18CSC302J-Computer Networks

Unit –IV – IPv6 Overview

Course Learning Outcome:

- The student will be able to learn and understand IPv6 technologies.
- The students can able to analyze and compare the IPv4 and IPv6 protocols.

Topics Covered

- IPV6 Features
- IPV6 Addressing Modes
- IPV6 Address Types
- Address Space Allocation
- Global Unicast Addresses
- Auto configuration, Renumbering
- IPV6 Routing Protocols
- IPV6 Packet Format
- Comparison between IPV4 and IPV6 Header
- IPV4 to IPV6 Tunneling
- IPV4 to IPV6 Translation Techniques
- NAT Protocol Translation
- IPV6 Mobility
- Protocols Changed to Support IPV6

IPv6 Overview

- IPv4 stood the test of time- a tribute to its initial design
  - ➢ Proven to be robust
  - ➢ Easily implemented
  - ➢ Interoperable

- The initial design did not anticipate today's Internet scale and size.
  - ➢ Exhausting IPv4 address space
  - ➢ Large routing tables
  - ➢ Simpler management of IPv4 addresses
  - ➢ Security at IP level
  - ➢ QoS requirements

IPv6 Features

The changes introduced by IPv6 can be grouped into seven categories:

- **Larger Addresses:** The new address size is the most noticeable change. IPv6 quadruples the size of an IPv4 address from 32 bits to 128 bits. The IPv6 address space is so large that it cannot be exhausted in the foreseeable future.
- **Extended Address Hierarchy**:IPv6 uses the larger address space to create additional levels of addressing hierarchy. In particular, Pv6 can define a hierarchy of ISPs as well as a hierarchical structure within a given site.
- **Flexible Header Format:** IPv6 uses an entirely new and incompatible datagram format. Unlike the IPv4 fixed-format header, IPv6 defines a set of optional headers.
- **Improved Options:** Like IPv4, IPv6 allows a datagram to include optional control information. IPv6 includes new options that provide additional facilities not available in IPv4.
- . **Provision For Protocol Extension: Perhaps** the most significant change in IPv6 is a move away from a protocol that fully specifies all details to a protocol that can permit additional features. The extension capability has the potential to allow the IETF to adapt the protocol to changes in underlying network hardware or to new applications.
- **Support For Auto configuration And Renumbering:** IPv6 provides facilities that allow computers on an isolated network to assign themselves addresses and begin communicating without depending on a router or manual configuration. The protocol also includes a facility that permits a manager to renumber networks dynamically.
- **Support For Resource Allocation:** IPv6 has two facilities that permit pre-allocation of network resources: a flow abstraction and a differentiated service specification. The latter will use the same approach as IPv4's differentiated services.

IPv6 Addressing Modes:

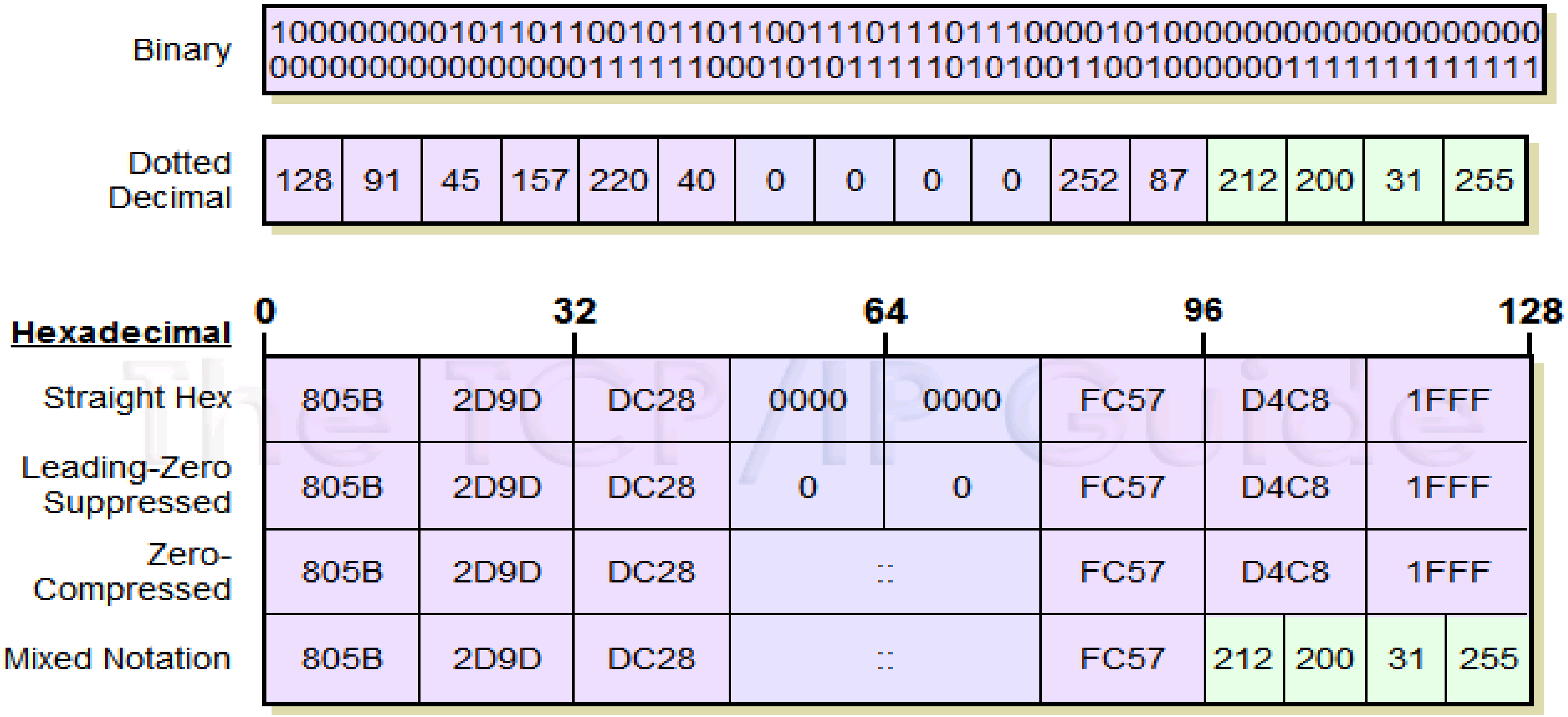
- **128 bits (or 16 bytes) long:** four times as long as its predecessor.
- **2^{128}** : about 340 billion billion billion billion different addresses
- **Colon hexadecimal notation**:
  - addresses are written using 32 hexadecimal digits.
  - digits are arranged into 8 groups of four to improve the readability.
  - Groups are separated by colons

2001:0718:1c01:0016:020d:56ff:fe77:52a3

- Note:
  - DNS plays an important role in the IPv6 world
    - (manual typing of IPv6 addresses is not an easy thing,
    - Some **zero suppression rules** are allowed to lighten this task at least a little.

IPv6 Address Notation: Example

128.91.45.157.220.40.0.0.0.0.252.87.212.200.31.255



Binary

```
1000000001011011001011011001110111011100001010000000000000000000
0000000000000000111111000101011111010100110010000001111111111111
```

Dotted Decimal	128	91	45	157	220	40	0	0	0	0	252	87	212	200	31	255

Hexadecimal

0
32
64
96
128

Straight Hex	805B	2D9D	DC28	0000	0000	FC57	D4C8	1FFF
Leading-Zero Suppressed	805B	2D9D	DC28	0	0	FC57	D4C8	1FFF
Zero-Compressed	805B	2D9D	DC28	::		FC57	D4C8	1FFF
Mixed Notation	805B	2D9D	DC28	::		FC57	212 200	31 255

Rule 1- IPv6 Zero Suppression

- Some types of addresses contain long sequences of zeros.
- To further simplify the representation of IPv6 addresses, a contiguous sequence of 16-bit blocks set to 0 in the colon hexadecimal format can be compressed to **"::", known as *double-colon*.**

- For example:
  - **link-local address**
    - FE80:0:0:0:2AA:FF:FE9A:4CA2 → FE80::2AA:FF:FE9A:4CA2.
  - **multicast address**
    - FF02:0:0:0:0:0:0:2 → FF02::2
  - **loopback address**
    - 0:0:0:0:0:0:0:1 → ::1

Rule 1- IPv6 Zero Suppression

- Zero compression can only be used to compress a single contiguous series of 16-bit blocks expressed in colon hexadecimal notation.
- You cannot use zero compression to include part of a 16-bit block.

- For example,
  - cannot express FF02:30:0:0:0:0:0:5 as FF02:3::5
  - correct representation = FF02:30::5
- Leading zeroes in every group can be omitted.

2001:718:1c01:16:20d:56ff:fe77:52a3

Rule 1- IPv6 Zero Suppression

- **To determine the number of 0 bits represented by the "::"**
  1. count the number of blocks in the compressed address
  2. (-) subtract this number from 8
  3. (*) multiply the result by 16.

- **For example**
  1. FF02::2
  2. two blocks - "FF02" block and "2" block.
  3. The number of bits expressed by the "::" is 96 (96 = (8 – 2)×16).

- **Zero compression can only be used once in a given address.**
  - Otherwise, you could not determine the number of 0 bits represented by each instance of "::".

Example1:

Show the unabbreviated colon hex notation for the following IPv6 addresses:

a. An address with 64 0s followed by 64 1s.
b. An address with 128 0s.
c. An address with 128 1s.
d. An address with 128 alternative 1s and 0s.

Solution

a. 0000:0000:0000:0000:FFFF:FFFF:FFFF:FFFF
b. 0000:0000:0000:0000:0000:0000:0000:0000
c. FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF
d. AAAA:AAAA:AAAA:AAAA:AAAA:AAAA:AAAA:AAAA

Example 2

The following shows the zero contraction version of addresses in Example 1 (part c and d cannot be abbreviated)

a. :: FFFF:FFFF:FFFF:FFFF
b. ::
c. FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF:FFFF
d. AAAA:AAAA:AAAA:AAAA:AAAA:AAAA:AAAA:AAAA

Example 3

Show abbreviations for the following addresses:

a. 0000:0000:FFFF:0000:0000:0000:0000:0000

b. 1234:2346:0000:0000:0000:0000:0000:1111

c. 0000:0001:0000:0000:0000:0000:1200:1000

d. 0000:0000:0000:0000:0000:FFFF:24.123.12.6

Solution

a. 0:0:FFFF::

b. 1234:2346::1111

c. 0:1::1200:1000

d. ::FFFF:24.123.12.6

Example 4

Decompress the following addresses and show the complete unabbreviated IPv6 address:

a. 1111::2222
b. ::
c. 0:1::
d. AAAA:A:AA::1234

Solution

a. 1111:0000:0000:0000:0000:0000:0000:2222
b. 0000:0000:0000:0000:0000:0000:0000:0000
c. 0000:0001:0000:0000:0000:0000:0000:0000
d. AAAA:000A:00AA:0000:0000:0000:0000:1234

IPv6 Prefixes

- The prefix is the part of the address that indicates the bits that have fixed values or are the bits of the subnet prefix.
- Prefixes for IPv6 subnets, routes, and address ranges are expressed in the same way as Classless Inter-Domain Routing (CIDR) notation for IPv4.
- An IPv6 prefix is written in ***address/prefix-length*** notation.
    - For example, **21DA:D3::/48 and 21DA:D3:0:2F3B::/64** are IPv6 address prefixes.

- **Note** IPv4 implementations commonly use a dotted decimal representation of the network prefix known as the subnet mask. A subnet mask is not used for IPv6. Only the prefix length notation is supported.

IPv6 Prefixes

- When writing both a node address and a prefix of that node address (e.g., the node's subnet prefix), the two can combined as follows:
- The node address:

  12AB:0:0:CD30:123:4567:89AB:CDEF

- And its subnet number:

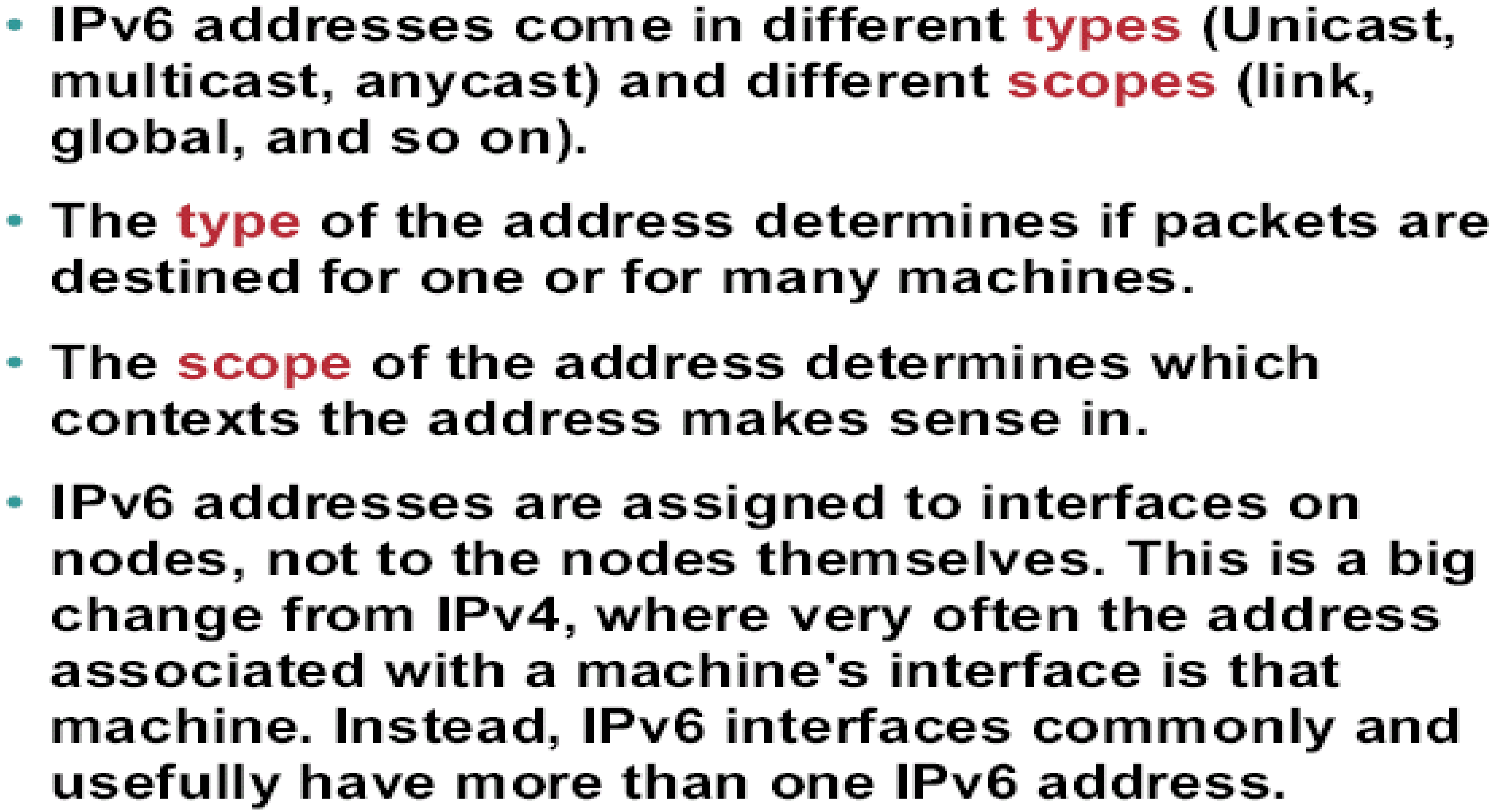
  12AB:0:0:CD30::/60

- Can be represented as

  12AB:0:0:CD30:123:4567:89AB:CDEF/60

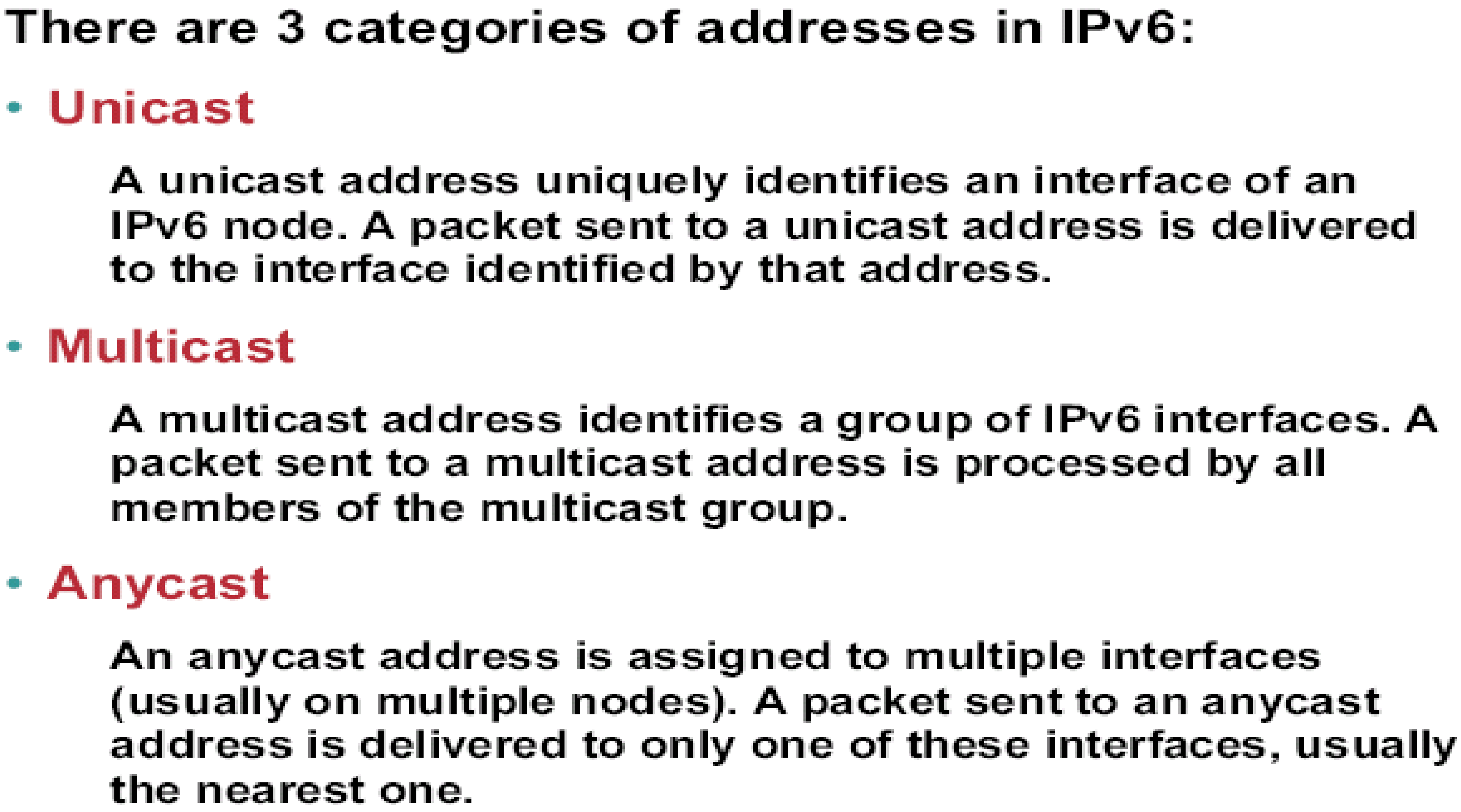
IPv6 Address Types

- IPv6 Addresses: Types and Scopes



  - IPv6 addresses come in different types (Unicast, multicast, anycast) and different scopes (link, global, and so on).
  - The type of the address determines if packets are destined for one or for many machines.
  - The scope of the address determines which contexts the address makes sense in.
  - IPv6 addresses are assigned to interfaces on nodes, not to the nodes themselves. This is a big change from IPv4, where very often the address associated with a machine's interface is that machine. Instead, IPv6 interfaces commonly and usefully have more than one IPv6 address.

IPv6 Address Categories



There are 3 categories of addresses in IPv6:

- **Unicast**

  A unicast address uniquely identifies an interface of an IPv6 node. A packet sent to a unicast address is delivered to the interface identified by that address.

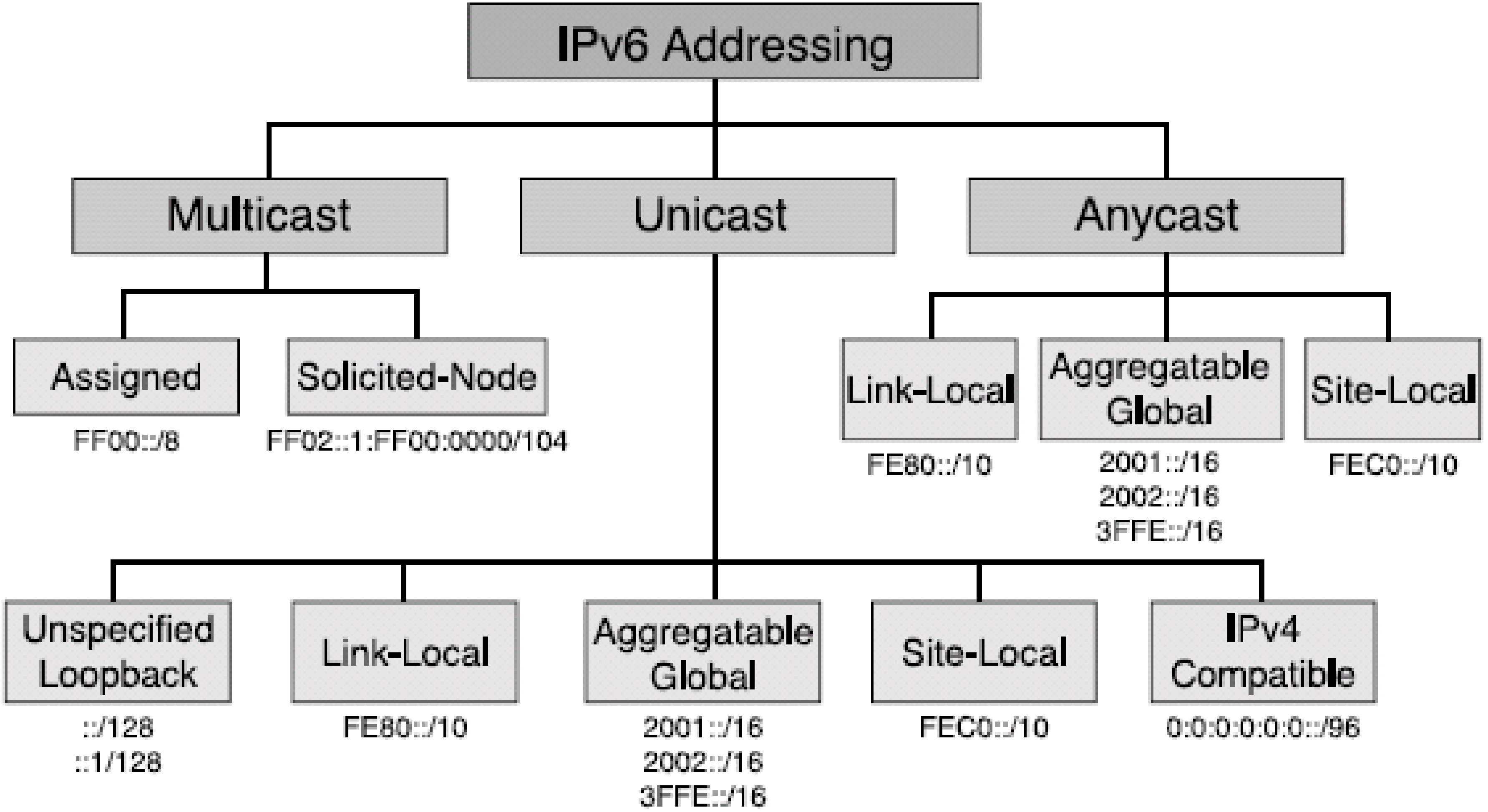
- **Multicast**

  A multicast address identifies a group of IPv6 interfaces. A packet sent to a multicast address is processed by all members of the multicast group.

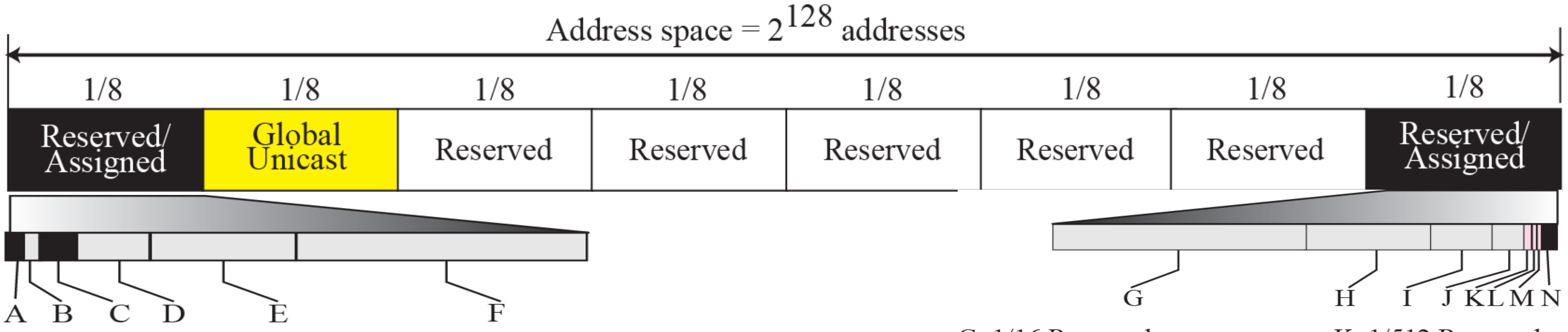
- **Anycast**

  An anycast address is assigned to multiple interfaces (usually on multiple nodes). A packet sent to an anycast address is delivered to only one of these interfaces, usually the nearest one.

IPv6 Address Types



Address space allocation



A: 1/256 IPv4 Compatible

B: 1/256 Reserverd

C: 1/128 Reserved

D: 1/64 Reserved

E: 1/32 Reserved

F: 1/16 Reserved

G: 1/16 Reserved

H: 1/32 Reserved

I: 1/64 Reserved

J: 1/128 Unique Local Unicast

K: 1/512 Reserved

L: 1/1024 Link Local

M: 1/256 Reserved

N: 1/256 Multicast

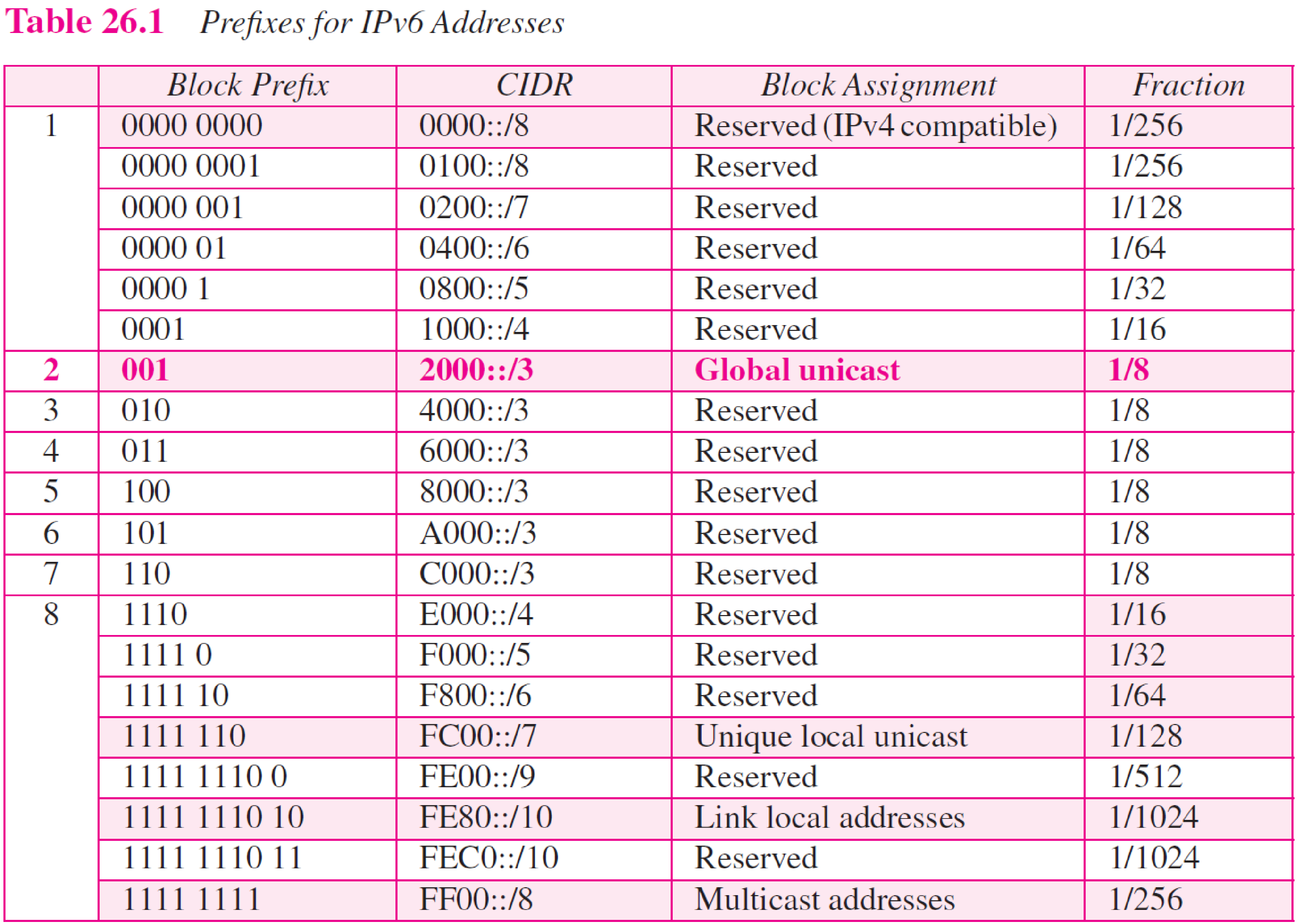


Table 26.1 *Prefixes for IPv6 Addresses*

	Block Prefix	*CIDR*	*Block Assignment*	*Fraction*
1	0000 0000	0000::/8	Reserved (IPv4 compatible)	1/256
	0000 0001	0100::/8	Reserved	1/256
	0000 001	0200::/7	Reserved	1/128
	0000 01	0400::/6	Reserved	1/64
	0000 1	0800::/5	Reserved	1/32
	0001	1000::/4	Reserved	1/16
2	**001**	**2000::/3**	**Global unicast**	**1/8**
3	010	4000::/3	Reserved	1/8
4	011	6000::/3	Reserved	1/8
5	100	8000::/3	Reserved	1/8
6	101	A000::/3	Reserved	1/8
7	110	C000::/3	Reserved	1/8
8	1110	E000::/4	Reserved	1/16
	1111 0	F000::/5	Reserved	1/32
	1111 10	F800::/6	Reserved	1/64
	1111 110	FC00::/7	Unique local unicast	1/128
	1111 1110 0	FE00::/9	Reserved	1/512
	1111 1110 10	FE80::/10	Link local addresses	1/1024
	1111 1110 11	FEC0::/10	Reserved	1/1024
	1111 1111	FF00::/8	Multicast addresses	1/256

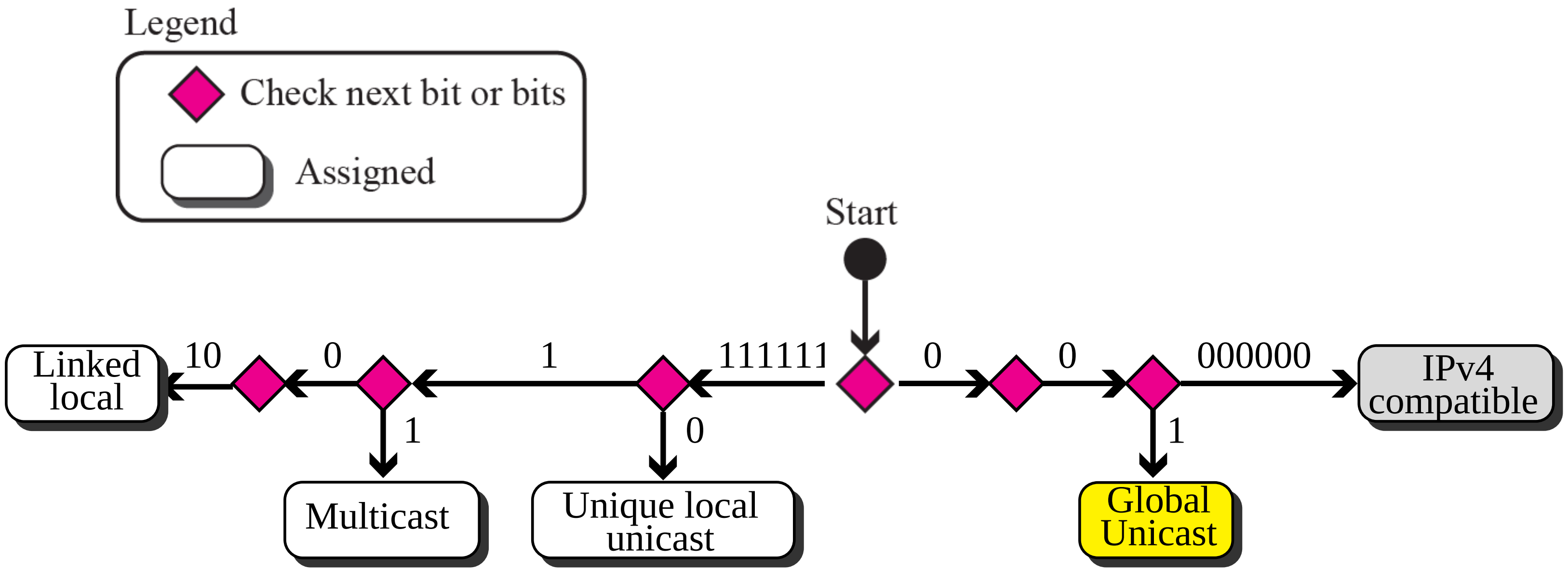
Example

- Figure Address space allocation shows that only a portion of the address space can be used for global unicast communication. How many addresses are in this block?

Solution

- This block occupies only one-eighth of the address spaces. To find the number of addresses, we can divide the total address space by 8 or 2^3 . The result is $(2^{128})/(2^3) = 2^{125}$ —a huge block.

Algorithm for finding the allocated blocks

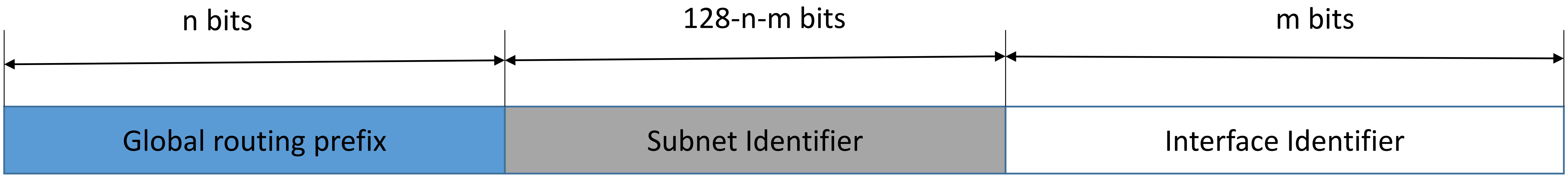


Global Unicast Addresses

- Primary used to address the System for one-one Communication mechanism i.e host to host direct communication over the internet.
- Global unicast address is equivalent to public IPV4 address
- Global unicast address objective is to reach any host globally across the internet uniquely
- Address block refer this is called global unicast address block
- CIDR Notation for the block is 2000::/3, where 3 refers to that 3 leftmost bit is common for all address in this block (001)
- The size of the address space is 2^{125} which is more than for expansion of internet in many years

Global Unicast Address

- test



n bits

128-n-m bits

m bits

Global routing prefix

Subnet Identifier

Interface Identifier

Global Unicast Address

Block Assignment	**Length of block**
Global routing prefix (n)	48 bits
Subnet Identifier (128-n-m)	16 bits
Interface Identifier	64 bits

Recommended length for each block in Global unicast address

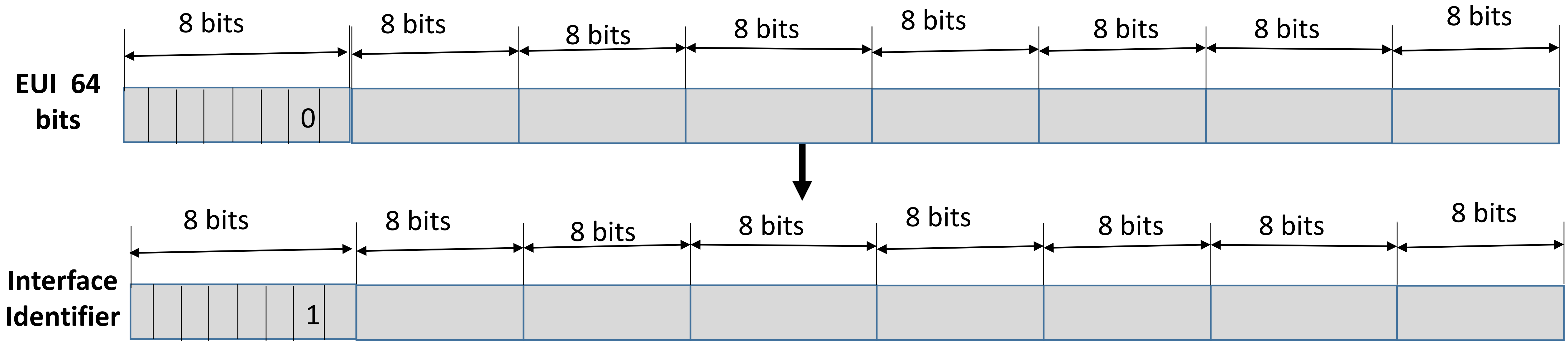
Three levels of Hierarchy

- **Global Routing Prefix** : First block of 48 bits together form global routing preference. Its used to route the packet to the organization site (ISP) through the internet, Since the first three bits (001) is predefined, the next 45 bits can used to define 2^{45} sites. The routers across the internet route the packet to the destination based on the value of n.

- **Subnet Identifier :** 16 bit block is used to identify the specific subnet of an organization. An organization can have upto 2^{16} subnets.

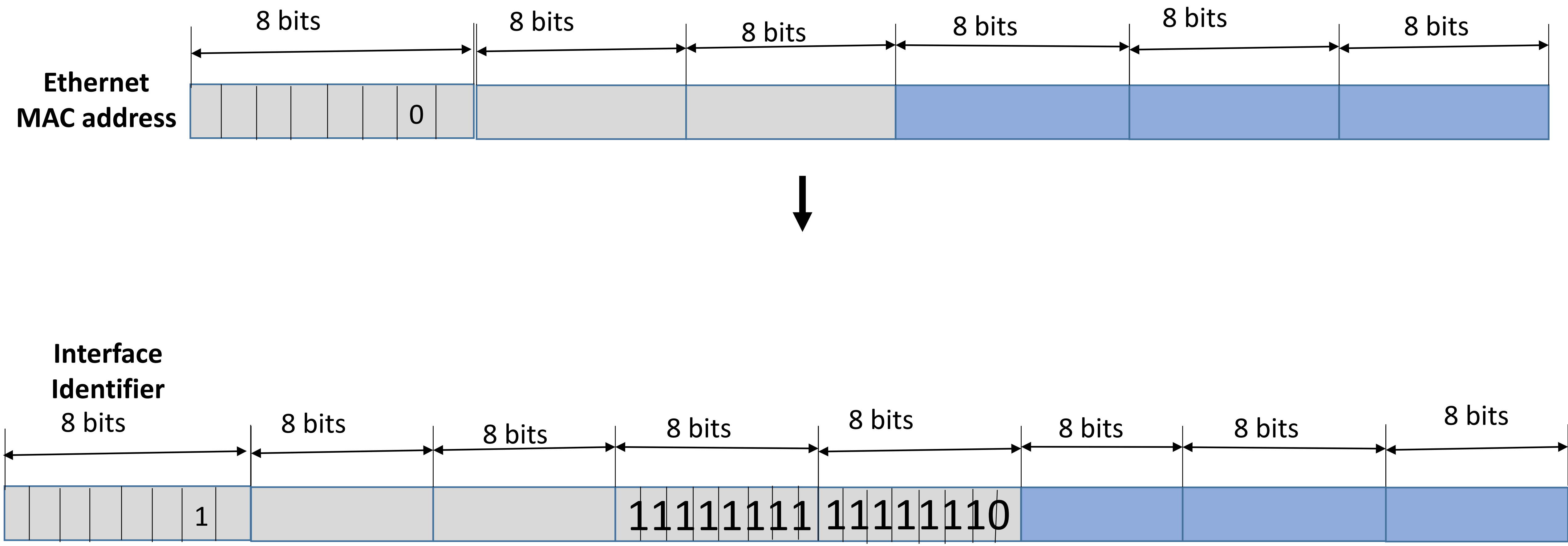
- **Interface Identifier :** last 64 bits refers to the interface identifier which is used to identify the Host which is similar to the hostId in IPV4 scheme. In IPV4 addressing, there is no relation between the hostid (32 bits) and MAC(48 bits) due to the difference in length. Since IPV6 64 bits of length the Physical address of the host can be embedded as whole or part of the identifier which helps in locating the host without any mapping. Two common physical addressing scheme can be considered for this purpose: the 64-bit extended unique identifier (EUI-64) defined by IEEE and the 48-bit physical address defined by Ethernet.

Mapping EUI-64 to interface identifier



To map the physical address (MAC), the global bit of the format needs to be changed from 0 to 1

Mapping MAC Address to interface identifier



To map the physical address (MAC), the global bit of the format needs to be changed from 0 to 1

Example of mapping address from one format to other

1. Find the interface identifier if the physical address in the EUI is (F5-A9-23-EF-07-14-7A-D2)16 using the format we defined for Ethernet addresses.

Soln : To map the EUI to interface identifier we need to invert the seventh bit of first octet. The first octet value

F5 -> (11110101)

F7 -> (11110111)

Interface identifier is F7-A9-23-EF-07-14-7A-D2

2. Find the interface identifier if the Ethernet physical address is (F5-A9-23-14-7A-D2)16 using the format we defined for Ethernet addresses.

Soln : To map the Ethernet address to interface identifier, we need to invert the seventh bit of first octet and FFFE has to be inserted after the 3 octet.

F5-A9-23-14-7A-D2 (11110101)

F7-A9-23-FF-FE-14-7A-D2 (11110111)

Auto Configuration

- Auto configuration enables the host to assign the IPV6 address on its own. In IPV4 usually the network manager will assign the address (static) for each host in the network in-case if the network uses DHCP (dynamic host configuration protocol) the host will be assign a dynamic address when the host join the network and it keeps changing every time it joins the network**.**

- **Auto Configuration process:**

  1. Host create a link local address by taking 10 bit local prefix (1111 1110 10) and add 54 zeros and adding 64 bits interface identifier of its own from the interface card which makes as 128 bit link local address.

  2. The host verifies the uniqueness of the link local address by sending the neighbour solicitation message and waits for the neighbour advertisement message. Incase if any of the host address matches then auto configuration process results in failure which can be counter by either DHCP or manual configuration

  3. If the uniqueness test for link local address is successful, then the host send router solicitation message to the local router. If the local router running in the network sends a router advertisement message from which thee host extract the global unicast prefix and the subnet prefix and append the same with local link to complete the address. Incase if the router cant help for auto configuration it inform the host by setting the flag in the advertisement message.

Computing the global unicast address

1. Assume a host with Ethernet address (F5-A9-23-11-9B-E2)16 has joined the network. What would be its global unicast address if the global unicast prefix of the organization is 3A21:1216:2165 and the subnet identifier is A245:1232.

Soln:

Step 1 : Creating a local link address by adding 10 bit prefix (1111 1110 10) and 54 zeros and append its 64 bit interface ID extracted from the Ethernet address :

FE80 : :F7A9-23FF-FE11-9BE2 (by inverting the seventh bit of 1st octet and adding FFFE after the third octet)

Step 2 : On assuming this uniqueness it send the router solicitation message upon receiving the advertisement message it complete the auto configuration process by extracting the global unicast prefix and subnet identifier from the message as follows 3A21:1216:2165:A245:1232 and append it to the local link address

3A21:1216:2165:A245:1232: F7A9-23FF-FE11-9BE2

Renumbering

Renumbering allow the site to change the service provider and reconfigure the IPV6 address. If the site changes the service provider the address prefix needs to be changed. Once the service provider changes the router advertises the new prefix and the site uses the old prefix before its disabling. The main hindrance in renumbering is support of the DNS, which needs to propagate the new addressing associated with a domain name. Anew protocol called Next generation DNS in exploration.

IPV6 Introduction

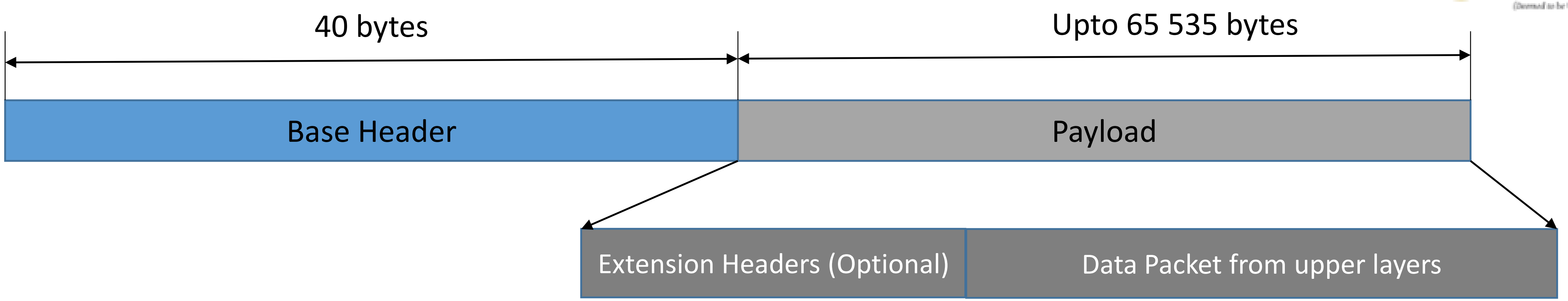
The evolution of IPV6 was due to two major factors:

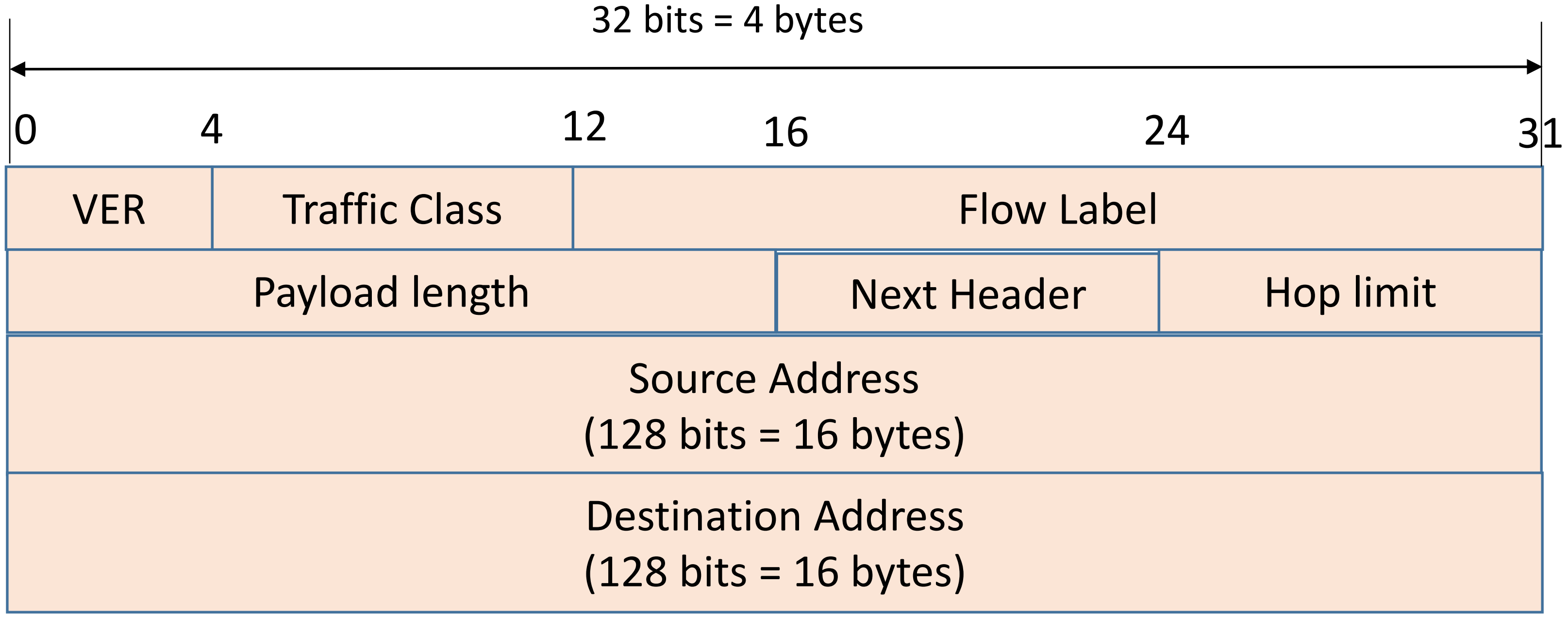
1. Address depletion
2. slowness of the process due to some unnecessary processing, and support for multimedia and security.

IPV6 responds to the above issue by the following modifications

- **Larger address space.** Instead of 32bit addressing scheme it uses 128 bit addressing format.
- **Better header format**. Options are flexible i.e., removed from base header and inserted when needed which speed up the routing process.
- **New options**. IPv6 has new options to allow for additional functionalities.
- **Allowance for extension**.
- **Support for resource allocation**. In place of type-of-service field two new fields, traffic class and flow label have been added to enable the source to request special handling of the packet which enable the support for multimedia transmission
- **Support for more security.** The encryption and authentication options in IPv6 provide confidentiality and integrity of the packet.

IPV6 Packet format





Format of Base Header

- Version : To specify whether IPV4 or IPV6
- Traffic Class : Distinguish the payload.
- Flow label : Mention special handling for a particular flow of data.
- Payload length : Defines the length of the IP datagram in payload.
- Next Header : Optional extension headers used by IP or the header of an encapsulated packet such as UDP or TCP.
- Hop Limit : TTL
- Source Address: Original source address
- Destination Address: Final destination of datagram

Next Header codes

Code	Next Header
0	Hop-by-Hop option
2	ICMP
6	TCP
17	UDP
43	Source routing
44	Fragmentation
50	Encrypted Security Payload
51	Authentication
59	NULL (no next header)
60	Destination Option

Flow Label

- IP protocol designed to serve as connectionless protocol, but it has the ability to serve as connection oriented protocol.
- Router consider the flow as a sequence of packet share the same characteristics such as path, resources, and security.
- Router support the handling of flow label table which has entry for each active flow. When a router receives a packet it check the flow label table for the entry and provides the service mentioned. The information is provided by other means such as the hop-by-hop options or other protocols.
- Flow label objective is to speed up the processing of packet i.e. while receiving a packet instead checking the routing table it consults the flow label table to find the next hop.
- A flow label used to support the transmission of real-time audio and video. Real-time audio or video, particularly in digital form

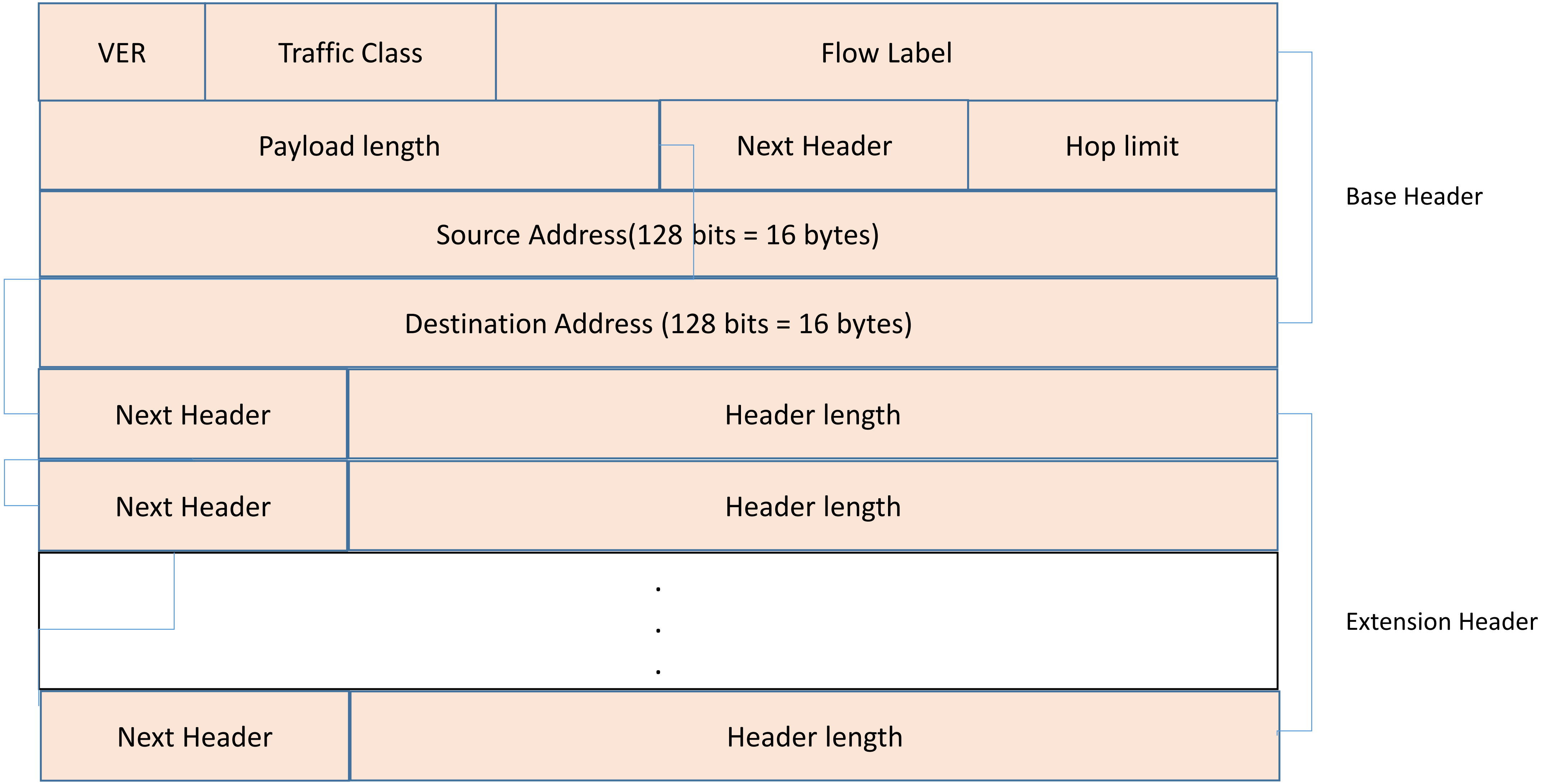
Rules for flow label:

- Flow label is defined by source host which takes a random number form 1 to 2^{24} -1 and it must be unique one.
- If host doesn’t support the feature its set to zero and if router doesn’t support it simply ignore the field.
- All the packets belong to the same flow has same source, same destination, same priority and same option

IPV4 vs IPV6

IPV4	IPV6
Header length Field	Header length field is eliminated
Service type field	Traffic class and flow label
Total length field	Payload length field
Identification , flag, offset	Fragmentation extension header
TTL	Hop-limit
Protocol	Next Header
Header Checksum	No Checksum
Option fields	Extension Headers

Extension Header



VER | Traffic Class | Flow Label

Payload length | Next Header | Hop limit

Source Address(128 bits = 16 bytes)

Destination Address (128 bits = 16 bytes)

Next Header | Header length

Next Header | Header length

.
.
.

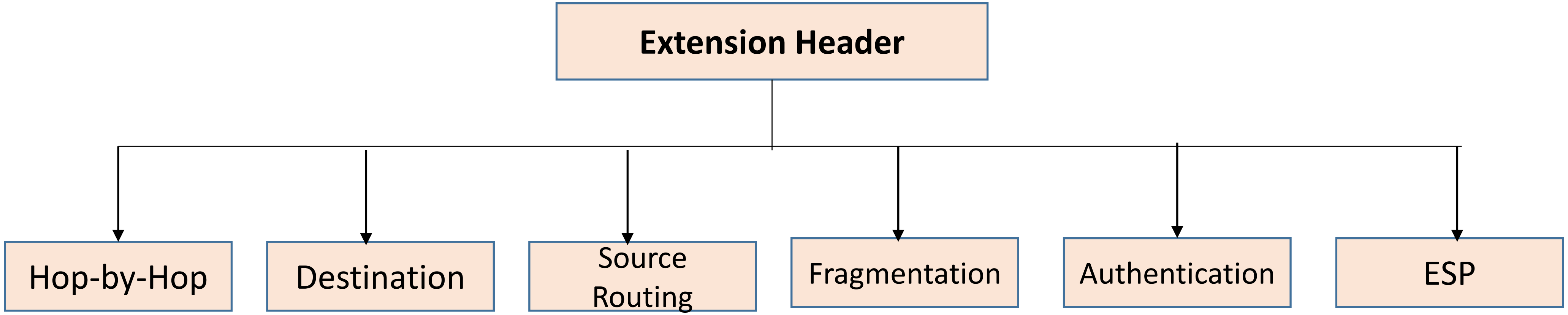
Next Header | Header length

Base Header

Extension Header

Format of Base Header

Extension Headers



Extension Header

Hop-by-Hop

Destination

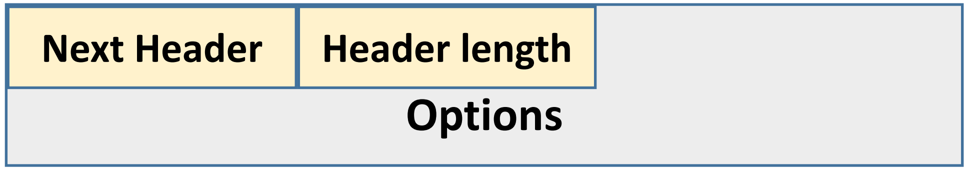
Source Routing

Fragmentation

Authentication

ESP

- Hop-by-Hop is used when source needs to send information to all routers along the path. Used to specify information such as management, debugging and control function also used too specify when datagram size exceeds 65535 bits. The first field defines the header and next fields defines the length and rest of the field makes options. Only 3 options have been defined are Pad1, PadN, and jumbo Payload.

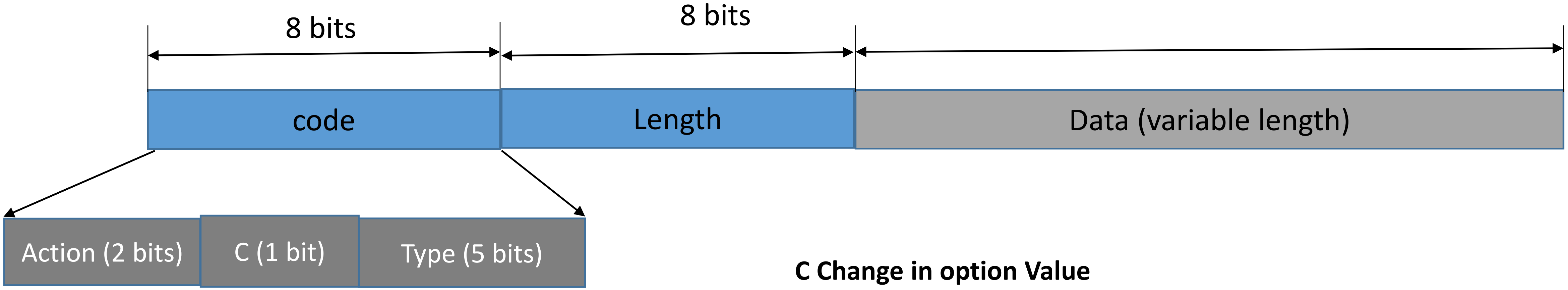


Next Header

Header length

Options

Options in Hop-by-Hop header



C Change in option Value

- 0 Doesn’t change in transit
- 1 maybe changed in transit

Action (if option not required)

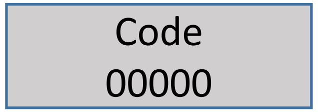
- 00 Skip this option
- 01 Discard datagram, no more action
- 10 Discard datagram and send ICMP message
- 11 Discard datagram and send ICMP message if not Multicast

Type

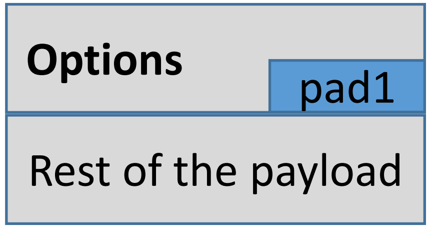
- 00000 Pad1
- 00001 PadN
- 00010 Jumbo Payload

Type

- Pad1 – used for alignment purpose. Options need to start at specific bit of 32 bit word, if it fall short by 1 bit Pad1 is used. Pad1 excludes the length of option and data field. Pad1 can be inserted anywhere in hop-by-hop

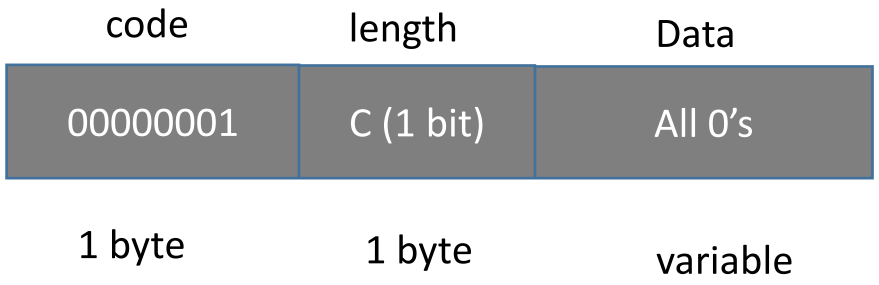


a. Pad1



b. Used for padding

Options in Hop-by-Hop header

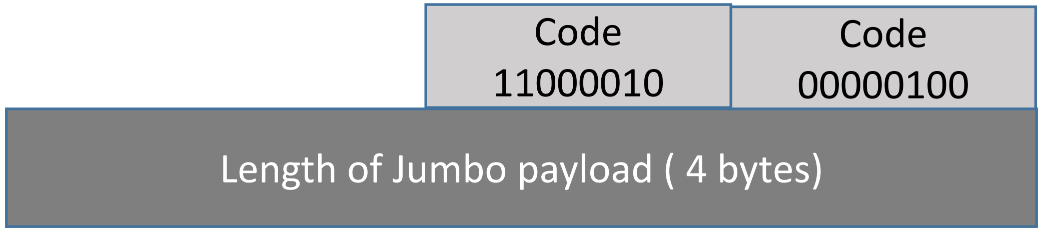


code	length	Data
00000001	C (1 bit)	All 0’s
1 byte	1 byte	variable

Pad N

Type

- PadN – PadN is used when 2 or more bytes are needed for alignment. PadN is made of 1 byte of option code, 1 byte of the option length, and a variable number of zero padding bytes. The value of the option code is 1 (action is 00, the change bit is 0, and type is 00001). The option length contains the number of padding bytes



Code 11000010

Code 00000100

Length of Jumbo payload (4 bytes)

Jumbo Payload

Type

- **Jumbo payload.** Payload in the IP datagram can be a maximum of 65,535 bytes in length, if for any reason payload length is larger than prefer jumbo payload option to define this longer length. The jumbo payload option must always start at a multiple of 4 bytes plus 2 from the beginning of the extension headers. The jumbo payload option starts at the (4n + 2) byte, where n is a small integer.

Extension Headers

Destination: Destination is used when the source needs to send information only to the destination and prevents the information get accessed by the router along the path. The format of the destination field is same as hop-by-hop option.

Source Routing: Combines the concept of strict and loose routing and minimum of 7 fields length.

Next Header	Header Length	Code	Address left
Reserved	Strict/loose mask		
First Address			
Second Address			
: :			
Last address			

- Next Header and Header length are same as in Hop-by-Hop
- Type field defines strict or loose routing.
- Address left represents the host need to reach the destination.
- strict/loose mask field determines the rigidity of routing.
  - If set to strict, routing must follow the source specification
  - If set to loose, it may visit other routers

Note : Destination address in source routing does not refer to the final destination of the datagram instead, it changes from router to router. The addresses in the extension headers also change from router to router.

Source Routing Example

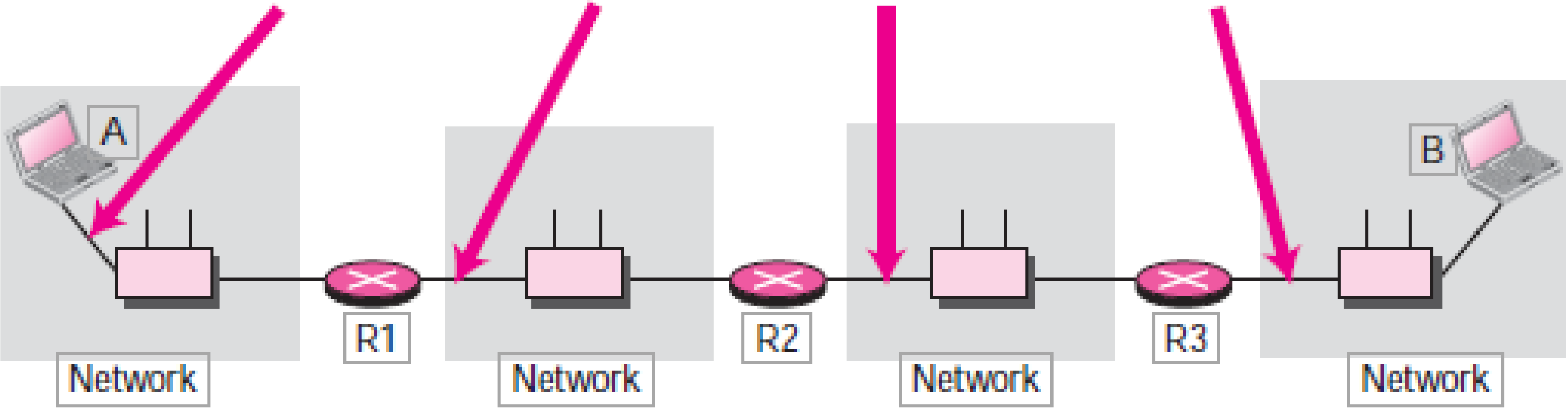
If source **A** wants to send a packet to destination B then, the routing info

Source : A
Destination : R1
Left : 3
R2
R3
B

Source : A
Destination : R2
Left : 2
R1
R3
B

Source : A
Destination : R3
Left : 1
R1
R2
B

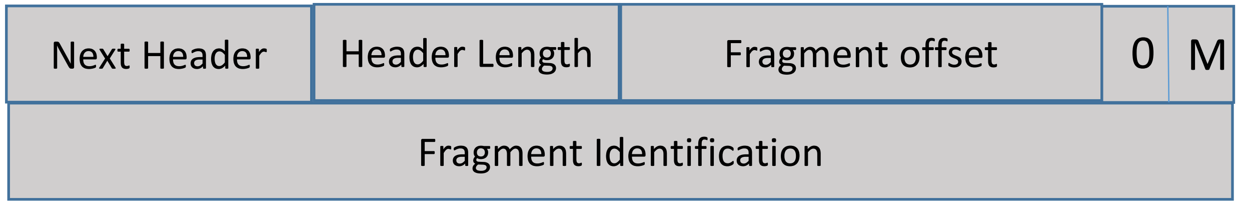
Source : A
Destination : R3
Left : 0
R1
R2
R3



Extension Header – Fragmentation & Authentication

Fragmentation

- Refers to the process of breaking the segment into smaller fragment
- IPV4 either the source (host) or the router performs the fragmentation process based on the MTU (Maximum Transmission Unit)



Next Header

Header Length

Fragment offset

0

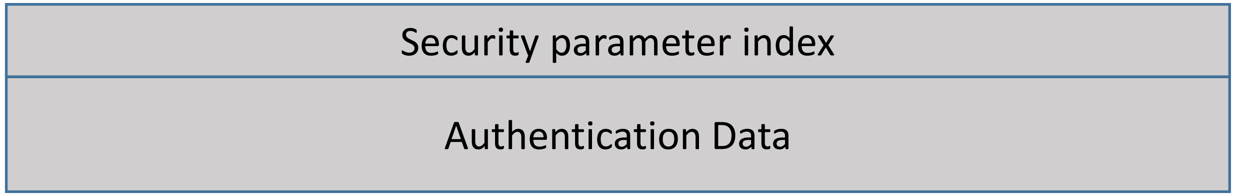
M

Fragment Identification

Fragmentation

- In IPV6 only the source performs the fragmentation by using Path MTU discovery technique in-order to find the smallest MTU on the path. If the source doesn't use MTU discovery, it fragment the packet into size of 1280 bytes or smaller

Authentication:



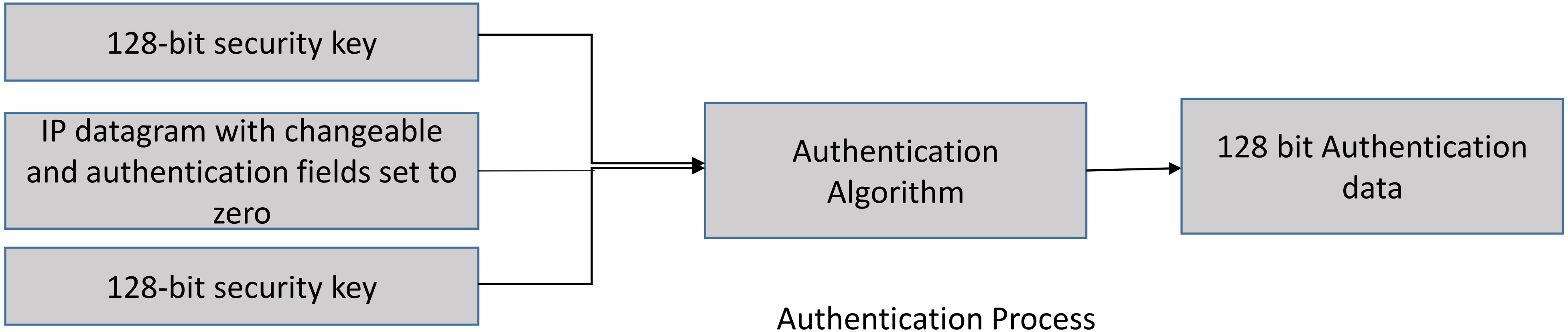
Security parameter index

Authentication Data

Authentication

- Validates the sender and ensure integrity of data. Validation of sender to insure the message comes for genuine source not from intruder. The integrity is verify the original transmitted message reaches the receiver end.
- Security parameter index defines the algorithm used for authentication and the data field contain the actual data generated by the algorithm.

Authentication of data


128-bit security key
IP datagram with changeable and authentication fields set to zero
128-bit security key
Authentication Algorithm
128 bit Authentication data

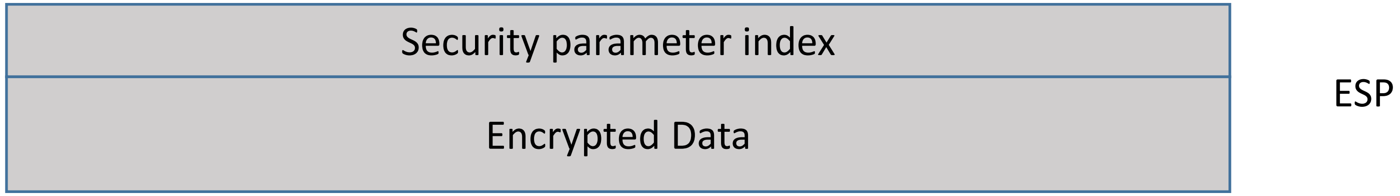
Authentication Process

The sender passes a 128-bit security key, the entire IP datagram, and the 128-bit security key again to the algorithm. Those fields in the datagram with values that change during transmission (for example, hop count) are set to zero. The datagram passed to the algorithm includes the authentication header extension, with the authentication data field set to zero. The algorithm creates authentication data which is inserted into the extension header prior to datagram transmission.

The receiver on receiving the message passes the datagram and the secret to the algorithm and compare the result, if it matches the datagram is accepted else discarded.

Extension Header – ESP

Encrypted Security Payload (ESP)



Security parameter index

Encrypted Data

ESP

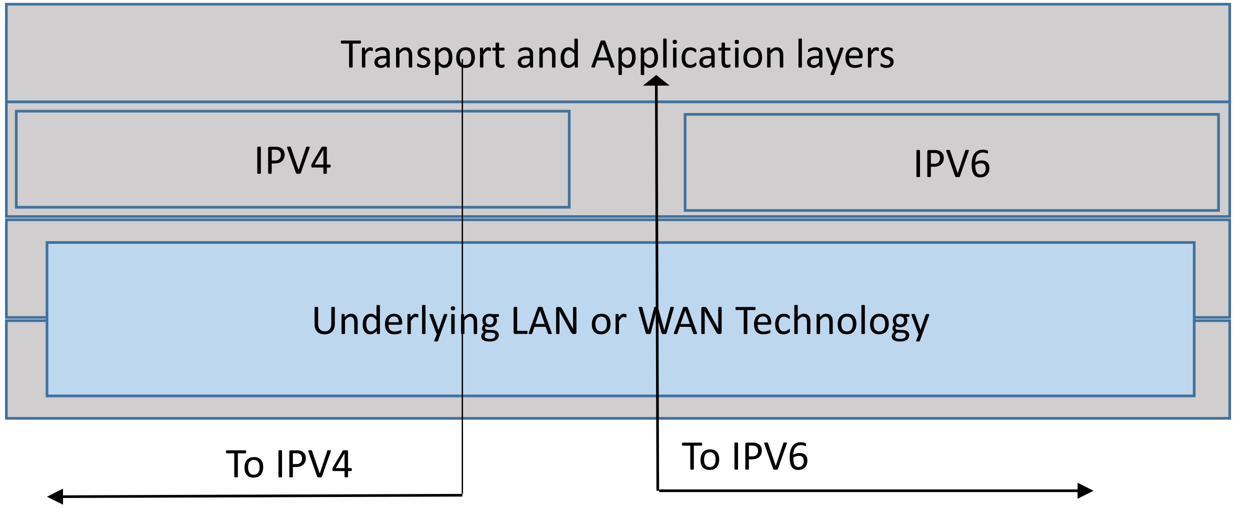
- To provide confidentiality and prevent eavesdropping.
- The security parameter index defines the type of encryption algorithm used and data field carries the encrypted data and other information if any needed for the algorithm
- Encryption can be done either by transport model or tunnel model

Transition from IPV4 to IPV6

Three strategies used are:

- Dual stack
- Tunnelling
- Header translation

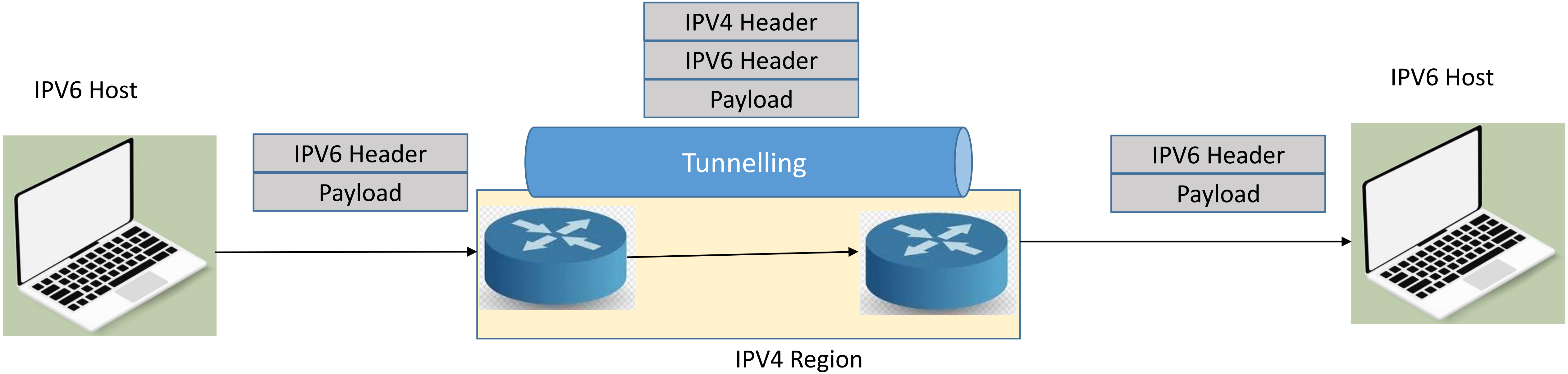
Dual Stack: Before complete migration all station must run in dual mode i.e. Both IPV4 and IPV6


Transport and Application layers
IPV4
IPV6
Underlying LAN or WAN Technology
To IPV4
To IPV6

Before sending a packet to the destination the source queries the DNS, if it returns IPV4 then source sends IPV4 packet else send IPV6 packet.

Transition from IPV4 to IPV6 : Tunnelling

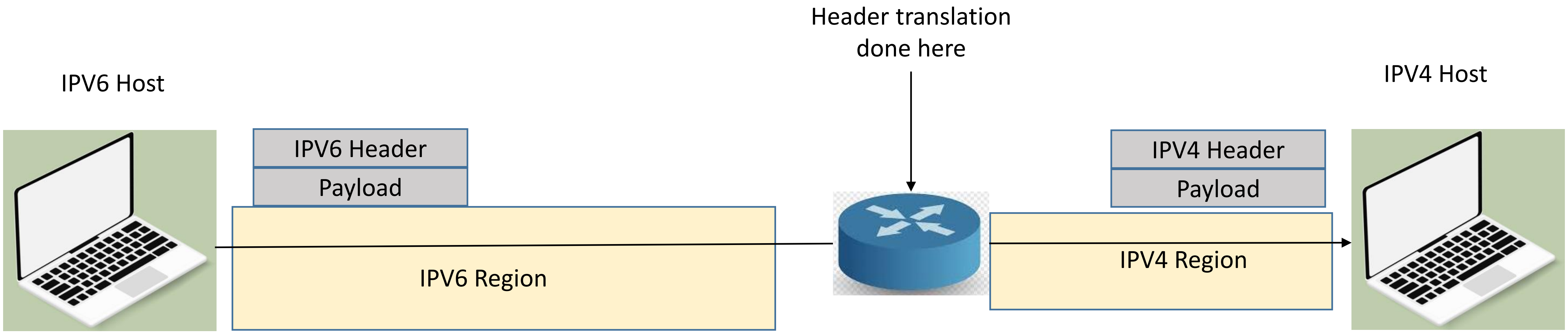
Tunnelling : Process happens when two IPV6 host wants to communicate through a IPV4 Channel, to pass through this channel it requires a IPV4 address. So IPV6 packet is encapsulated in a IPV4 packet and enter the region.



IPv4 packet is carrying an IPv6 packet as data, the protocol value is set to 41

Transition from IPV4 to IPV6 : Header translation

Header Translation acquires when sender uses a IPV6 and receiver uses IPV4, where the IPV6 address needs to be translated to IPV4.



Header translation uses the mapped address to translate an IPv6 address to an IPv4 address. Rules for Translation:

The IPv6 mapped address is changed to an IPv4 address by extracting the right-most 32 bits.

The value of the IPv6 priority field is discarded. The type of service field in IPv4 is set to zero.

The checksum for IPv4 is calculated and inserted in the corresponding field. The IPv6 flow label is ignored.

Compatible extension headers are converted to options and inserted in the IPv4 header. Some may have to be dropped.

The length of IPv4 header is calculated and inserted into the corresponding field.

The total length of the IPv4 packet is calculated and inserted in the corresponding field.

NAT Protocol

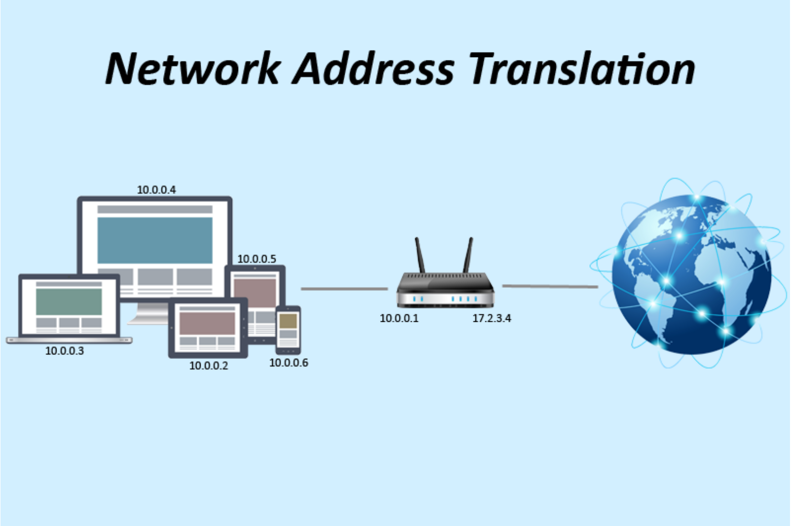
What is NAT????

To access the Internet one public IP address is needed but we can use a private IP address in our private network. The idea of NAT is to allow multiple devices to access the Internet through a single public address. To achieve this the translation of private IP address to a public IP address is required. Network Address Translation (NAT) is a process in which one or more local IP address is translated into one or more Global IP address and vice versa in order to provide Internet access to the local hosts.

NAT Working

Generally, the border router is configured for NAT i.e the router which has one interface in local (inside) network and one interface in the global (outside) network. When a packet traverse outside the local (inside) network, then NAT converts that local (private) IP address to a global (public) IP address. When a packet enters the local network, the global (public) IP address is converted to a local (private) IP address.

If NAT run out of addresses, i.e., no address is left in the pool configured then the packets will be dropped and an Internet Control Message Protocol (ICMP) host unreachable packet to the destination is sent.



Types of NAT :

There are 3 ways to Configure NAT :

- Static NAT – In this, a single unregistered (Private) IP address is mapped with a legally registered (Public) IP address i.e one-to-one mapping between local and global address.
- Dynamic NAT – In this type of NAT, an unregistered IP address is translated into a registered (Public) IP address from a pool of public IP address.
- Port Address Translation (PAT) – This is also known as NAT overload. In this, many local (private) IP addresses can be translated to a single registered IP address. Port numbers are used to distinguish the traffic i.e., which traffic belongs to which IP address.

Advantages of NAT

- NAT conserves legally registered IP addresses .
- It provides privacy as the device IP address, sending and receiving the traffic, will be hidden.
- Eliminates address renumbering when a network evolves.

Disadvantages of NAT

- Translation results in switching path delays.
- Certain applications will not function while NAT is enabled.
- Complicates tunneling protocols such as IPsec.
- Also, router being a network layer device, should not tamper with port numbers(transport layer) but it has to do so because of NAT.

IPV6 Mobility

IPV6 Mobility

- When a host is connected to a link or network, it acquires an IP address and all communication take place using that IP address on that link. As soon as, the same host changes its physical location, that is, moves into another area / subnet / network / link, its IP address changes accordingly, and all the communication taking place on the host using old IP address, goes down.
- IPv6 mobility provides a mechanism for the host to roam around different links without losing any communication/connection and its IP address.

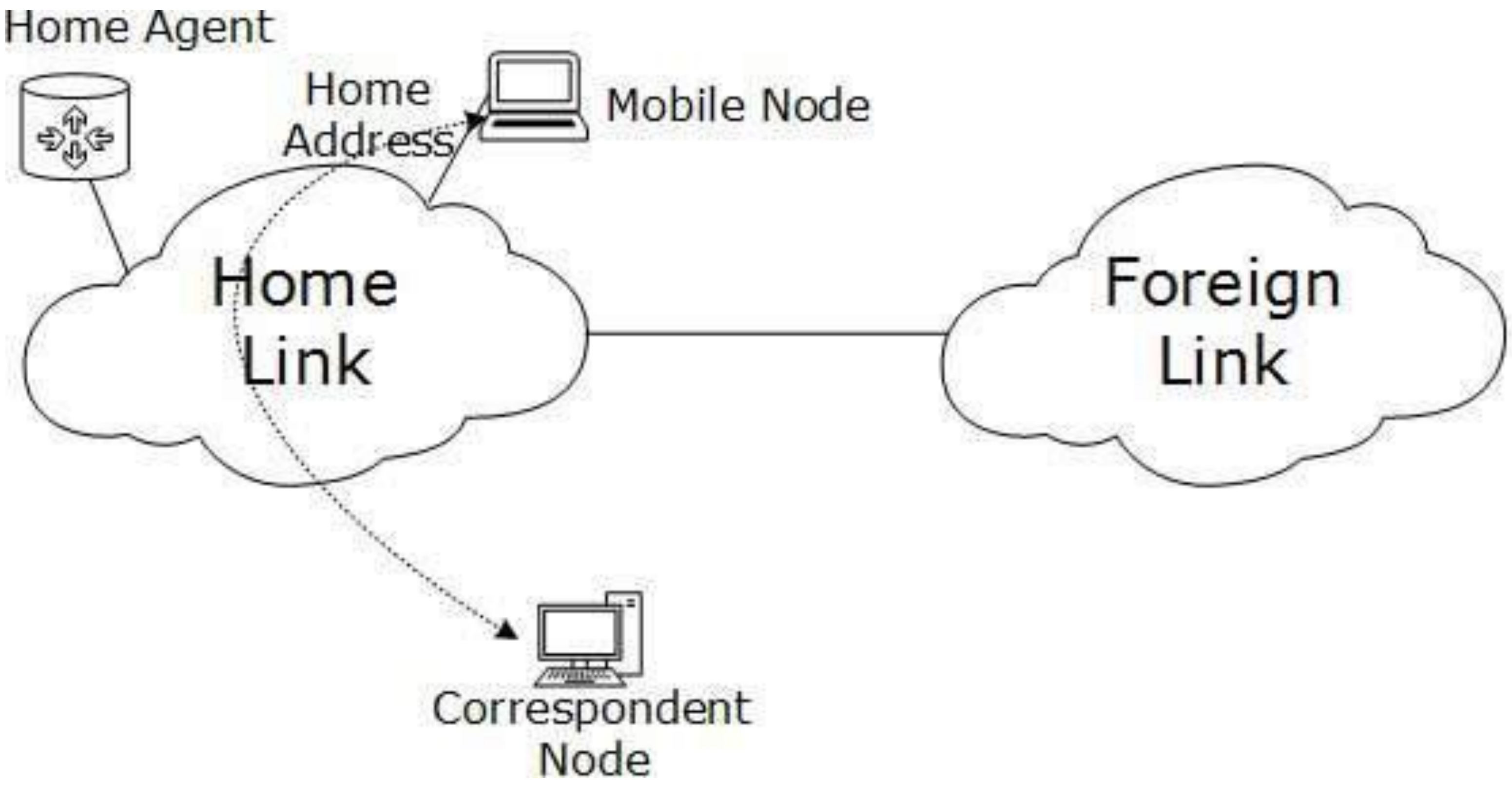
Modules associated

- Mobile Node: The device that needs IPv6 mobility.
- Home Link: This link is configured with the home subnet prefix and this is where the Mobile IPv6 device gets its Home Address.
- Home Address: This is the address which the Mobile Node acquires from the Home Link. This is the permanent address of the Mobile Node. If the Mobile Node remains in the same Home Link, the communication among various entities take place as usual.
- Home Agent: This is a router that acts as a registrar for Mobile Nodes. Home Agent is connected to Home Link and maintains information about all Mobile Nodes, their Home Addresses, and their present IP addresses.

Modules associated

- Foreign Link: Any other Link that is not Mobile Node’s Home Link.
- Care-of Address: When a Mobile Node gets attached to a Foreign Link, it acquires a new IP address of that Foreign Link’s subnet.
- Correspondent Node: Any IPv6 enabled device that intends to have communication with Mobile Node.

Diagramatic Representation



IPV6 Mobility Operations

When a Mobile Node leaves its Home Link and is connected to some Foreign Link, the Mobility feature of IPv6 comes into play. After getting connected to a Foreign Link, the Mobile Node acquires an IPv6 address from the Foreign Link. This address is called Care-of Address. The Mobile Node sends a binding request to its Home Agent with the new Care-of Address. The Home Agent binds the Mobile Node’s Home Address with the Care-of Address, establishing a Tunnel between both.

Whenever a Correspondent Node tries to establish connection with the Mobile Node (on its Home Address), the Home Agent intercepts the packet and forwards to Mobile Node’s Care-of Address over the Tunnel which was already established.

Route Optimization

When a Correspondent Node initiates a communication by sending packets to Mobile the Node on the Home Address, these packets are tunneled to the Mobile Node by the Home Agent. In Route Optimization mode, when the Mobile Node receives a packet from the Correspondent Node, it does not forward replies to the Home Agent. Rather, it sends its packet directly to the Correspondent Node using Home Address as Source Address.

Protocols Changed to Support IPV6

ICMPv6

Internet Control Message Protocol version 6 is an upgraded implementation of ICMP to accommodate IPv6 requirements. This protocol is used for diagnostic functions, error and information message, statistical purposes. ICMPv6’s Neighbor Discovery Protocol replaces ARP and helps discover neighbor and routers on the link.

DHCPv6

Dynamic Host Configuration Protocol version 6 is an implementation of DHCP. Though IPv6 enabled hosts do not require any DHCPv6 Server to acquire IP address as they can be auto-configured. Neither do they need DHCPv6 to locate DNS server because DNS can be discovered and configured via ICMPv6 Neighbor Discovery Protocol. Yet DHCPv6 Server can be used to provide these information.

DNS

There has been no new version of DNS but it is now equipped with extensions to provide support for querying IPv6 addresses. A new AAAA (quad-A) record has been added to reply IPv6 query messages. Now DNS can reply with both IP versions (4 & 6) without any change in query format.