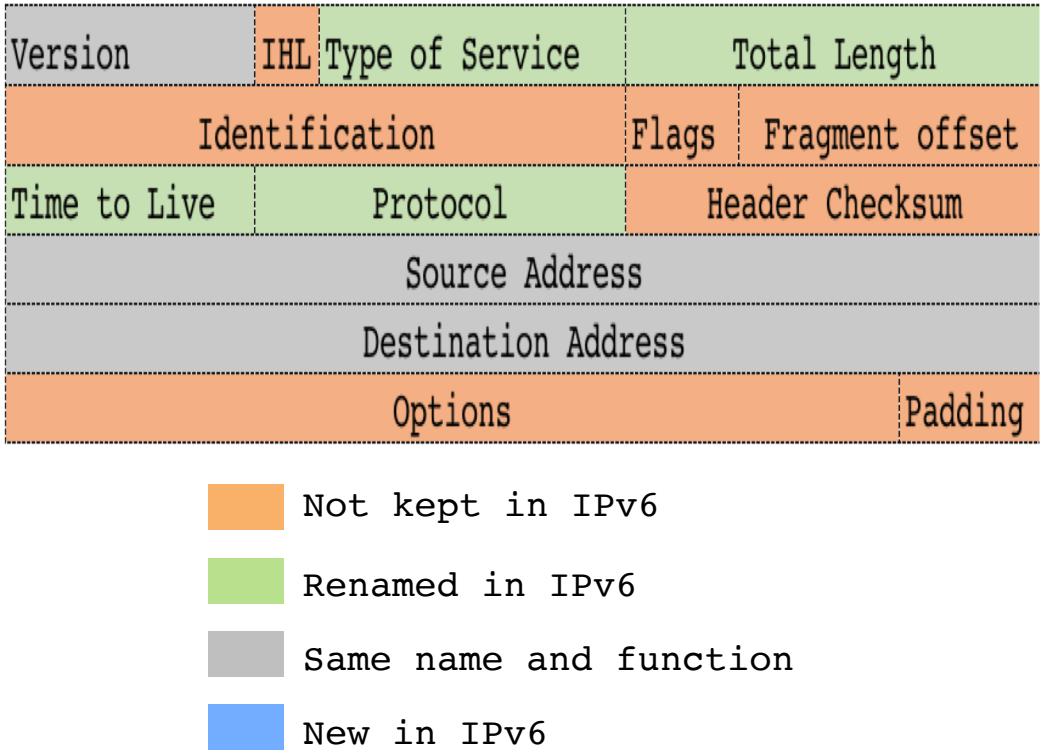
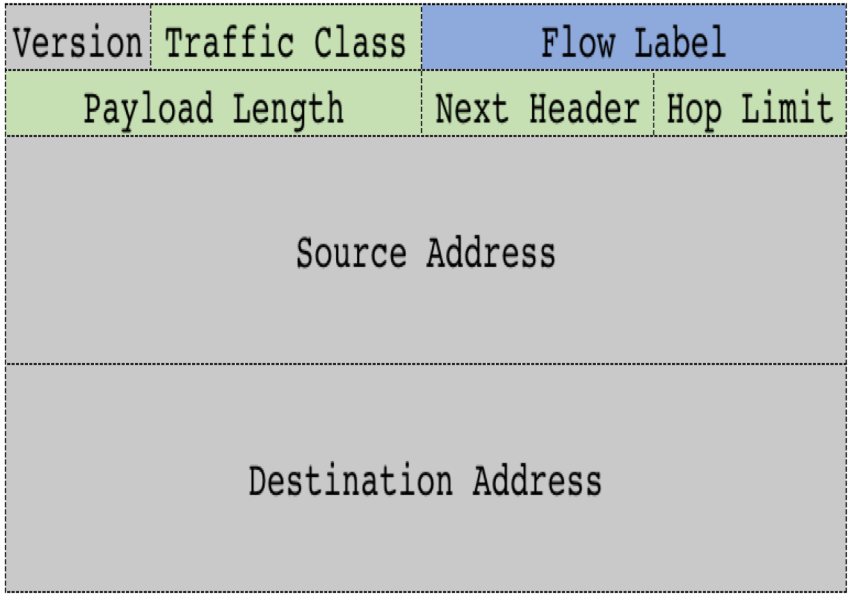
IPv6 Protocol Architecture

APNIC

v4/v6 Header Comparison



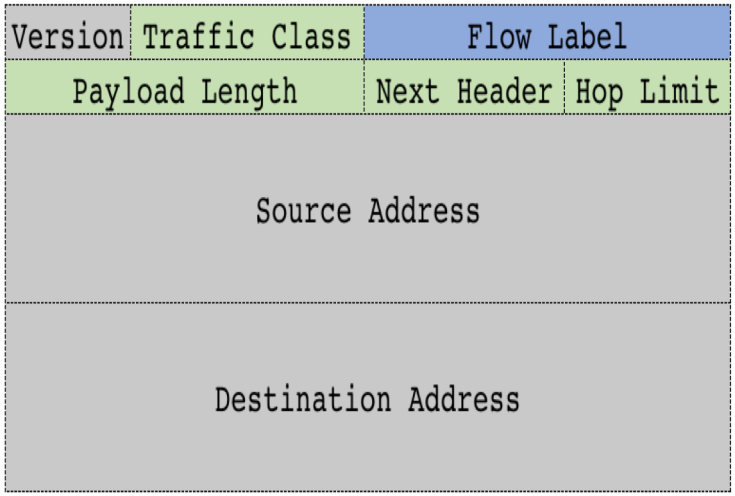


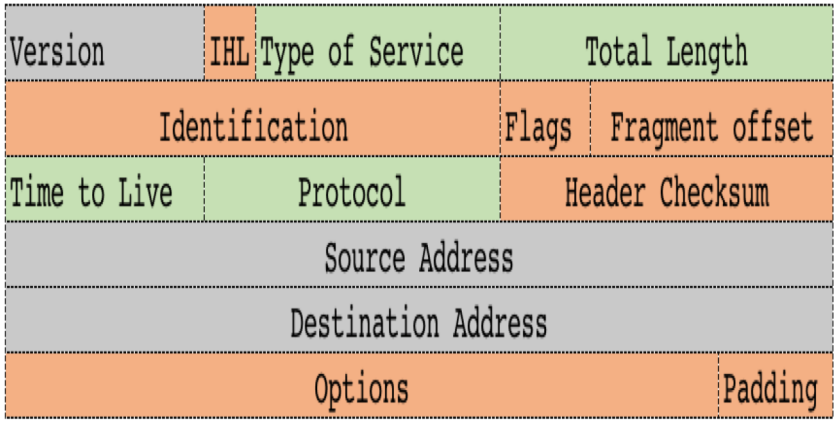
New Functional Improvement

- Address Space
  - Increase from 32-bit to 128-bit address space
- Management
  - Stateless autoconfiguration (SLAAC) means no more need to configure IP addresses for end systems, even via DHCP
- Performance
  - Simplified header means efficient packet processing
  - No header checksum re-calculation at every hop (when TTL is decremented) => left up to the lower and upper layers!
- No hop-by-hop fragmentation - PMTUD

IPv6 Protocol Header Format

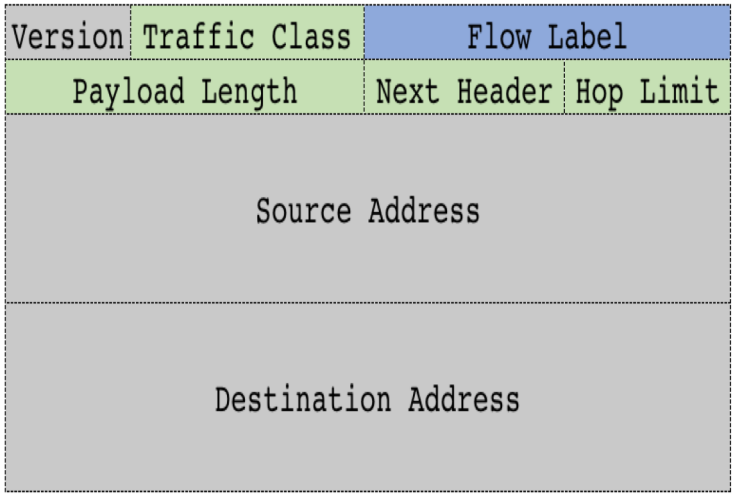
- Version (4-bit):
  - 4-bit IP version number (6)
- Traffic class (8-bit):
  - Similar to DiffServ in IPv4; define different classes or priorities.
- Flow label (20-bit):
  - allows IPv6 packets to be identified based on flows (multilayer switching techniques and faster packet-switching performance)

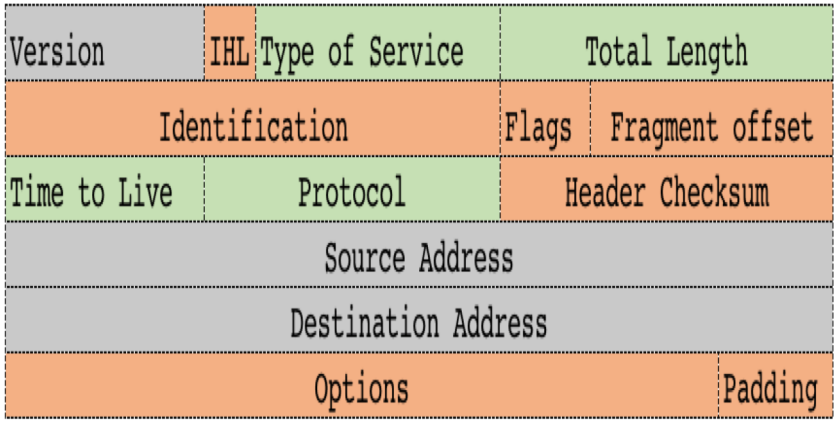




IPv6 Protocol Header Format

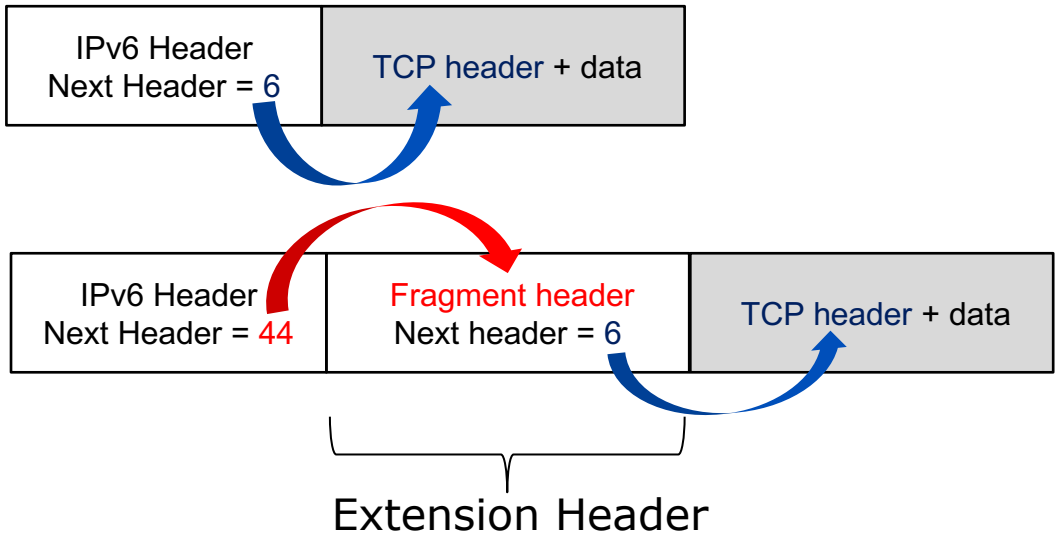
- Payload length (16-bit):
  - Defines the length of the IPv6 payload (including extension headers); Total Length in IPv4 includes the header.
- Next header (8-bit):
  - Identifies the type of information following IPv6 header. Could be upper layer (TCP/UDP), or an extension header (similar to Protocol field in IPv4).
- Hop limit (8-bit):
  - Similar to TTL in IPv4





IPv6 Extension Header

- IPv6 allows an optional *Extension Header* in between the IPv6 header and upper layer header
  - Allows adding new features to IPv6 protocol without major re-engineering



Next Header values:

0 Hop-by-hop option
2 ICMP
4 SRH
6 TCP
17 UDP
43 Source routing (RFC5095)
44 Fragmentation
50 Encrypted security payload
51 Authentication
59 Null (No next header)
60 Destination option

IPv6 Extension Header (contd)

- An IPv6 packet may carry none or many extension headers
  - A next header value of 6 or 17 (TCP/UDP) indicates there is no extension header
    - the next header field points to TCP/UDP header, which is the payload

- Unless the next header value is 0 (*Hop-by-Hop option*), extension headers are processed only by the destination node, specified by the destination address.

Fragmentation Handling In IPv6

- In IPv6, fragmentation is only performed by the host/source nodes, and not the routers along the path (unlike IPv4)
- Each source device tracks the MTU size for each session
- When a IPv6 host has large amount of data to be sent, it will be send in a series of IPv6 packets (fragmented)
  - IPv6 hosts use Path MTU Discovery (PMTUD) to determine the most optimum MTU size along the path

Source: www.cisco.com

Path MTU Discovery

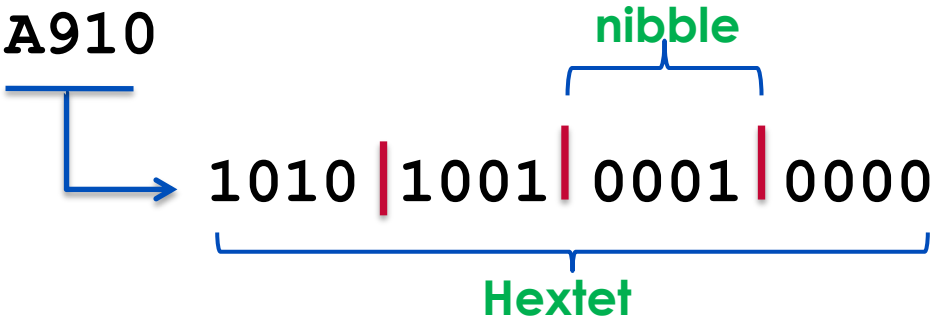
- With PMTUD, the source IPv6 device assumes the initial PMTU is the MTU of the first hop in the path
  - upper layers (Transport/Application) send packets based on the first hop MTU
  - If the device receives an "*ICMPv6 packet too big (Type 2)*" message, it informs the upper layer to reduce its packet size, based on the actual MTU size (contained in the message) of the node that dropped the packet

Path MTU =1100

IPv6 Address Representation

- IPv6 address is 128 bits
- Number of IPv6 addresses : 2^128 ~ 3.4×10^{38}
- IPv6 address is represented in hexadecimal
  - 4-bits (nibble) represent a hexadecimal digit
  - 4 nibbles (16-bits) make a hextet
  - represented as eight hextets (4 nibbles or 16 bits), each separated by a colon (:)

2001:ABCD:1234::DC0:A910



IPv6 Address Representation (2)

2001:0DB8:0000:0000:0000:036E:1250:2B00

- Abbreviated form

2001:**0**DB8:**0000:0000:0000:0**36E:1250:2B00 ⇦ **Leading 0s**

  - Leading zeroes (**0**) in any hextet can be omitted
    2001:DB8:**0:0:0**:36E:1250:2B00 ⇦ **Sequence of 0s**
  - A double colon (**::**) can replace contiguous hextet segments of zeroes
    2001:DB8**::**36E:1250:2B00 ⇦ **Double colons**
  - (**::**) can only be used once!

IPv6 Address Representation (3)

- Double colons (::) representation
  - RFC5952 recommends that the rightmost set of :0: be replaced with :: for consistency

    2001:DB8:0:0:2F:0:0:5

    2001:DB8:0:0:2F::5 instead of 2001:DB8::2F:0:0:5

- Prefix Representation
  - Representation of prefix is similar to IPv4 CIDR

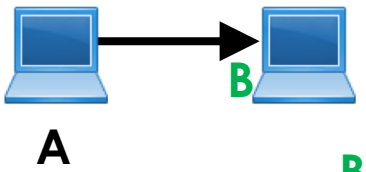
→ prefix/prefix-length

2001:DB8:12::/40

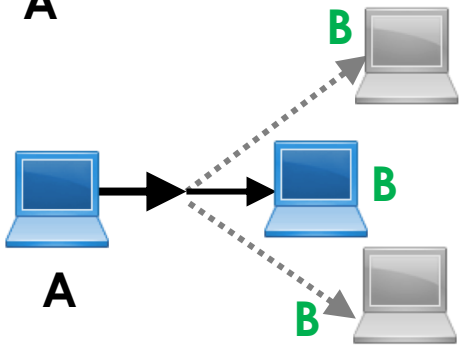
IPv6 Addressing Model

RFC 4291

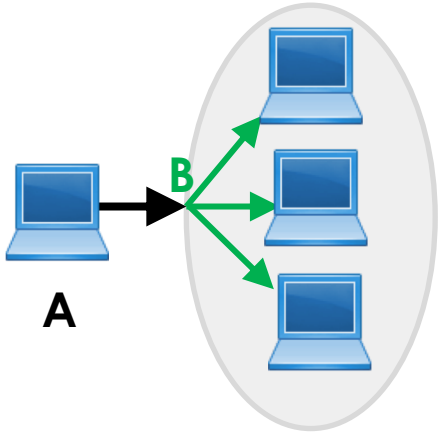
- Unicast Address
  - Assigned to a single interface
  - Packet sent only to the interface with that address



- Anycast Address
  - Same address assigned to more than one interface (on different nodes)
  - Packet for an anycast address routed to the nearest interface (routing distance)



- Multicast Address
  - group of interfaces (on different nodes) join a multicast group
  - A multicast address identifies the interface group
  - Packet sent to the multicast address is replicated to all interfaces in the group



Special Unicast Addresses

- Unspecified Address (absence of a address)

  `::/128`

- Loopback (test OSI/TCP-IP stack implementation)

  `::1/128`

Global Unicast Addresses

- Globally unique and routable IPv6 address

- Currently, only global unicast address with first three bits of **001** have been assigned

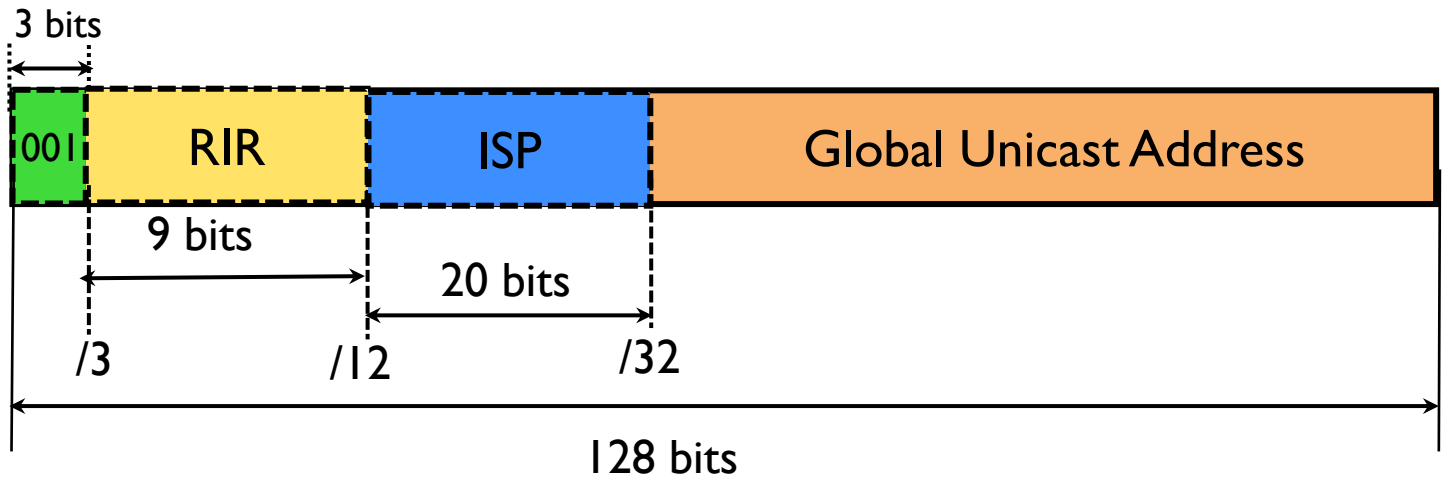
```
0010 0000 0000 0000 (2000::/3)

0011 1111 1111 1111 (3FFF::/3)
```

- IANA gives a /12 each from 2000-3FFF::/3 to each RIR

APNIC	2400::/12
ARIN	2600::/12
LACNIC	2800::/12
RIPE NCC	2A00::/12
AfriNIC	2C00::/12

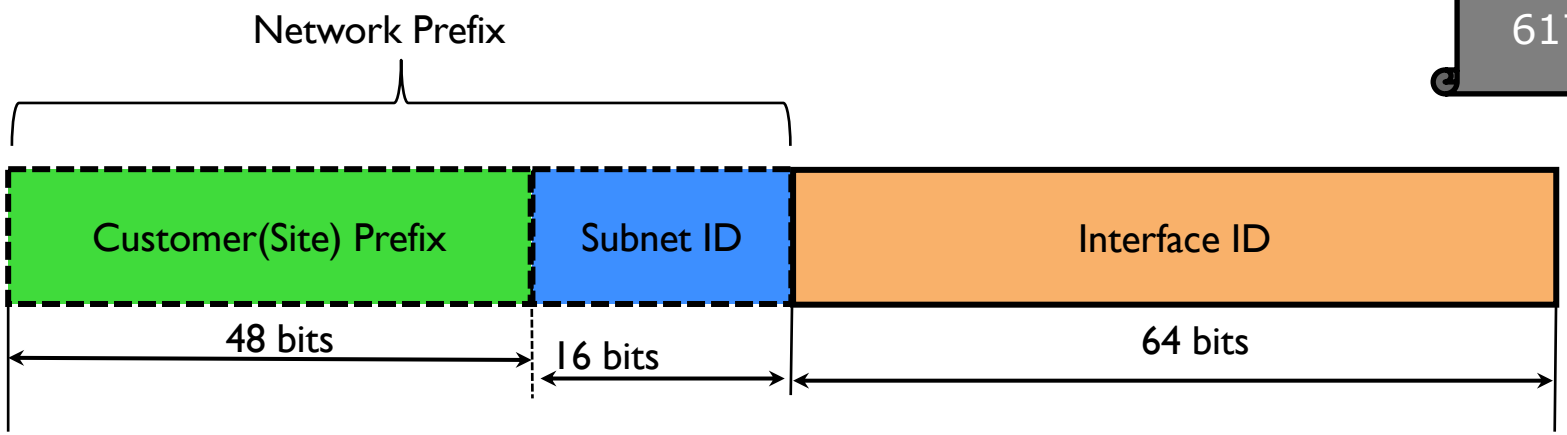
Global Unicast Addresses

- RIRs assign /32 to ISPs



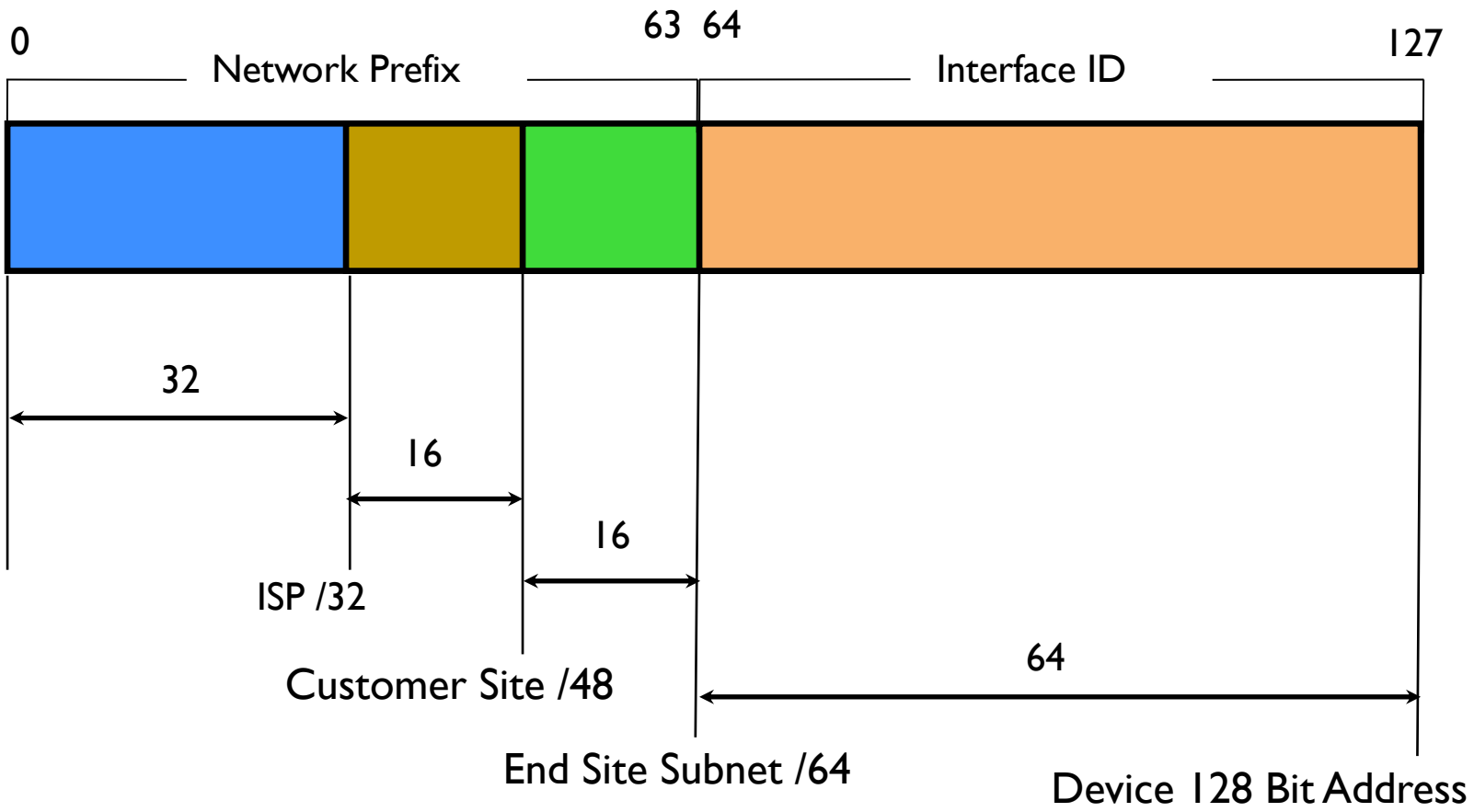
IPv6 Addressing Structure

RFC 6177



- Customer (Site) Prefix: assigned to a customer site
  - Group of subnets
  - ISPs/RIRs 'would' assign /48 (/56 to customers)
- Subnet ID: identifies the subnets (links) within a site
- Interface ID: host portion of the IPv6 address
  - how many hosts within a subnet

IPv6 Addressing Structure



Link-local Unicast Addresses

- Auto configured address (similar to APIPA)
  - Every IPv6 enabled device must have a link-local address
  - To communicate with other IPv6 devices on the same link
  - `FE80::/10`

- The link-local address is used by routers as the next-hop address when forwarding IPv6 packets

- All IPv6 hosts on a subnet/link, uses the router's link-local as the default gateway
  - Routers use the link-local as the source in ND-RA messages

Well-known Multicast Addresses

- Multicast addresses can only be destinations and never a source

  `FF00::/8`

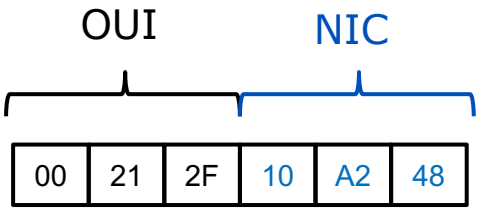
- Pre-defined multicast addresses:
  - `FF02::1` All nodes multicast
    - All IPv6 enabled devices join this multicast group
    - Packets sent to this address is received by all nodes
  - `FF02::2` All routers multicast
    - The moment IPv6 is enabled on a router (`#ipv6 unicast-routing`), the router becomes a member of this group
  - **`FF02::1:FF`**`XX:XXXX/104` Solicited Node multicast
    - NS messages (~ARP request) are sent to this address
    - Uses the least significant 24-bits of its unicast/anycast address
    - Must compute and join for every unicast (link-local & global) on a interface

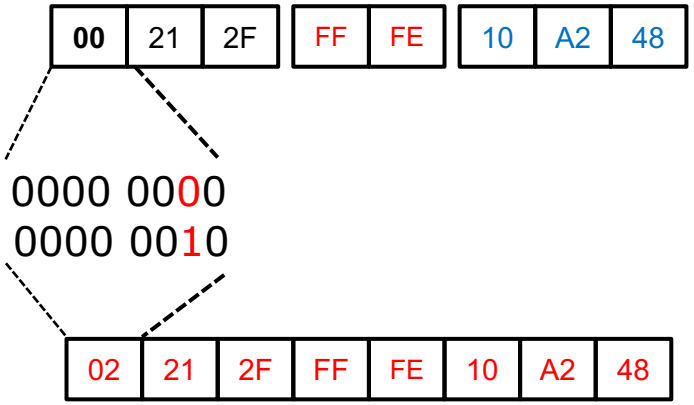
Well-known Multicast Addresses

- Pre-defined multicast addresses:
  - `FF05::1:2` All DHCP Servers/Relay Agents
    - Clients use this multicast address to discover any DHCPv6 servers/relays on the local link (link-scoped)
  - `FF05::1:3` All DHCP servers
    - Generally used by Relays to talk to servers
    - Site-scoped

Modified EUI-64 format

- Allows IPv6 device to compute a unique 64 bit Interface ID using the interface MAC address (48 bit)
  - MAC address is split into two 24 bit halves
    - OUI and NIC
  - Then `0xFFFE` is inserted between the two halves
    - 0xFFFE is reserved value, not assigned to any OEM
  - Invert 7th bit (U/L) of the OUI to get the EUI-64 address
    - addresses assigned to OEMs have this bit set to `0` to indicate global uniqueness
    - Set to 1 (invert 0) to indicate IEEE identifier is used, or 0 if otherwise.





IPv6 Addressing EUI-64

LAN: 2001:db8:213:1::/64

Eth0

```
interface Ethernet0
 ipv6 address 2001:db8:213:1::/64 eui-64
```

MAC address: 0060.3e47.1530

```
router# 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:db8:213:1:260:3EFF:FE47:1530, subnet is 2001:db8:213:1::/64
  Joined group address(es):
    FF02::1:FF47:1530
    FF02::1
    FF02::2
  MTU is 1500 bytes
```

ICMPv6 Neighbor Discovery

RFC 4861

- Router Solicitation (RS):
  - sent by IPv6 host to "all routers" multicast to request RA
- Router Advertisement (RA):
  - sent by a IPv6 router to the "all nodes" multicast (200 secs)
  - IPv6 prefix/prefix length, and default gateway
- Neighbor Solicitation (NS):
  - sent by IPv6 host to the "solicited node" multicast to find the MAC address of a given IPv6 address (~ARP request).
- Neighbor Advertisement (NA):
  - sent in response to a NS and informs of its MAC address.
- ICMPv6 Redirect:
  - informs the source of a better next-hop

IPv6 Neighbor Discovery (ND)

- Host **A** would like to communicate with Host B
  - Global address `2406:6400::10`
  - Link-local `fe80::226:bbff:fe06:ff81`
  - MAC address `00:26:bb:06:ff:81`

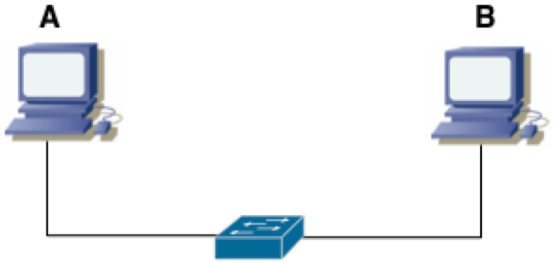
- Host **B** IPv6 global address `2406:6400::20`
  - Link-local **UNKNOWN** (if GW outside the link)
  - MAC address **UNKNOWN**

- How will Host A create L2 frame and send to Host B?

IPv6 Neighbor Discovery (ND)

Host A
IPV6 global address: 2406:6400::0010
IPv6 Link local: fe80::0226:bbff:fe06:ff81
MAC address: 00:26:bb:06:ff:81
Listen to other then above:

FF02::1 [All node multicast]
FF02:0:0:0:0:1:ff00:0010 [Solicited node m.cast unicast]
FF02:0:0:0:0:1:ff06:ff81 [Solicited node m.cast link local]



Packet

S: 2406:6400::0010 D:2406:6400::0020

ICMP6 NS Type 135

S: fe80::0226:bbff:fe06:ff81
D:FF02:0:0:0:0:1:ff00:0020

Frame

S: 00:26:bb:06:ff:81 D 33:33:ff:00:00:20
Ethernet reserved IPv6 m.cast: 33:33:xx:xx:xx:xx

Multicast enable switch: Unicast by IGMP snooping
Non multicast enable switch: broadcast, PC LAN card filter or discard

Host B
IPV6 global address: 2406:6400::0020
IPv6 Link local: fe80::0226:bbff:fe06:ff82 [Unknown to A]
MAC address: 00:26:bb:06:ff:82 [Unknown to A]
Listen to other then above:

FF02::1 [All node multicast]
FF02:0:0:0:0:1:ff00:0020 [Solicited node m.cast unicast]
FF02:0:0:0:0:1:ff06:ff82 [Solicited node m.cast link local]

Packet

S: 2406:6400::0020 D:2406:6400::0010

ICMP6 NA Type 136

S: fe80::0226:bbff:fe06:ff82
D:fe80::0226:bbff:fe06:ff81

Frame

S: 00:26:bb:06:ff:82 D 00:26:bb:06:ff:81

IPv6 Address Resolution

1

ICMPv6 NS Type135 → Multicast

SMAC: 00:26:BB:06:FF:81	**DMAC:** 33:33:FF:00:00:20
Source IPv6: FE80::0226:BBFF:FE06:FF81	**Destination IPv6:** FF02:0:0:0:0:1:FF00:0020

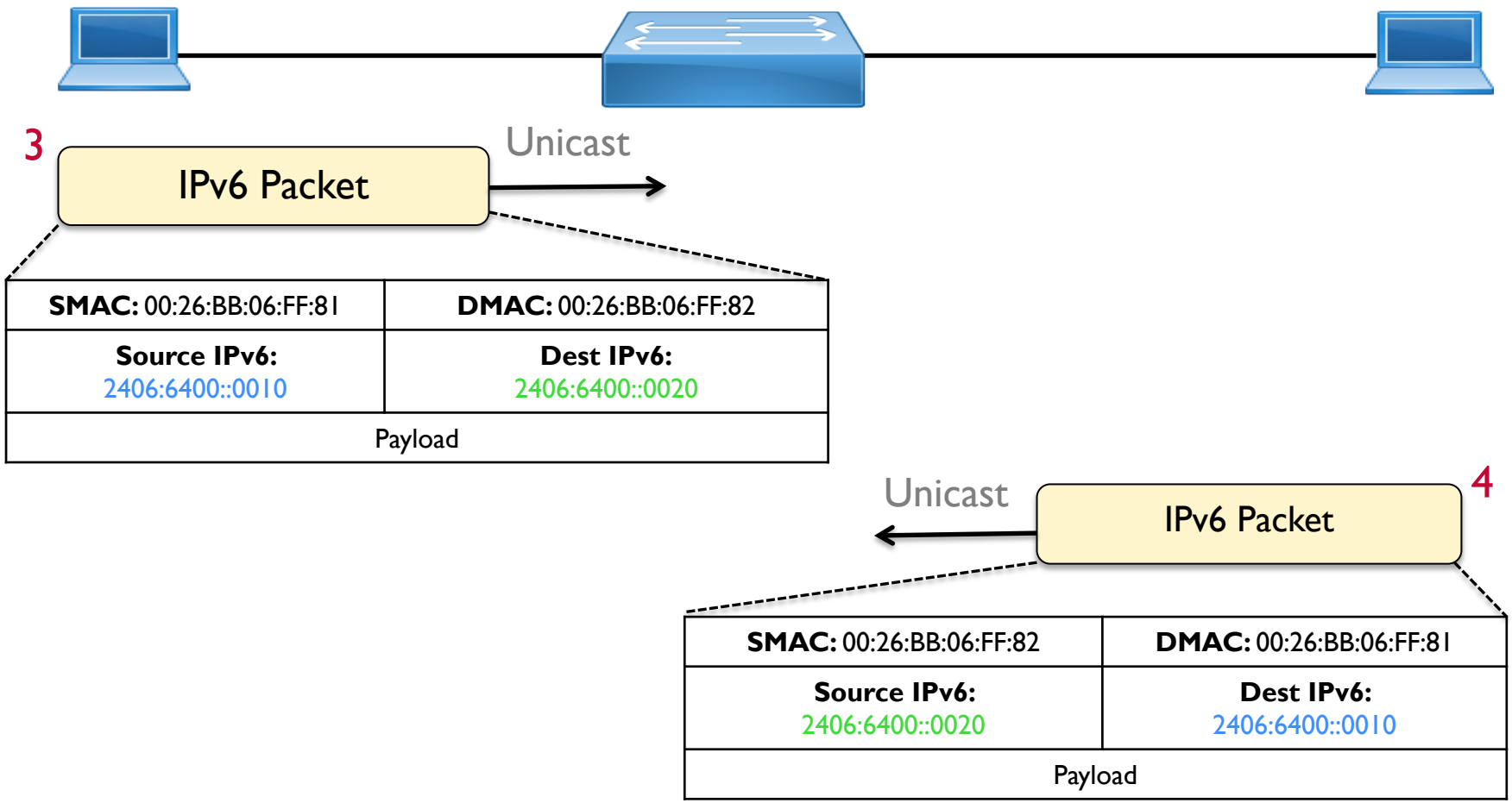
RFC2464 – 33:33:xx:xx:xx:xx

2

Unicast ← ICMPv6 NA Type136

SMAC: 00:26:BB:06:FF:82	**DMAC:** 00:26:BB:06:FF:81
Source IPv6: FE80::0226:BBFF:FE06:FF82	**Destination IPv6:** FE80::0226:BBFF:FE06:FF81

IPv6 Address Resolution



IPv6 Address Auto-configuration

- Stateless address auto-configuration (SLAAC)
  - No manual configuration required
  - Gets the IPv6 prefix and prefix length through RA (local router)
  - EUI-64 for interface ID (pseudo random)

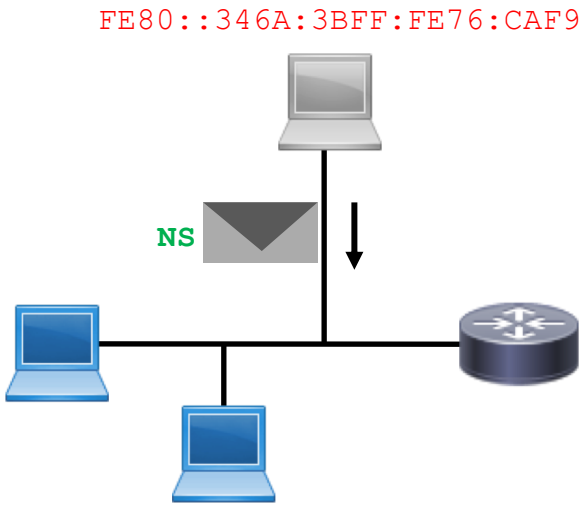
- Stateful - DHCPv6
  - To track address assignments

Stateless Address Autoconfig (1)

RFC 2462

When a host joins a link/subnet:

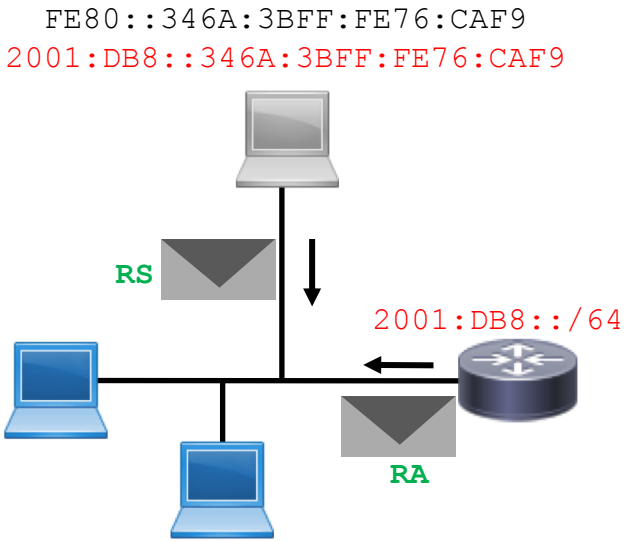
- It auto-generates a link-local using the `FE80::/10` prefix and EUI-64:
  - Ex: `FE80::346A:3BFF:FE76:CAF9`
- DAD is performed on the link-local:
  - NS message is sent to the "solicited-node" multicast (`FF02::1:FF76:CAF9`), with `::/128` as the source
  - If no NA message is received back, the generated address can be used



Stateless Address Autoconfig (2)

Once the node has a link-local address:

- sends a RS message to the "all-routers" multicast (`FF02::2`)
  - link-local as the source address
- The router responds with a RA message
  - IPv6 prefix and prefix length
  - link-local as the source
  - Auto flag by default (Managed and Other flags are not set!)
- The node generates the IPv6 address
  - uses the received prefix (`2001:DB8::/64`)
  - Interface ID (EUI-64)
  - `2001:DB8::346A:3BFF:FE76:CAF9`
  - DAD not necessary (link-local validated for the same interface!)



DHCPv6 (1)

RFC 3315

DHCPv6 is used:

- If there are no router(s) on the subnet/link, OR
- If the RA message specifies to get addressing information via DHCPv6

If the router's RA message has the:

- O (other) flag set: ***stateless DHCPv6***
  - auto-generate IPv6 address using IPv6 prefix & prefix length in the RA
  - obtain other information (DNS server, domain) via DHCPv6
- M (managed) flag set:
  - obtain all addressing information via DHCPv6
  - 'O' flag is redundant

Stateful Autoconfig – DHCPv6 (2)

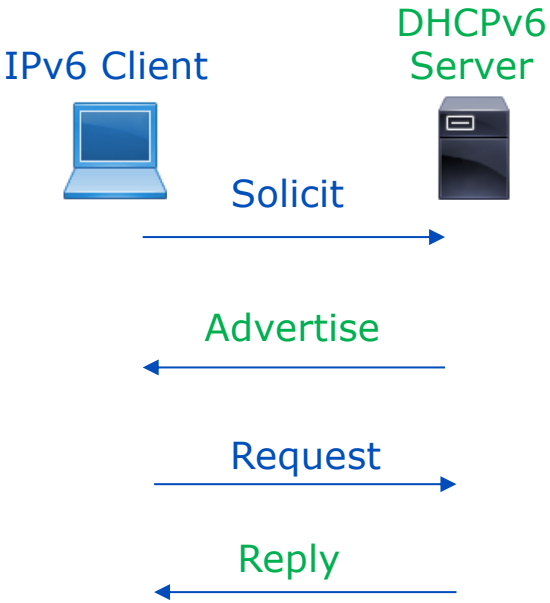
1. Client sends Solicit message to `FF02::1:2` to find any available DHCPv6 servers

2. Server responds with an Advertise message
   - the tentative IPv6 address/prefix
   - Other parameters (DNS, domain, default gateway, lease time)
   - *could receive multiple Advertise messages*

3. Client selects the server, and sends a Request asking to formally request the indicated IPv6 address

4. Server responds with a Reply to confirm the assignment

5. Performs DAD before using!



IPv6 Interface ID – Privacy

- Overcome the ability to track (interface ID based on MAC address):
  - Temporary address (changes): outgoing connections
  - Secured address: incoming connection

RFC 4941

```
Temp > 2001:dc0:a000:4:84a3:49b6:1919:26fb
Secured> 2001:dc0:a000:4:108b:3690:9335:b7ec
Temp > 2001:dc0:a000:4:14e6:d4a3:815d:91dd
```

- Ease network management yet improve privacy:
  - Stable interface identifiers for each subnet

```
Secured> 2001:dc0:a000:4:cbb:347c:6215:1083
```

RFC 7217

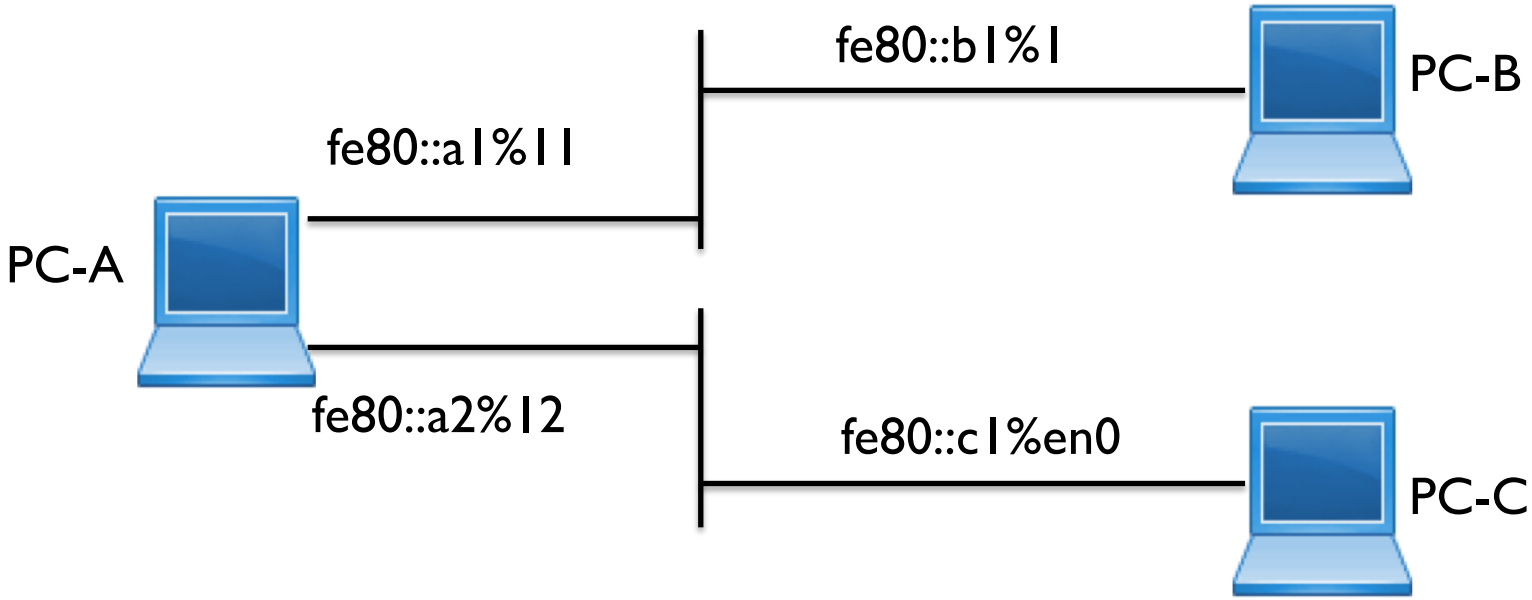
Zone IDs for Link-locals

Interface en0 - `fe80::4e0:37e4:c5d1:c845`**`%en0`**
Interface en5 - `fe80::aede:48ff:fe00:1122`**`%en5`**

- Zone IDs help uniquely distinguish which link/subnet an interface is connected to

- To ping a remote IPv6 node, use your interface zone ID (so that the response packet has a path)

Quiz - Zone ID



- Please write down the commands:
  - PC-A pings PC-B
  - PC-A telnet PC-C

Subnetting (Example)

- Provider **A** has been allocated

  2001:DB8::/32

  – will delegate /48 blocks to its customers

Q. Find the blocks provided to the first 4 customers

Subnetting (Example)

Original block: **2001:0DB8::/32**

Rewrite as a /48 block: **2001:0DB8:0000::/48**

This is your network prefix!

How many /48 blocks are there in a /32?

2^16 = 65K

Find only the first 4 /48 blocks…

Subnetting (Example)

Start by manipulating the LSB of your network prefix – write in bits

2001:0DB8:0000::/48

In bits

2001:0DB8:	**0000 0000 0000 0000**	**::/48**	➡	**2001:0DB8:0000::/48**
2001:0DB8:	**0000 0000 0000 0001**	**::/48**	➡	**2001:0DB8:0001::/48**
2001:0DB8:	**0000 0000 0000 0010**	**::/48**	➡	**2001:0DB8:0002::/48**
2001:0DB8:	**0000 0000 0000 0011**	**::/48**	➡	**2001:0DB8:0003::/48**

Then write back into hex digits

Exercise 1.1: IPv6 subnetting

Identify the first four /36 address blocks out of 2406:6400::/32

1. _____
2. _____
3. _____
4. _____

Exercise 1.2: IPv6 subnetting

Identify the first four /35 address blocks out of 2406:6400::/32

1. _____
2. _____
3. _____
4. _____


Questions