

# Introduction to IPv6



These materials are licensed under the Creative Commons *Attribution-Noncommercial 3.0 Unported* license (<http://creativecommons.org/licenses/by-nc/3.0/>)



# Problems with IPv4

- Address space limited and poorly distributed
  - IANA allocated the last blocks to the RIRs in January 2011!
  - This is actually THE most important problem
- Variable size header
- Not flexible for adding new extensions and options
- QoS is an add-on
- Authentication and privacy are add-ons

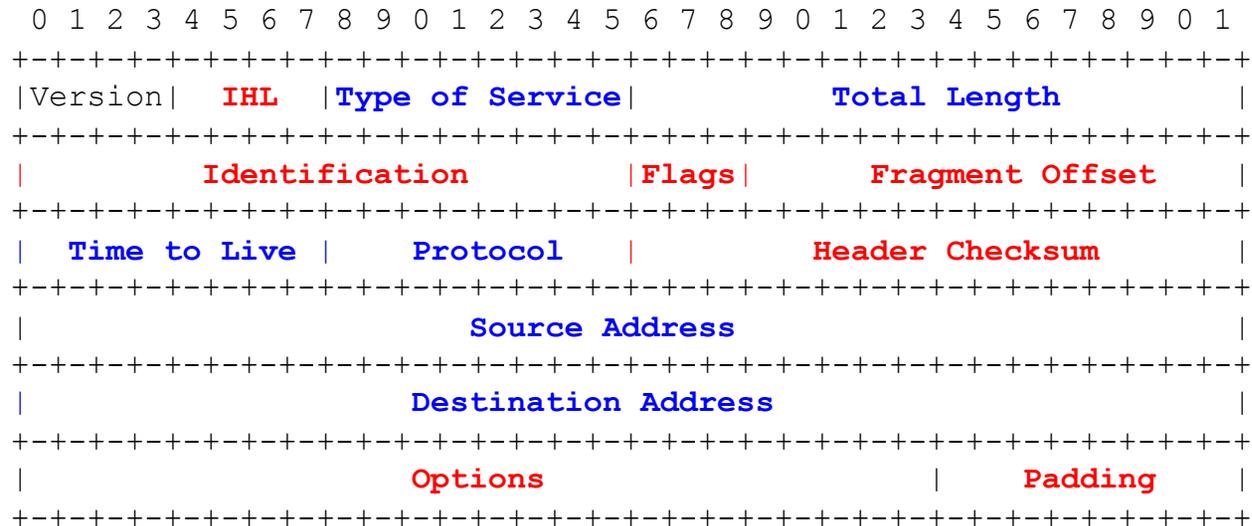
# Changes in Headers

- *Version*: No changes (4 or 6)
- *Header Length*: Not necessary since IPv6 header is of fixed length
- *Type of Service*: Renamed as *Traffic Class*
- *Flow Label*: Added for identifying flows
- *Total Length*: Now specifies payload length
- *Identification, Flags, Fragment Offset*: Functions moved to the fragmentation header extension

# Changes in Headers

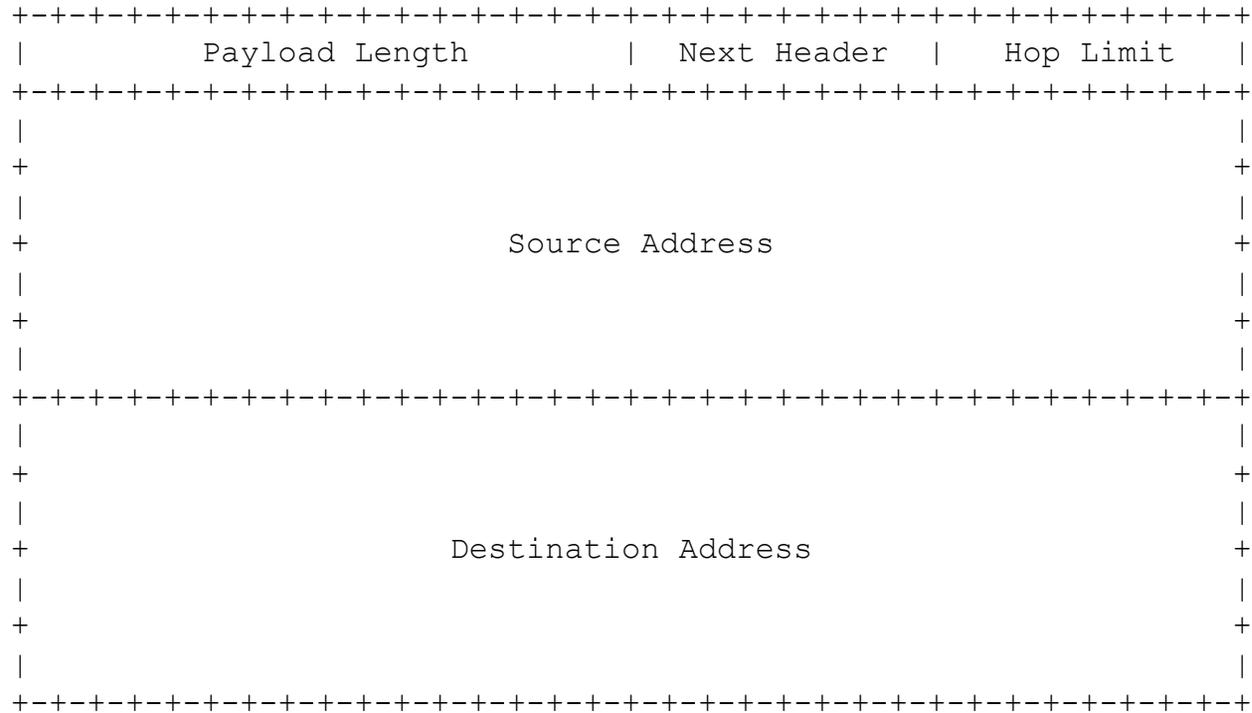
- *Time To Live*: Renamed as *Hop Limit* (makes more sense)
- *Protocol*: Now called *Next Header*
- *Header Checksum*: Removed for better efficiency
- *Source/Destination Address*: Now have 128 bits instead of 32
- *Options y Padding*: Removed

# IPv4 Header



**Removed Changed**

# IPv6 Header



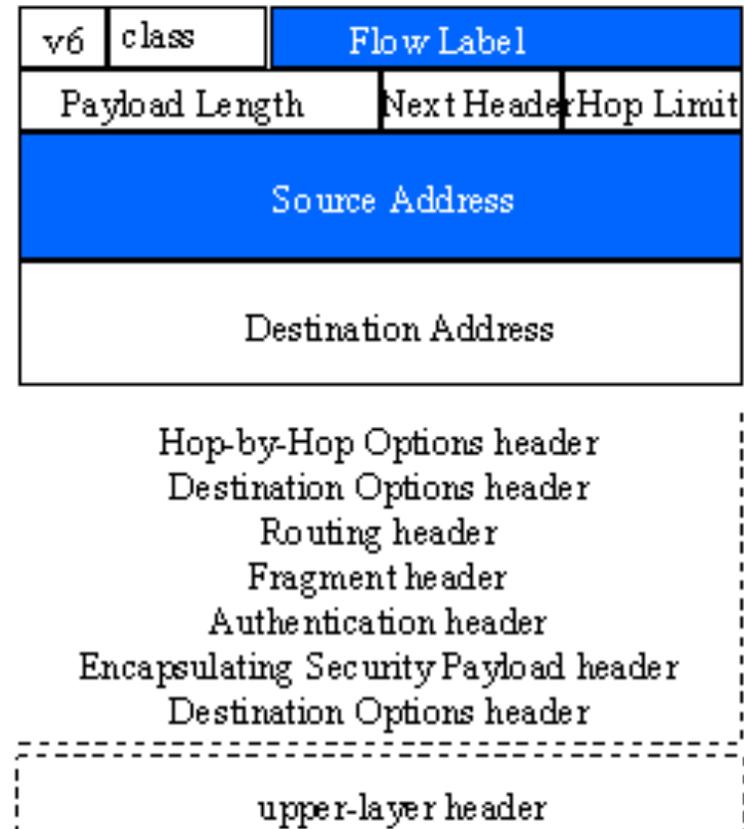
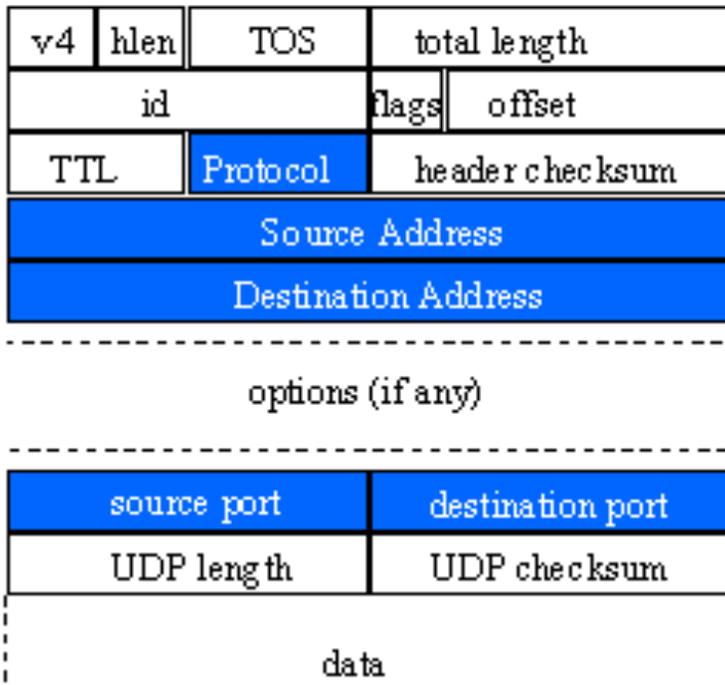


# New fields in IPv6 header

- Traffic Class (Priority)
- Flow label
  - All common packets must have same source, destination and flow label
  - Flow label can identify different flows within the same two endpoints
  - Handling of flows is determined by some control plane protocol, such as RSVP (Resource reSerVation Protocol)

# New fields in IPv6 header

Recognizing flows in IPv4 vs IPv6





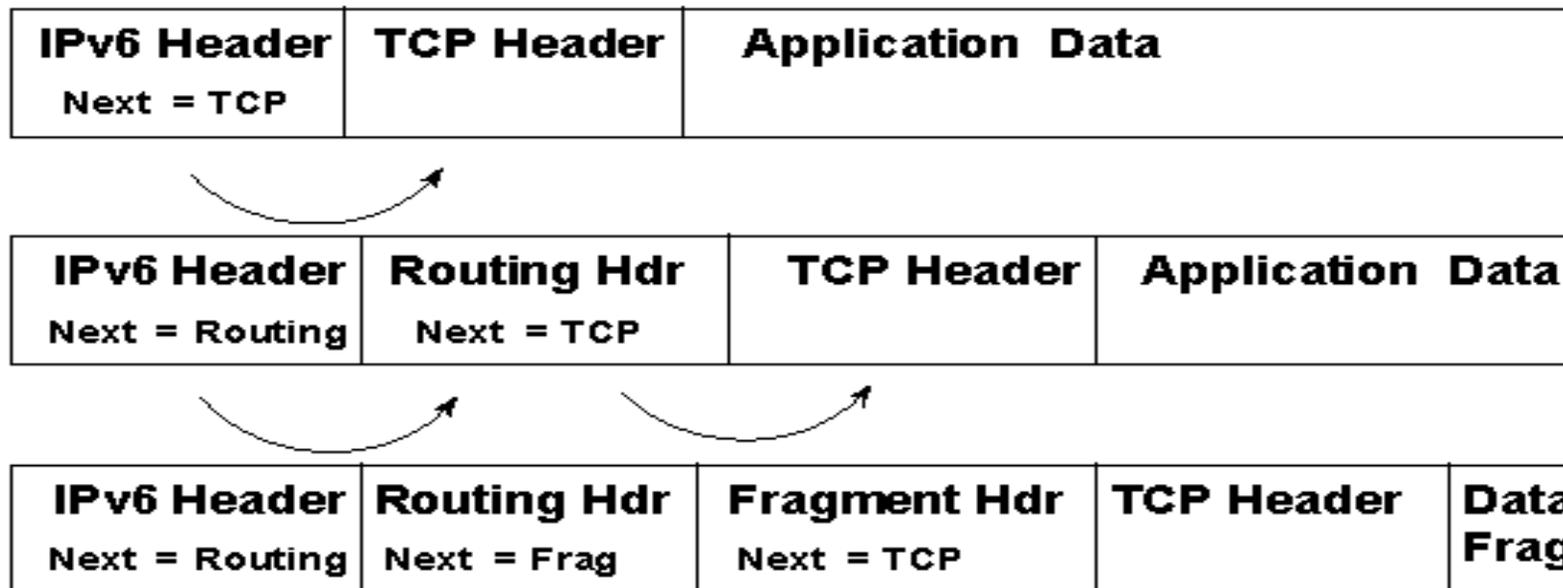
# New fields in IPv6 header

## Next Header

<b>Value</b>	<b>Header</b>
0	Hop-by-Hop Options
6	TCP
17	UDP
58	ICMPv6
60	Destination Options

# New fields in IPv6 header

Next-header Field

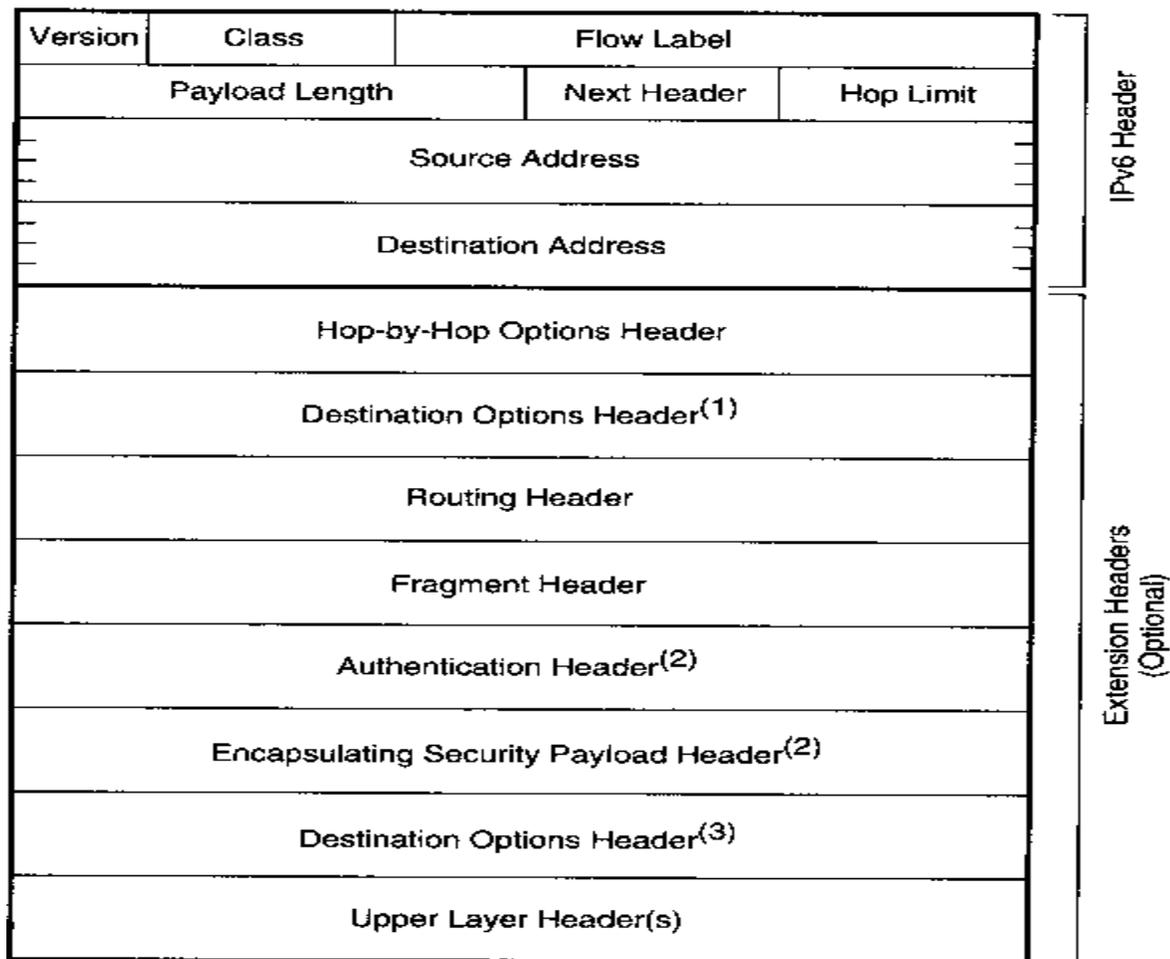




# Extension Headers

- Replace optional headers in IPv4
  - Options seldom used
- Lengths are multiples of 8 bytes (64 bits)
- An IPv6 implementation must support:
  - Hop-by Hop Options
  - Routing
  - Fragment
  - Destination
  - Authentication
  - Encapsulating Security Payload

# Extension Headers





# Extension Headers

## Hop-by-Hop Options Header

- Information analyzed by each node in the path (0)

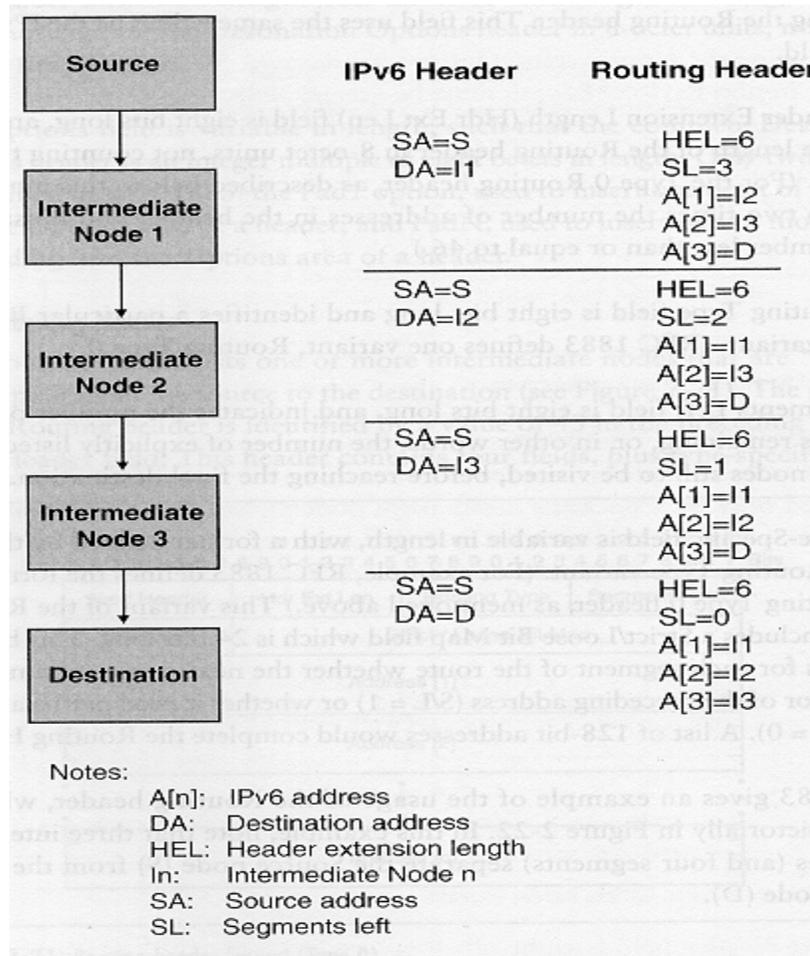
## Destination Options Header

- Optional information analyzed by the last node in the path (60)

## Routing Header

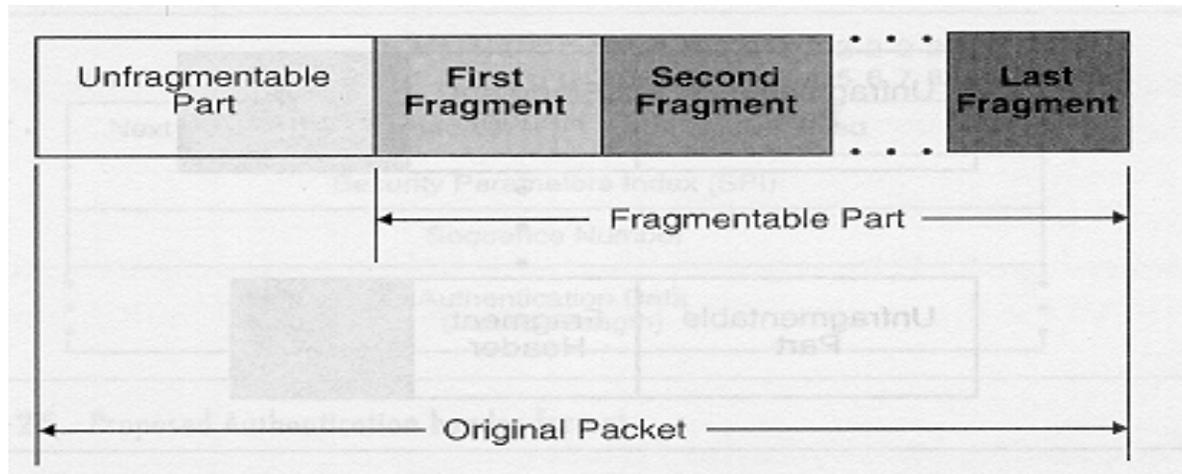
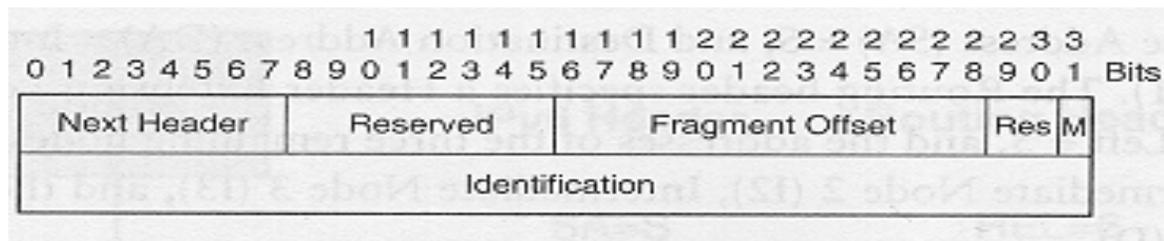
- Lists nodes to be visited on the way from the source to the destination (43)

# Routing Header



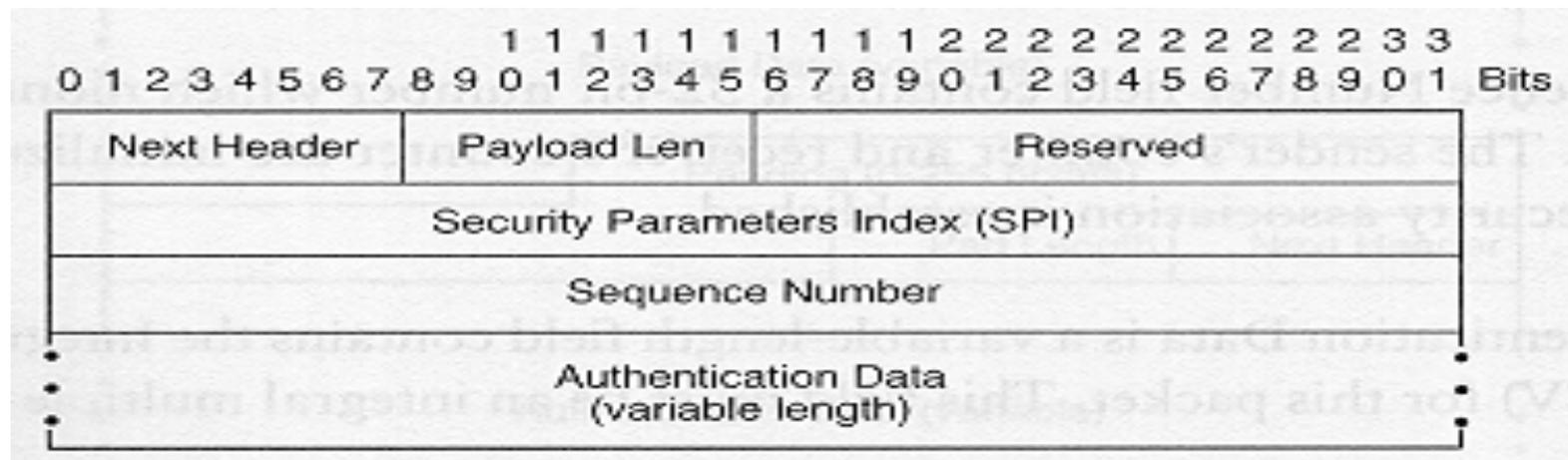
# Fragmentation Header

Used when sending packets larger than the MTU of the data link layer



# Authentication Header

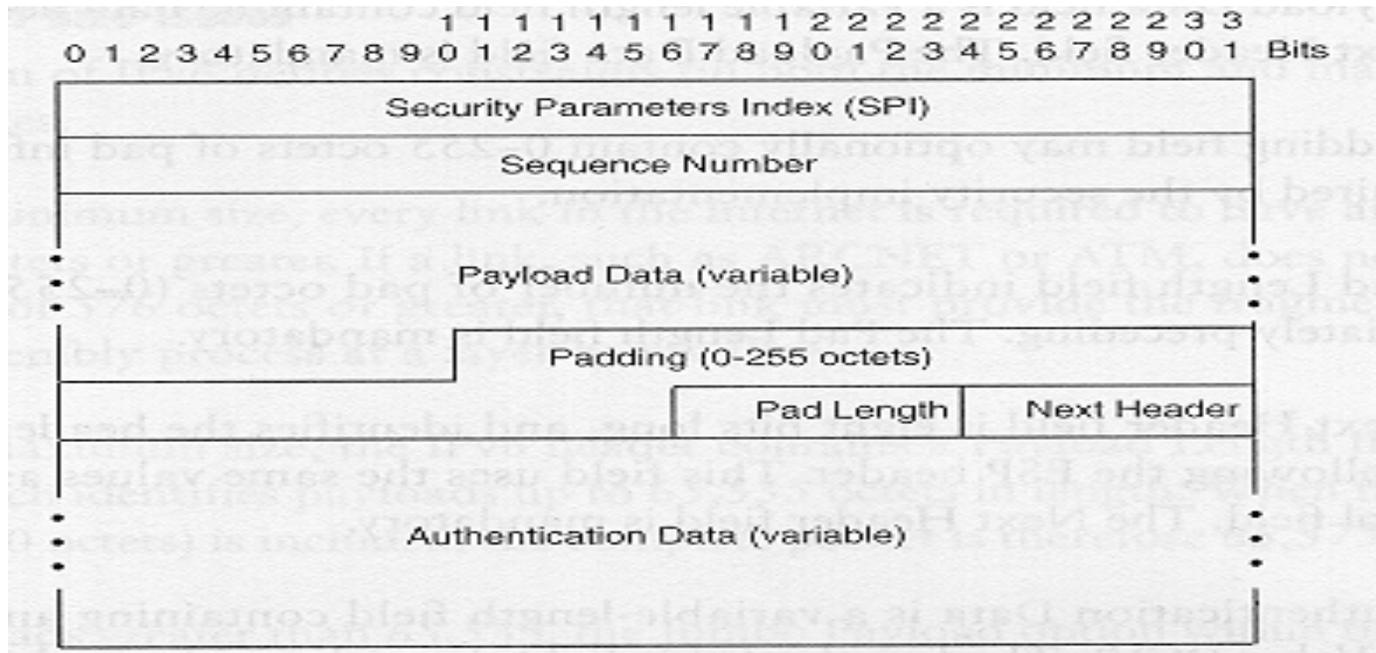
Provides authenticity and integrity (51)



# Encapsulating Security Payload (ESP)

Provides confidentiality (and, optionally, integrity, authentication and anti-replay) (50)

Can be used by itself, or in conjunction with AH





# Types of Delivery in IPv6

**Unicast:** To one host.

Destination address is unique

**Multicast:** To many hosts.

Destination address is shared

**Anycast:** To the nearest host.

Destination address is shared

# Interface identifiers

- 64 bits to identify one interface
- Must be unique within a subnet
- EUI-64: Formula to convert IEEE 802 MAC addresses (48 to 64 bits)
  - Example: MAC address 00:0A:95:F2:97:DB
  - Rules
    - Insert FF:FE between the 3rd and 4th bytes
    - Set the universal/local bit to 1 (second to last bit of first byte)
  - 02:0A:95:FF:FE:F2:97:DB



# IPv6 address structure

- IPv4 has 32 bits (4 bytes)
  - 130.192.1.143
- IPv6 has 128 bits (16 bytes)
  - 3FFE:0800:1200:300a:2A8:79FF:FE32:1982



# IPv6 address structure

- Sequence of 8 groups of 4 hexadecimal digits delimited by ‘:’
- Notations:
  - 1080:0000:0000:0000:0008:0800:200C:417A
  - 1080:0:0:0:8:800:200C:417A
  - 1080::8:800:200C:417A

# IPv6 address structure

## Notations

FF01:0:0:0:0:0:0:43	Multicast Address
0:0:0:0:0:0:0:1	Loopback Address
0:0:0:0:0:0:0:0	Unspecified Address

## Can be represented like this:

FF01::43	Multicast Address
::1	Loopback Address
::	Unspecified Address

# IPv6 address structure

- CIDR Notation

*IPv6-address/Prefix-length*

*IPv6-address:* Any of the previous notations

*Prefix-length:* Decimal number specifying the length of the prefix

- Example: 1080:0:0:8::/80



# Valid Notations

60 bits prefix 12AB00000000CD3:

- 12AB:0000:0000:CD30:0000:0000:0000:0000/60
- 12AB::CD30:0:0:0:0/60
- 12AB:0:0:CD30::/60



# Invalid Notations

**12AB:0000:0000:CD30:0000:0000:0000:0000/60**

12AB:0:0:CD3/60

Omitting zeros to the right is not valid within each group of 16 bits

12AB::

How to tell how many zeros were omitted on each side?

12AB::

Both errors, combined

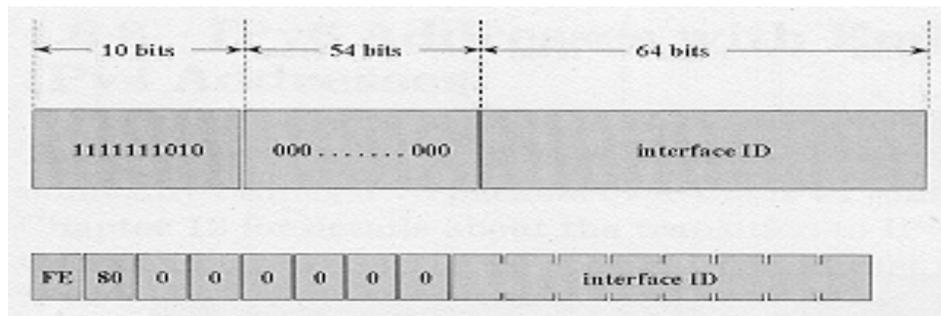


# Special Unicast Addresses

- Unspecified address
  - All zeros (::)
  - Used during initialization
  - Also used to represent the default route
- Loopback address
  - Only last bit set to 1 (:::1)
  - Similar to 127.0.0.1 in IPv4

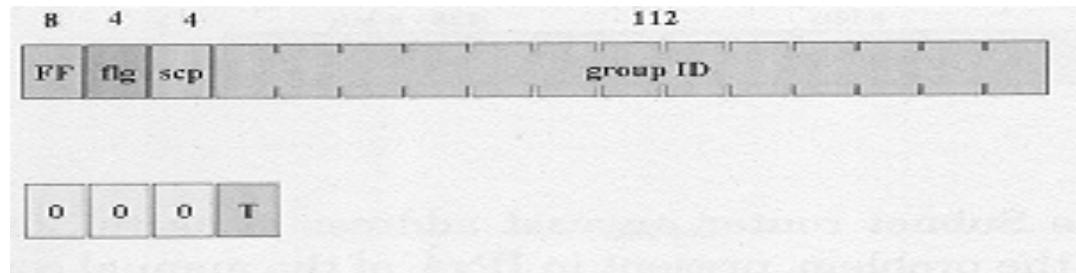
# Link Local addresses

- Used for auto-configuration and neighbor discovery  
1111111010 (FE)
- Only used within a physical segment
- Routers must not forward any packets with these addresses as either source or destination
- Example:
  - MAC = 08-00-02-12-34-56
  - IPv6 = FE80::A00:2FF:FE12:3456



# Multicast Addresses

1111 1111 (FF)



- T=0 permanent, T=1 not permanent
- SCP: limits the scope of the group
- Group ID: Identifies the multicast group



# Multicast Scope

0: Reservado

1: Interfaz-local

2: Enlace-local

3: Reservado

4: Admin-local

5: Sitio-local

6: No asignado

7: No asignado

8: Organización-local

9: No asignado

A: No asignado

B: No asignado

C: No asignado

D: No asignado

E: Global

F: Reservado



# Address Block Allocation

- RIRs (Regional Internet Registries) have two allocation schemes
  - Service providers obtain /32 blocks
  - Non service providers obtain /48 blocks
- In both schemes, the RIR reserves the next contiguous block in case that demand increases
- RIRs:
  - AFRINIC – Africa
  - APNIC – Asia/Pacific
  - ARIN – North America
  - LACNIC – Latin America and Caribbean
  - RIPE – Europe/Middle East



# Use at the U. Of Oregon

- 2001:468::/32 -> Internet2
  - 2001:468:0d00::/40 -> Oregon GigaPOP
    - 2001:468:0d01::/48 -> U. of Oregon



# Address Autoconfiguration

- Assumes that interfaces can provide a unique identifier
- Communication established using link-local addresses
- Facilitates renumbering when (if) changing providers



# Neighbor Discovery

## Types of Messages

- Router Advertisement
- Router Solicitation
- Routing Redirect
- Neighbor Solicitation
- Neighbor Advertisement

# Autoconfiguration

Generate a link-local address

Verify that this address is valid. Send neighbor solicitation to this address (DAD=Duplicate Address Detection)

If no answer, assign this address to the interface. Ready to communicate.

If address is being used:

Either configure manually or generate another address

Send a router solicitation message.

Router responds with router advertisement



# Autoconfiguration

Verify the value of the “managed address configuration” flag

If M=1, Must use stateful configuration for address assignment (e.g. DHCP)

If M=0, proceed with stateless configuration (SLAAC)

Verify the value of the “other stateful configuration” flag

If O=1, Must use stateful configuration for the other parameters

If O=0, Terminate the autoconfiguration process



# Autoconfiguration

- Stateless Autoconfiguration (SLAAC)
  - Routers must send periodic router advertisement (RA) messages to the all-hosts address
  - The host takes the IPv6 subnet prefix from the RA and uses that to build its own full IPv6 address
  - Only used for addresses



# Autoconfiguration

- Stateful Address configuration
  - When a more controlled mechanism is desired
  - Similar to what is done today with IPv4
  - DHCPv6 is the only option today



## Dual Stack - IPv6/IPv4

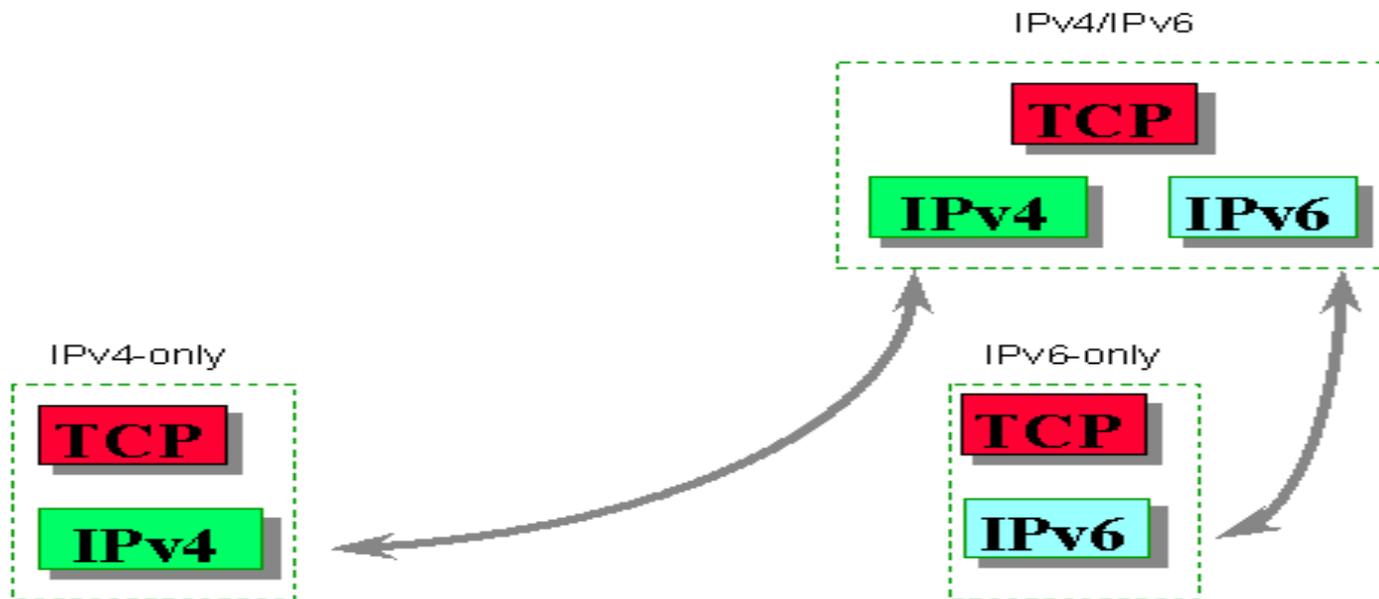
- Routers and hosts can communicate with others using either one
- Makes transition easier for end users



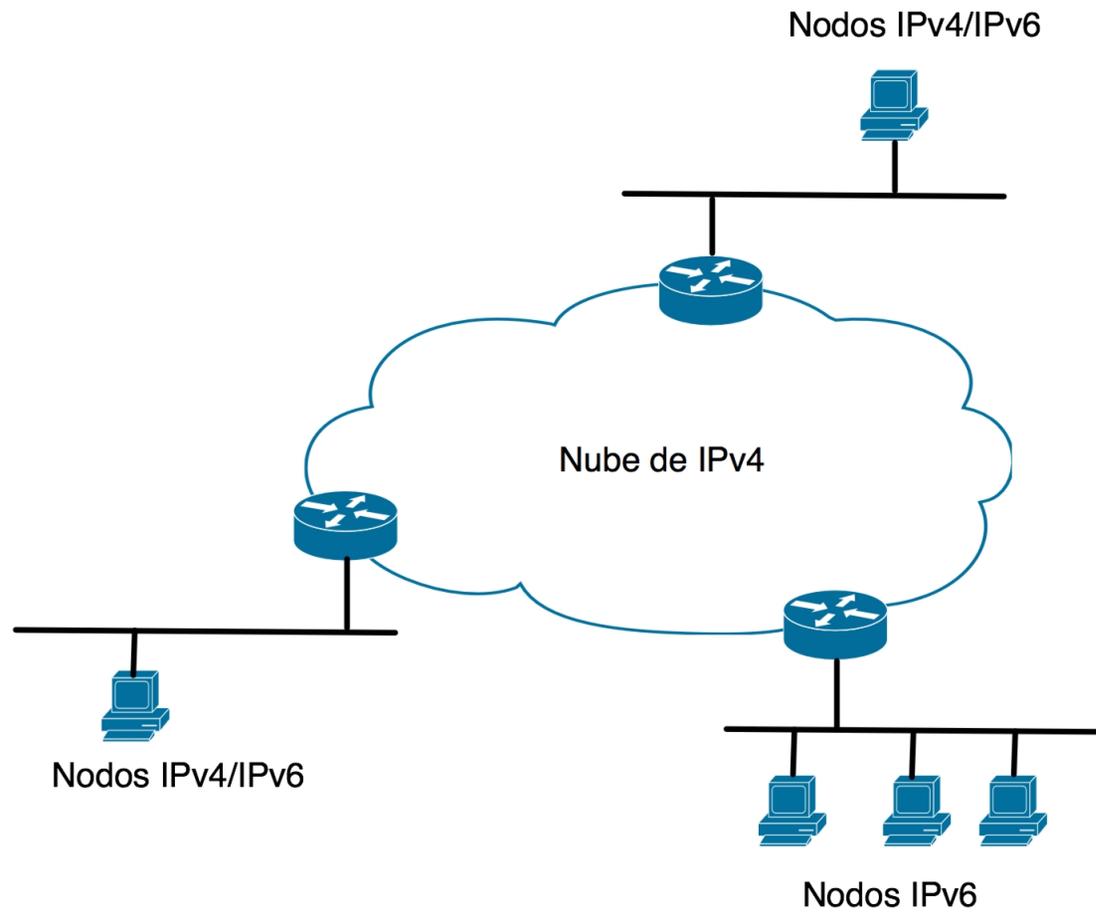
# Some Terminology

- Types of Nodes:
  - Pure IPv4 node (No IPv6)
  - Dual stack node (IPv6/IPv4) (Understands both versions)
  - Pure IPv6 node (No IPv4)

# Dual Stack Scheme



# Tunnels



# Multihoming

What happens when you have more than one provider?

- At first there was interest in distributing IPv6 blocks hierarchically to keep routing tables small
  - Did not work
- Today, the same strategy is used as with IPv4
  - Provider-Independent (PI) space

# DNS

- Not much change there
- Very important since it's almost impossible to remember an IPv6 address!
- Can use either IPv4 or IPv6 as transport
- Recent versions of ISC BIND support IPv6

# DNS

- Uses AAAA records to assign IPv6 addresses to names
- You can use both A and AAAA records with the same name
- Inverse resolution uses *ip6.arpa*
  - Replaces *in-addr.arpa*

# DHCPv6

- Available in ISC DHCPD starting with version 4
- Some notable differences with IPv4 DHCP
  - DHCPv6 does not assign the default gateway
    - Leaves this function to routers (RAs)
  - No mechanism for load balancing or failover yet
- Status of support by different operating systems:
  - Supported by Linux and Windows (Vista and later)
  - Supported by Mac OS X (Lion and later)

# References

## Web Sites:

Cisco Systems (<http://www.cisco.com/ipv6>)

<http://www.ipv6.org>

<http://www.6bone.net>

<http://www.ipv6forum.org>

<http://playground.sun.com/pub/ipng/html/ipng-main.html>

## Books:

Internetworking IPv6 with Cisco Routers

IPv6 for Cisco IOS

## IETF & RFCs

<http://www.ietf.org/html.charters/ipv6-charter.html>

<http://www.ietf.org/html.charters/multi6-charter.html>

<http://www.ietf.org/html.charters/v6ops-charter.html>