IPv6

A new generation of IP to meet the request of all IP world

IPv6 RFCs

- RFC1752 - Recommendations for the IP Next Generation Protocol
- RFC2460 - Overall specification
- RFC2373 - addressing structure
- others
- www.rfc-editor.org

IPv6 Design Issues

- Overcome IPv4 scaling problem
  - lack of address space.
- Flexible transition mechanism.
- New routing capabilities.
- Quality of Service.
- Security.
- Ability to add features in the future.

IPv6 Enhancements (1/2)

- Expanded address space
  - 128 bit ($\fallingdotseq 3.4 \times 10^{38}$)
- Improved option mechanism
  - Separate optional headers between IPv6 header and transport layer header
  - Most are not examined by intermediate routers
    - Improved speed and simplified router processing
    - Easier to extend options
- Address autoconfiguration
  - Dynamic assignment of addresses

IPv6 Enhancements (2/2)

- Increased addressing flexibility
  - Anycast - delivered to one of a set of nodes
  - Multicast - Improved scalability of addresses
- Support for resource allocation
  - Replace type of service
  - Labeling of packets to particular traffic flow
  - Allows special handling

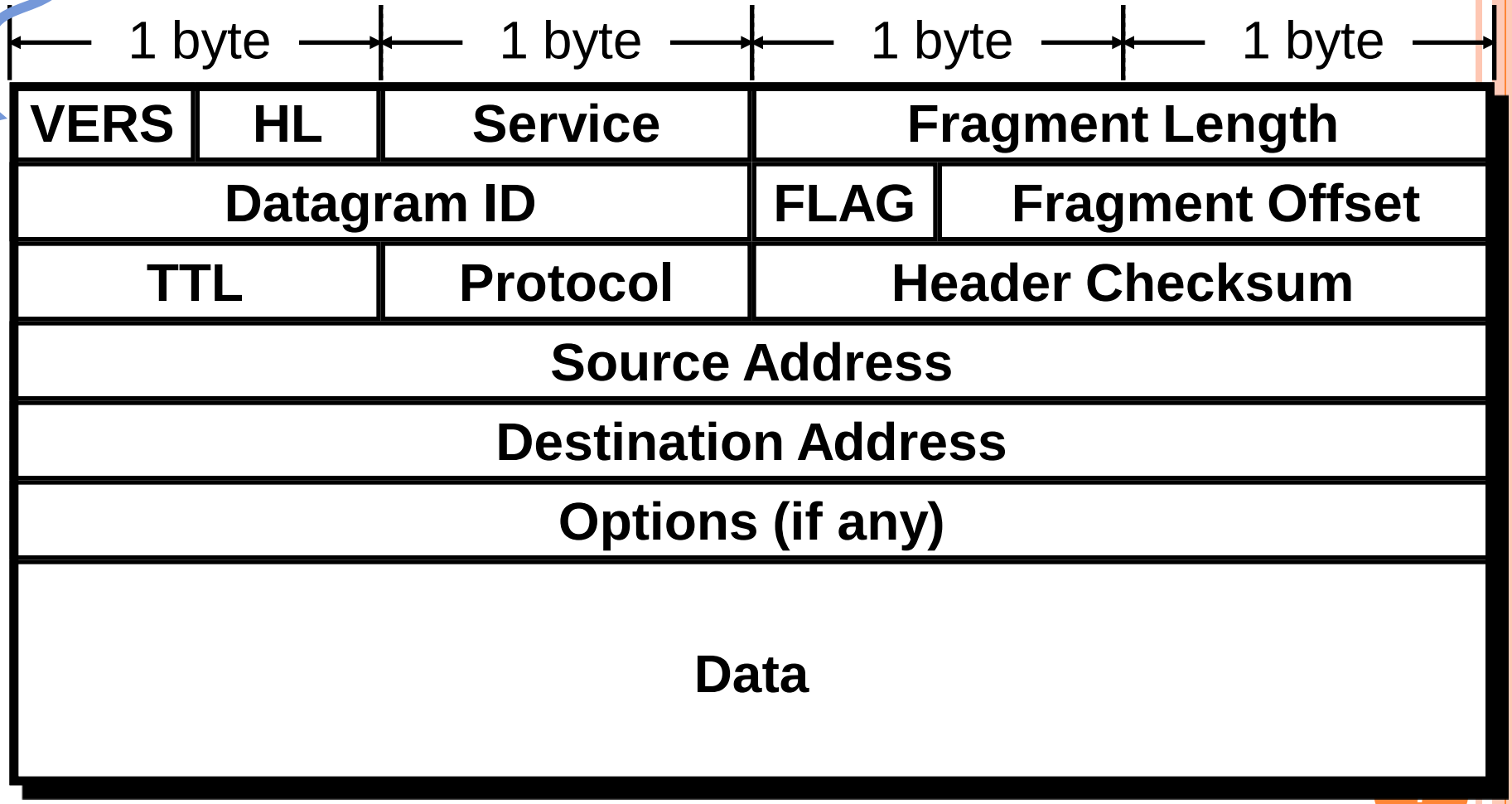
  e.g. real time video

IPv6 Headers

- Simpler header - faster processing by routers.
  - No optional fields - fixed size (40 bytes)
  - No fragmentation fields. (only hosts can fragment a packet, routers may not.)
  - No checksum
- Support for multiple (optional) headers
  - more flexible than simple “protocol” field.

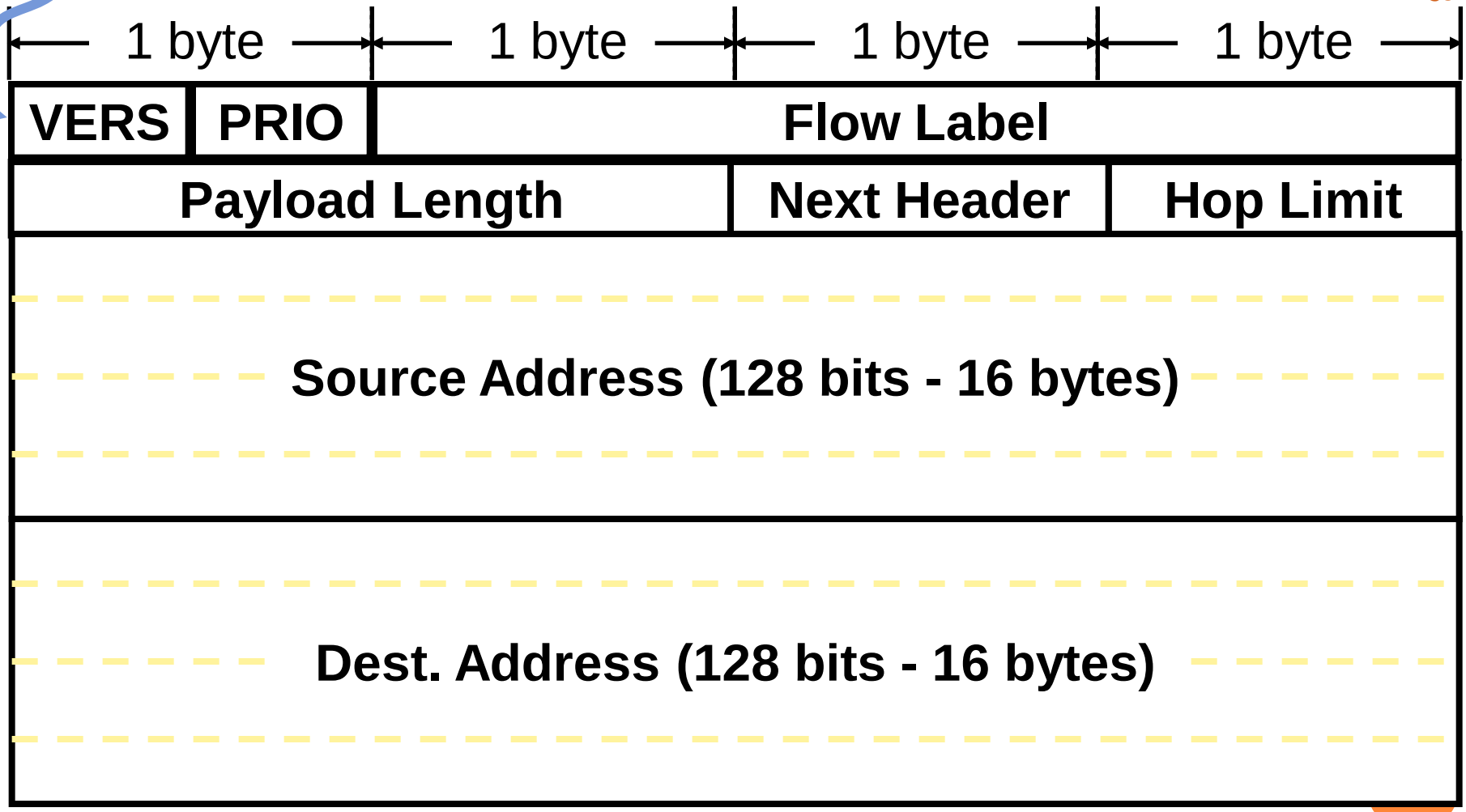
4 for IPv4

IPv4 Header



6 for IPv6

IPv6 Header



IPv6 Header Fields -1 (8 vs 12)

- VERS: 6 (IP version number)
- Priority: will be used in congestion control (traffic class) 6-bit of DS and 2-bit of ECN
- Flow Label: (20-bit) experimental - sender can label a sequence of packets as being in the same flow.
- Payload Length: number of bytes in everything following the 40 byte header, or 0 for a *Jumbogram*. Length of extension headers + transport-layer PDU.

IPv6 Header Fields - 2

- Next Header is similar to the IPv4 "protocol" field - indicates what type of header follows the IPv6 header.

- Hop Limit is similar to the IPv4 TTL field (but now it really means hops, not time).

Flow Label (1/3)

- A Flow is defined as a sequence of packets for which the source desires special handling by the intervening routers.

- A flow is uniquely identified by the combination of a source address, destination address, and a non-zero 20-bit flow label.

Flow Label (2/3)

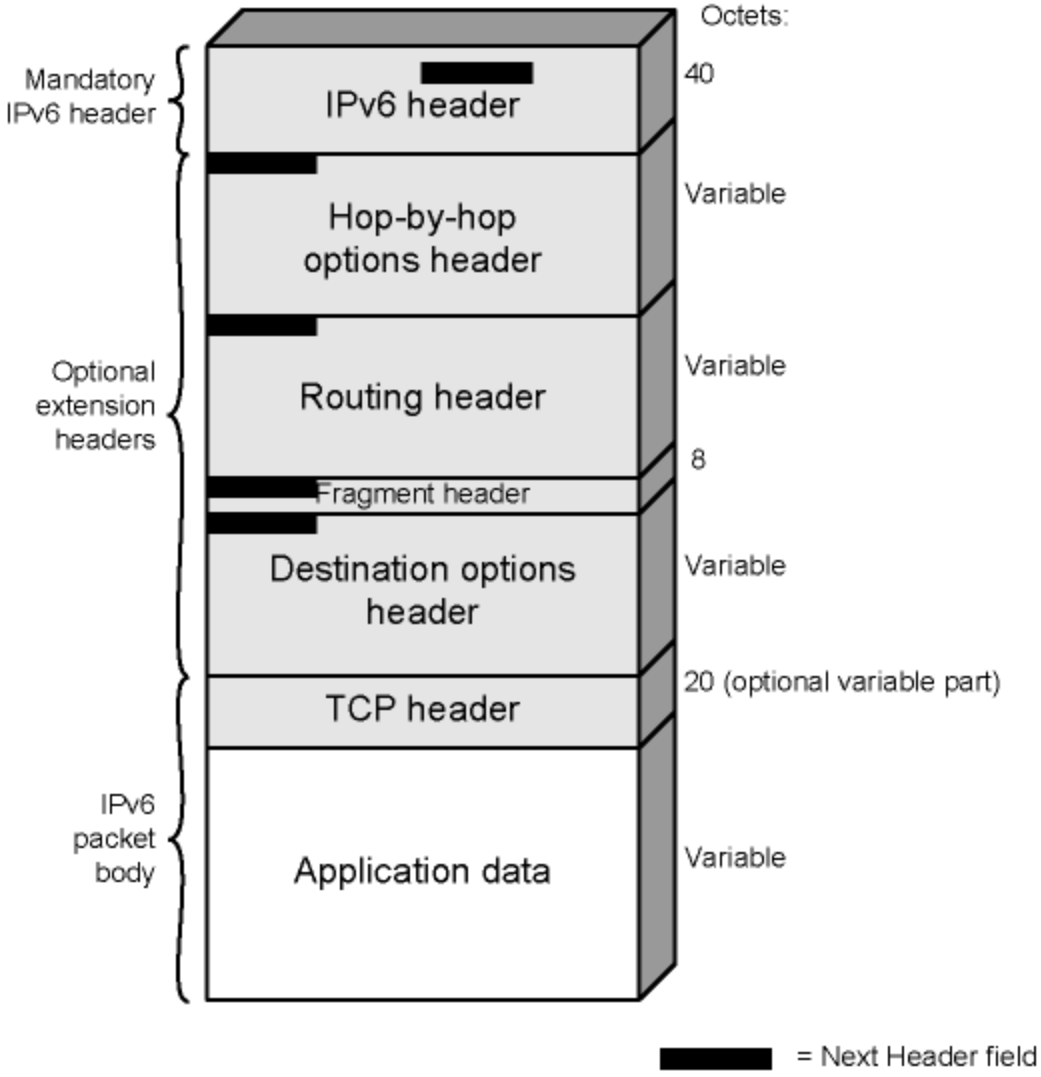
- From the source's PoV: a flow is a sequence of packets generated from a single application instance at the source and have the same transfer service requirements. An application may generates a single flow or multiple flows, each flow may comprise one or more TCP connections.

- From the router's PoV: A flow is a sequence of packets that share attributes of how these packets are handled by the routers, including path, resource allocation, discard requirements, accounting, and security attributes.

Flow Label (3/3)

- Three rules apply to the flow label:
  - Hosts or routers that do not support the Flow Label field must set it to zero when originating, pass the field unchanged when forwarding, and ignore when receiving .
  - All packets originating from a given source with the same nonzero Flow Label must have the same source address, destination address, Hop-by-hop Options header contents (if present), and routing header contents (if present).
  - The source assigns a flow label to a flow uniquely in the range of 1 to 2^{20} -1.

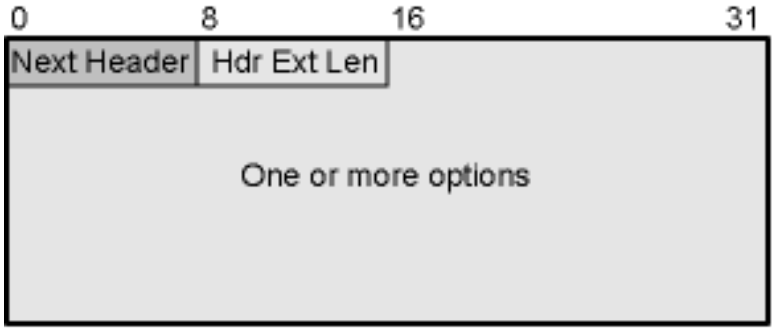
Sample IPv6 Packet with Extension Headers



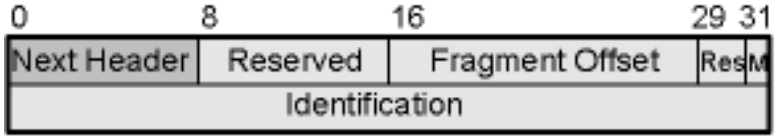
Extension Headers (in sequence)

- Hop-by-Hop Options
  - Require processing at each router, if present
- Routing
  - Similar to v4 source routing
- Fragment
- Authentication
- Encapsulating security payload
- Destination options (could also appear before Routing Header, for destination node only)

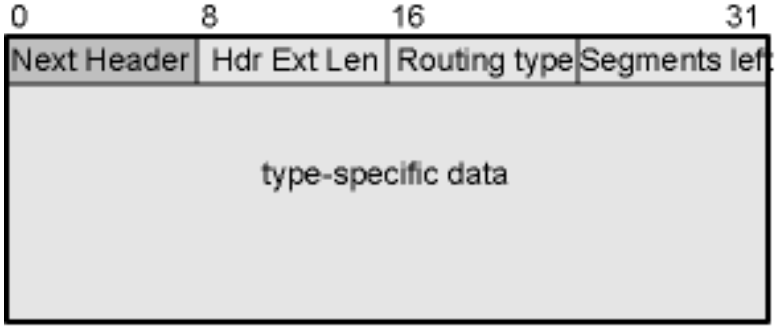
IPv6 Extension Headers



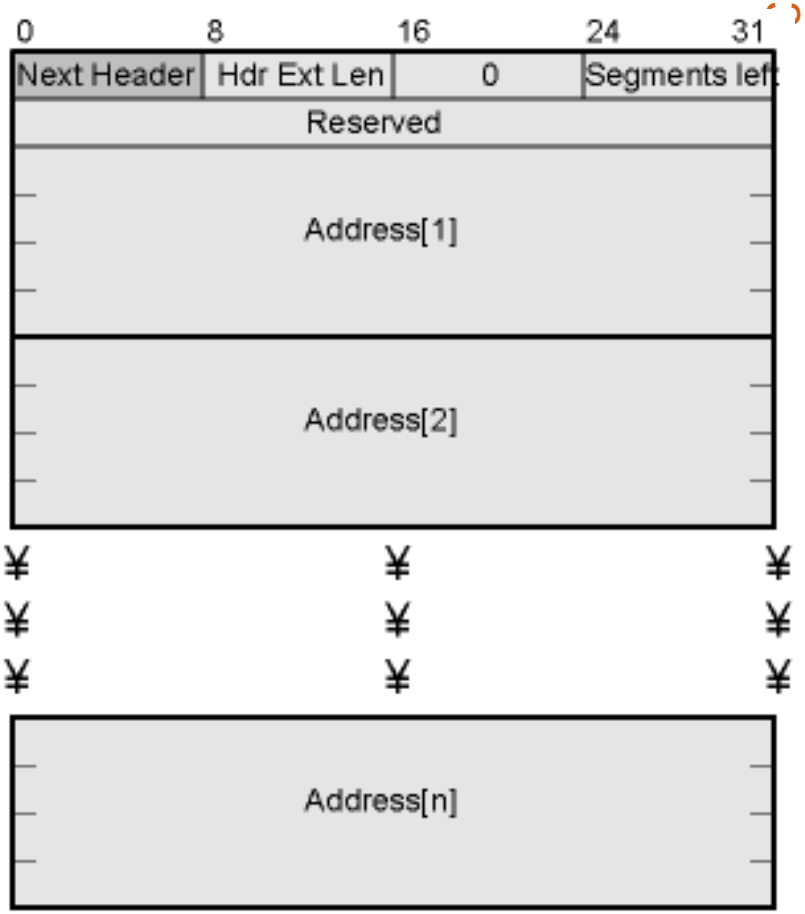
(a) Hop-by-hop options header; destination options header



(b) Fragment header



(c) Generic routing header



(d) Type 0 routing header

Hop-by-Hop Options (0)

- Option Type (8 bits)
  - Action code (higher order 2 bits)
  - Option data field may or may not change en route from source to destination, specified at the third higher order bit
  - Operation code (5 lower order bits)

Hop-by-Hop Options (0)

- Option Type – length – value
- Options (TLV encoding)
  - Pad1-- Insert one byte of padding into Options area
  - PadN
    - Insert *N* (≥2) bytes of padding into Options area
    - Ensure header is multiple of 8 bytes
  - Jumbo payload
    - Support datagram over 64K = 65,535 octets
  - Router alert
    - Tells router that contents of packet is of interest
    - Provides support for traffic control, such as RSVP

Routing Header (43)

- List of one or more intermediate nodes to be visited on the way to destination (like source routing in IPv4)
- Four fixed header fields:

1. Next Header
2. Header extension length (multiple of 8B)
3. Routing type (type 0 currently)
4. Segments left (decremented each visit)
   - i.e. number of nodes still to be visited

Fragment Header (44)

- Fragmentation only allowed at source
- No fragmentation at intermediate routers
- Node must perform path discovery to find smallest MTU of intermediate links
- Source fragments to match MTU
- Otherwise, limit to 1280 octets, which is the minimum MTU that must be supported by each network (minimum permitted IPv6 MTU).

Fragment Header Fields

- Next Header (8 bits)
- Reserved (8 bits) for future use
- Fragment offset (13 bits for 8B unit)
- Reserved (2 bits) for future use
- More flag (1= more fragment, 0 last)
- Identification (32 bits)

Authentication Header (AH, 51)

- Ensure integrity and prevent from replay attack, there are 6 header fields:
  1. Next Header (1B)
  2. Payload Length (1B) (excluding the first 8B and the number is divided by 2)
  3. Reserved (2B)
  4. Security parameter Index (SPI 4B, for Security Association SA)
  5. Sequence Number (4B) (against replay attacks)
  6. Authentication Data (multiple of 4B)

Encapsulating Security Payload Header (ESP, 50)

- Ensure data confidentiality according to AH specification
- Header format

Security Parameter Index (SPI, 32 bits)

Sequence Number Field (32 bits)

Payload Data (variable length)

Padding (0-255 bytes) (pad length and next header)

Authentication Data

Destination Options (60)

- If present, optional info. is examined only by the packet's destination node.
- Same format as Hop-by-Hop options header
- Only extension header which could appear twice (after routing header or after encrypted security payload header)
- Note: Next header = 59 means next extension header doesn't exist

IPv6 Addresses

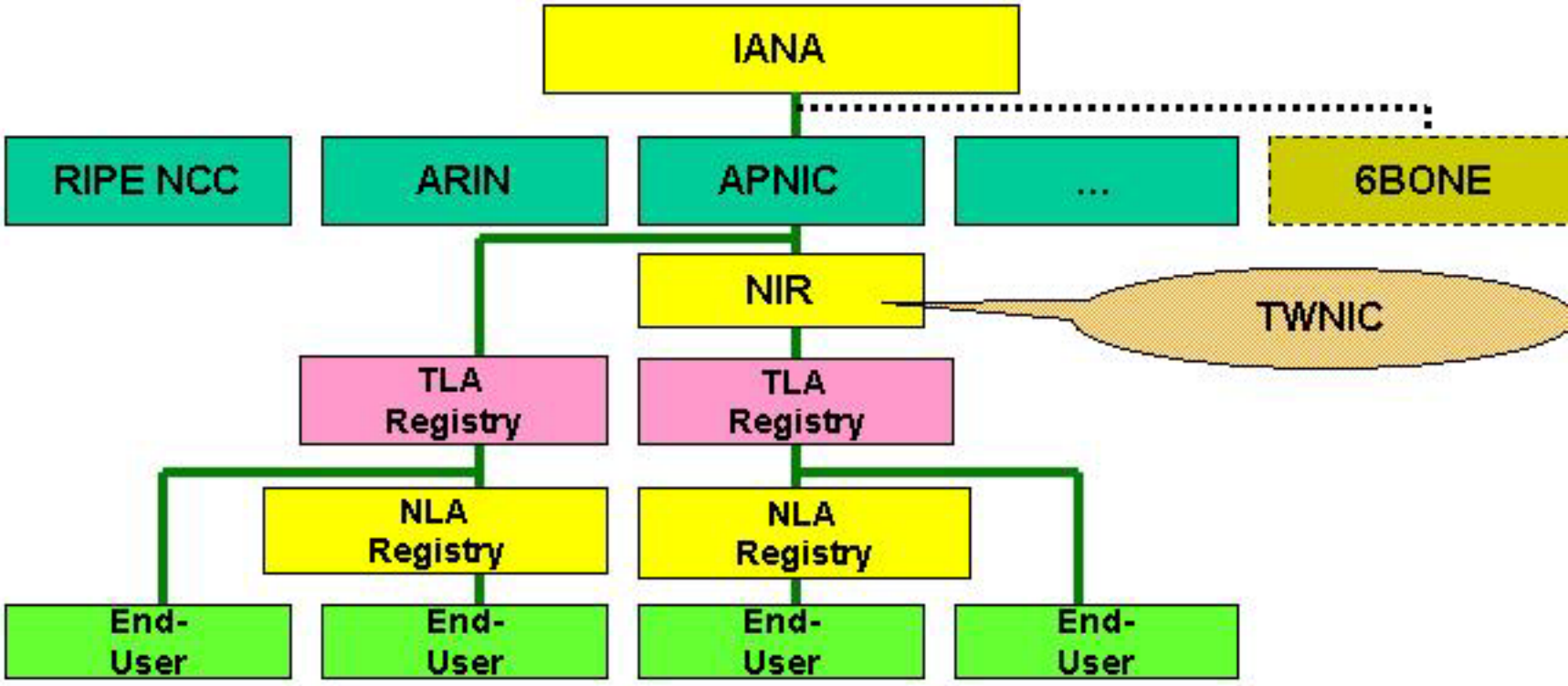
- 128 bits - written as eight 16-bit hex numbers, for example:

`5f1b:df00:ce3e:e200:0020:0800:2078:e3e3`

- High order bits determine the *type* of address.

International IPv6 Address Management

Management Method – Hierarchical Management



Three IPv6 Address Types

- **Unicast** – address of a single interface
  – for one-to-one communication
- **Anycast** – address of a set of interfaces
  – for one-to-nearest communication
- **Multicast** – address of a set of interfaces
  – for one-to-many communication

* No more broadcast addresses as used in IPv4

Some well-known Format Prefix

- 0000 001 NSAP Allocation
- 0000 010 IPX Allocation
- 001 Global unicast
- 010 Provider-based Unicast
- 100 Geographic-based Unicast
- 1111 1110 10 Link local use addresses
- 1111 1110 11 Site local use addresses
- 1111 1111 Multicast

:: for unassigned

::1 for loop back address

Unicast Addresses

- Subnet prefix and interface ID
- EUI-64 interface ID = 24-bit administered company ID + 11111111 11111110 (FFFE)+ interface number
- Link-local unicast addresses (1111111010, FE80::/10), Site-local unicast addresses (1111111011,FEC0::/10),

  Global unicast addresses (0010….0, 2000::/3)

Link-Local Unicast addresses

- Prefix= FE80::/64 and 64 bits for interface ID
- Link-Local addresses are assigned on a single link for the purpose of auto-address configuration, neighbor discovery, or when no routers are present
- Routers must not forward any packet with link-local source or destination addresses to other links

MULTICAST ADDRESSES (1/2)

- Prefix= FF0x::/64 and 64 bits for interface ID
- FF+4-bit Flags+4-bit scope+112-bit Group ID

  Flags (000T) T=0 for permanent group

  Scope 1 for interface-local, 2 for link local, 4 for admin-local, and 5 for site-local
- Distinguish between FF02::1AA and FF12::1AA, both are link-local group address, one is permanent and the other on is not.

MULTICAST ADDRESSES (2/2)

- Prefix= FF0x::/64 and 64 bits for interface ID
- Link-Local multicast addresses are useful for the purpose of auto-address configuration. When a node boots, it can configure an address uniquely, then use it to discover other neighbors for globally unique address.
- All-nodes multicast address (FF01::1, FF02::1), link-local all routers multicast address (FF02::2), site-local all routers multicast address (FF05::2)

ANYCAST ADDRESSES

- Have same scope as unicast addresses
- Are assigned to more than one interfaces
- IP packet arrives at a router, it depends router's implementation to choose a node
- Is useful for providing redundancy
- Is useful to discover some services like the home agent function in Mobile IPv6
- Most significant 57 bits of interface ID is set to 1 and leave 7 bits for total of 128 anycast addresses

Aggregatable Global Unicast Address – longest prefix match

FP: 3bit of Format Prefix (001)
TLA: top-level aggregation
NLA: next-level
SLA: site-level
* Interface ID is (typically) based on hardware MAC address

TLA and NLA address allocation

- From 3-13-32(8+24)-16-64 of
  FP-TLA -NLA -SLA - Interface
- To 3-13-13-19-16-64 of
  FP-TLA1-TLA2-NLA-SLA- Interface

or 3-13-13-6-13-16-64 of
FP-TLA1-TLA2-RES-NLA-SLA-Interface

Some IPv6 networks in Taiwan (min. conn. Unit is /48)

Hinet 2001:238::/35
TANet 2001:288::/35
6BONE 3FFE:3600::/24
NCHU 2001:288:01C0::/42
Taichung City 2001:288:1D80::/41
Area network centers have NLA1 length of 6,7,9 bits
County network centers have NLA1 length of 4-7 bits

Other IPv6 Address Assignment

- 6Bone 3FFE::/16
- 6to4 tunnels 2002::/16
- APNIC 2001:0200::/23
- ARIN 2001:0400::/23
- RIPE NCC 2001:0600::/23

IPv4 / IPv6 Transition Technology

- Dual Stack Mechanism
  - Support both IPv4 and IPv6
  - v6-v6 through v4 routers, some field, such as flow, may be lost
- Tunnel Mechanism
  - Encapsulating IPv6 packets by IPv4
  - for connecting both IPv6 islands through IPv4 networks (oceans)
  - Dual Stack, 6over4, 6to4 tunnel mechanisms
- Translation Mechanism
  - Network Address Translation - Protocol Translation (NAT-PT)
  - TCP-UDP Relay
  - Bump-In-the-Stack (BIS)
  - Socket-Based Gateway

IPv6 containing IPv4 addresses

- IPv4 addresses are embedded in the least significant 32 bits, such as IPv4-mapped IPv6 addresses and IPv4-translated (or compatible) IPv6 address
- Or, IPv4 addresses are embedded in the 32 bits following the first 2 octets, such as 6-to-4 addresses

IPv4-Mapped IPv6 Address (1)

- IPv4-Mapped addresses allow a host that support both IPv4 and IPv6 to communicate with a host that supports only IPv4.

- The IPv6 address is based completely on the IPv4 address.

IPv4-Mapped IPv6 Address (2)

- For an IPv6 node connecting to an IPv4-only node
- 80 bits of 0s followed by 16 bits of ones, followed by a 32 bit IPv4 Address:

WORKS WITH DNS

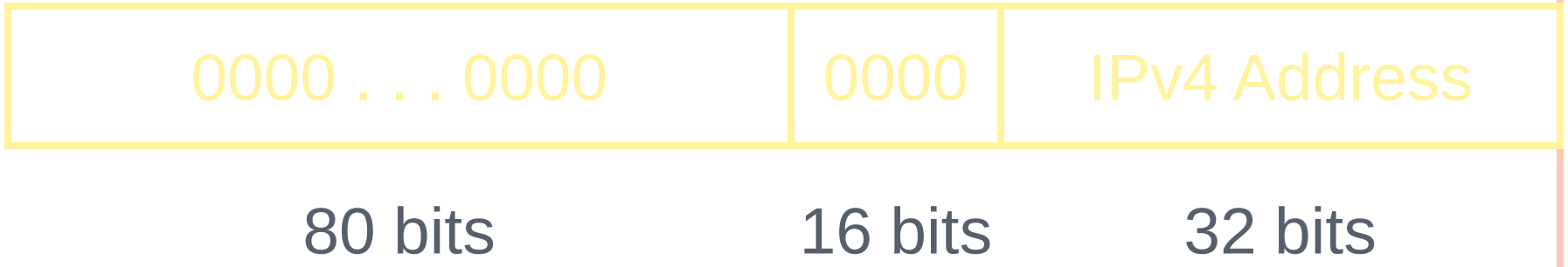
- An IPv6 application asks DNS for the address of a host, but the host only has an IPv4 address.
- DNS creates the IPv4-Mapped IPv6 address automatically.
- Kernel understands this is a special address and really uses IPv4 communication.

IPv4-Compatible IPv6 Address (1)

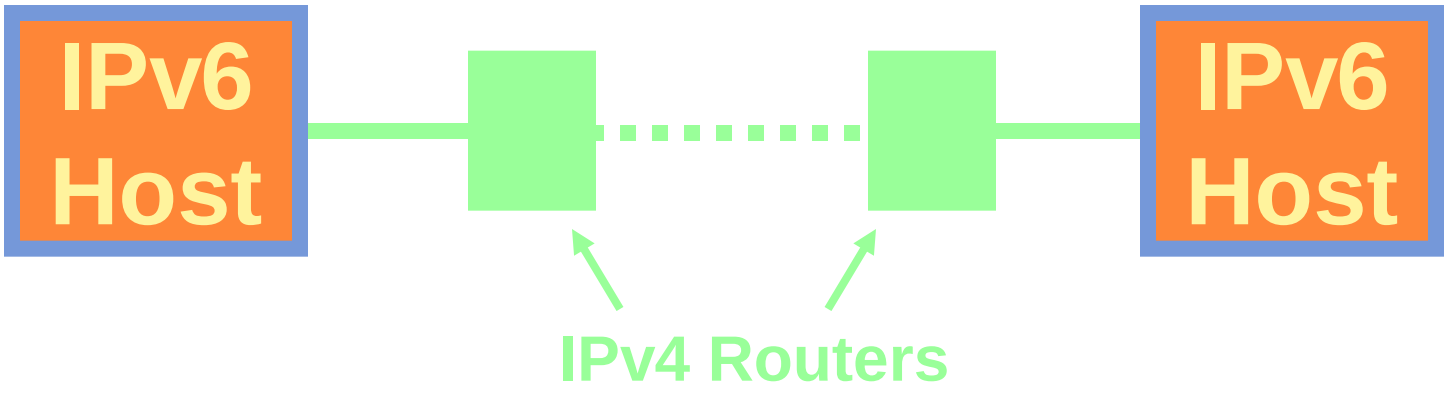
- An IPv4-compatible address allows a host supporting IPv6 to talk IPv6 even if the local router(s) don't talk IPv6.
- IPv4 compatible addresses tell endpoint software to create a tunnel by encapsulating the IPv6 packet in an IPv4 packet.

IPv4-Compatible IPv6 Address (2)

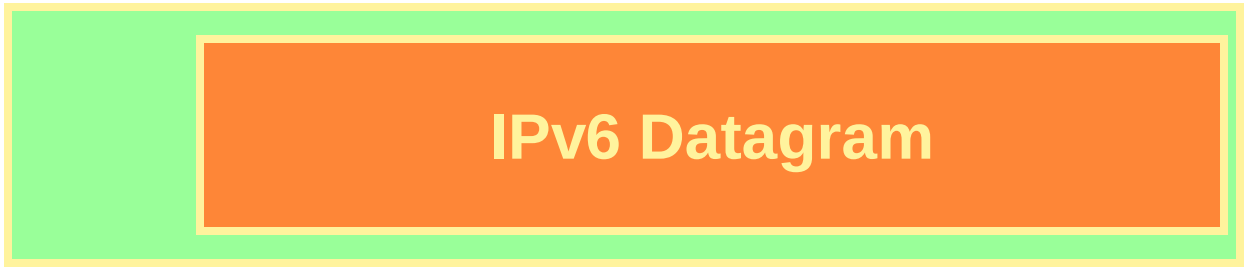
- 80 bits of 0s followed by 16 bits of 0s, followed by a 32 bit IPv4 Address:



Tunneling
(done automatically by kernel when IPv4-Compatible IPv6 addresses used)



IPv4 Datagram



IPv6 Islands over IPv4 networks

- Tunnel building mechanisms: manually configured, semi-automated configured, or automated configured.
- Manually configured – the simplest way, by encapsulating IPv6 header within IPv4 header
- Semi-automated – with the help of Tunnel Broker (RFC 3053)

IPv6 Islands over IPv4 networks

- Automated configured -- 6over4 and 6to4
- 6over4 (RFC 2029) uses IPv4 multicast mechanism, L4+&IPv6 ↔ L4+&IPv6&IPv4, a mapping is required at the end points.
- 6to4 (RFC 3056) allows IPv6 island to communicate with IPv6 end point via IPv4 ocean, prefixed by 2002:v4_addr::/16
- 001(FP) + 0002(TLA) + IPv4-address (ISP assigned) + SLA ID (local administered) + 64-bit Interface

6TO4 (TUNNELING)

- Is a mechanism allowing organizations to experiment with IPv6 without –
  1. an upstream ISP supporting IPv6
  2. applying for IPv6 address space
  3. arranging a "tunnel" with another IPv6 user
- The only thing a 6to4 user needs is a global IPv4 address, reachable on protocol 41.

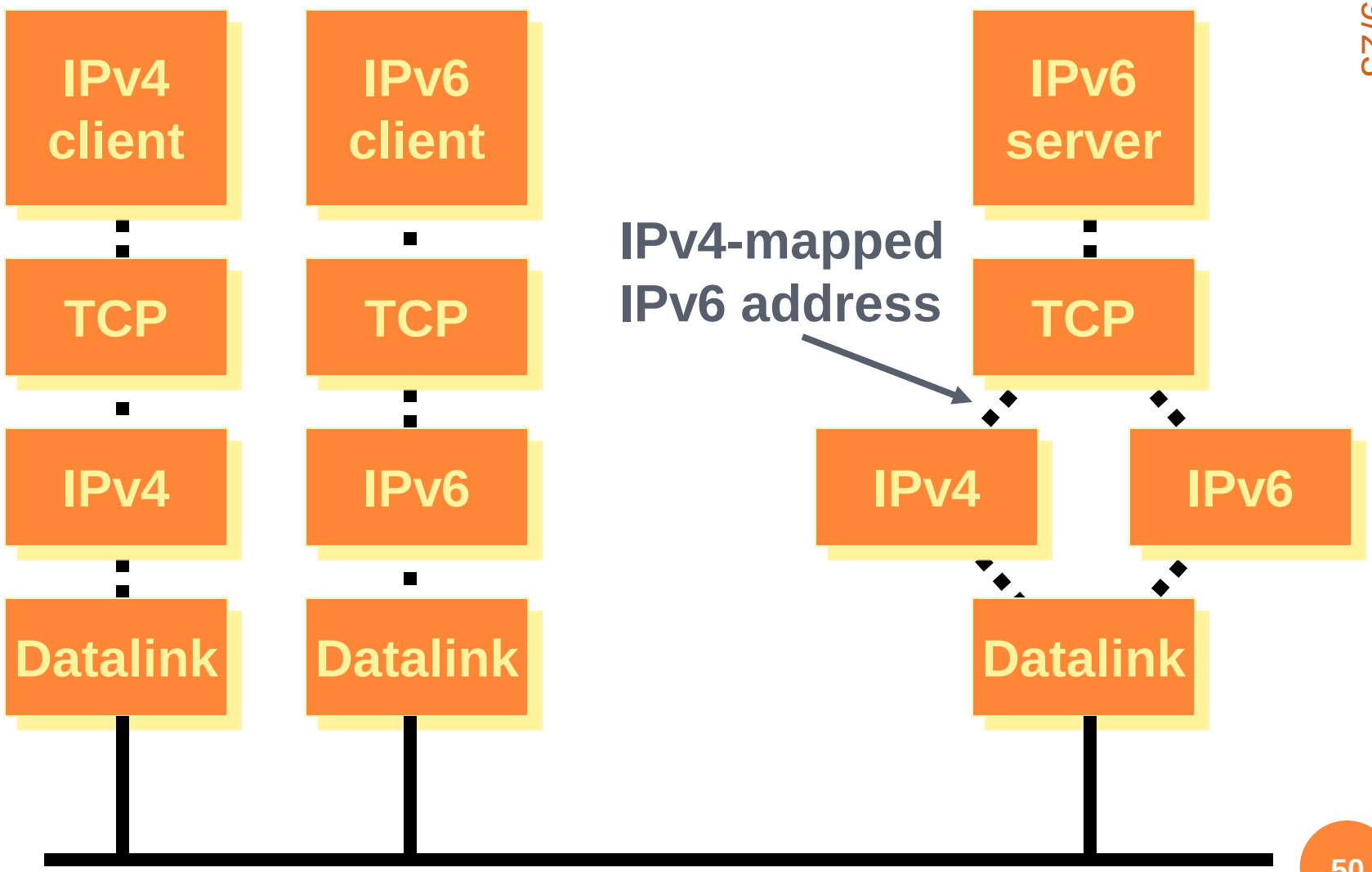
  For example, 192.0.2.4 at 6to4 relay server has address 2002:c000:0204::/48
- RFC 3056 and RFC 3068 (with addition of anycast capability)

TRANSLATOR

- With/without changing connecting hosts –
  1. TCP-UDP Relay
  2. BIS and NAT-PT
- TCP-UDP relay builds connection over the transport layer for peer points
- NAT-PT translates IPv4 and DNS requests over the network layer,using Staless IP/ICMP translator (SIIT) algorithm.

Dual Server

- It is important to create servers that handle both IPv4 and IPv6.
- The work is handled by the O.S. (which contains protocol stacks for both v4 and v6):
  - automatic creation of IPv6 address from an IPv4 client (IPv4-mapped IPv6 address).


IPv4 client
IPv6 client
IPv6 server
IPv4-mapped IPv6 address
TCP
TCP
TCP
IPv4
IPv6
IPv4
IPv6
Datalink
Datalink
Datalink

IPv6 Sockets programming

- New address family: **`AF_INET6`**
- New address data type: **`in6_addr`**
- New address structure: **`sockaddr_in6`**

IN6_ADDR

```
struct in6_addr {
    uint8_t s6_addr[16];
};
```

SOCKADDR_IN6

```
struct sockaddr_in6 {
  uint8_t          sin6_len;
  sa_family_t      sin6_family;
  in_port_t             sin6_port;
  uint32_t         sin6_flowinfo;
  struct in6_addr  sin6_addr;
};
```

IPv6 - IPv4 Programming

- The kernel does the work, we can assume we are talking IPv6 to everyone!
- In case we really want to know, there are some macros that determine the type of an IPv6 address.
  - We can find out if we are talking to an IPv4 client or server by checking whether the address is an IPv4 mapped address.

IPv6 Clients

- If an IPv6 client specifies an IPv4 address for the server, the kernel detects and talks IPv4 to the server.
- DNS support for IPv6 addresses can make everything work.
  - **`getaddrinfo()`** returns an IPv4 mapped IPv6 address for hosts that only support IPv4.

ICMPv6

- ICMPv6 encompasses the roles filled by ICMP, IGMP, and ARP in the IPv4 world.
- The most important change of ICMPv6 is in the area of neighbor discovery.
- RFC 2463 for similar parts of ICMP, RFC 2461 for neighbor discovery, RFC 2462 for DAD.
- ICMPv6 is an IP protocol (unlike ARP which is blurred in layering), and it can be secured by IPsec.

ICMPv6 Introduction(1/7)

- ICMPv6 is used by IPv6 nodes to report errors encountered in processing packets, and to perform other internet-layer functions, such as diagnostics (ICMPv6 "ping") and multicast membership reporting.
- An integral part of IPv6 and MUST be fully implemented by every IPv6 node.
- Report delivery or forwarding errors
- Next Header value = 58 (1 for ICMPv4)

ICMPv6 Introduction(2/7)

- Error messages (type = 1-127) or information messages (type = 128-255)
- ICMPv6(RFC2463) is more powerful than ICMPv4 and contains new functionality.
  - IGMP (Internet Group Management Protocol)
  - ARP/RARP (Address Resolution Protocol / Reverse Address Resolution Protocol))
  - Neighbor Discovery (RFC 2461)
  - Supports Mobile IPv6

ICMPv6 Introduction(3/7)

- Message format: type (8-bit) code (8-bit) checksum (16-bit) and message content (length depends on the type)
- Checksum is taken over the source and destination fields of IPV6 header (no checksum in the IPV6 header)
- For type 1 (destination unreachable)

code 0 – no route to destination

code 1 – comm. With destination is prohibited

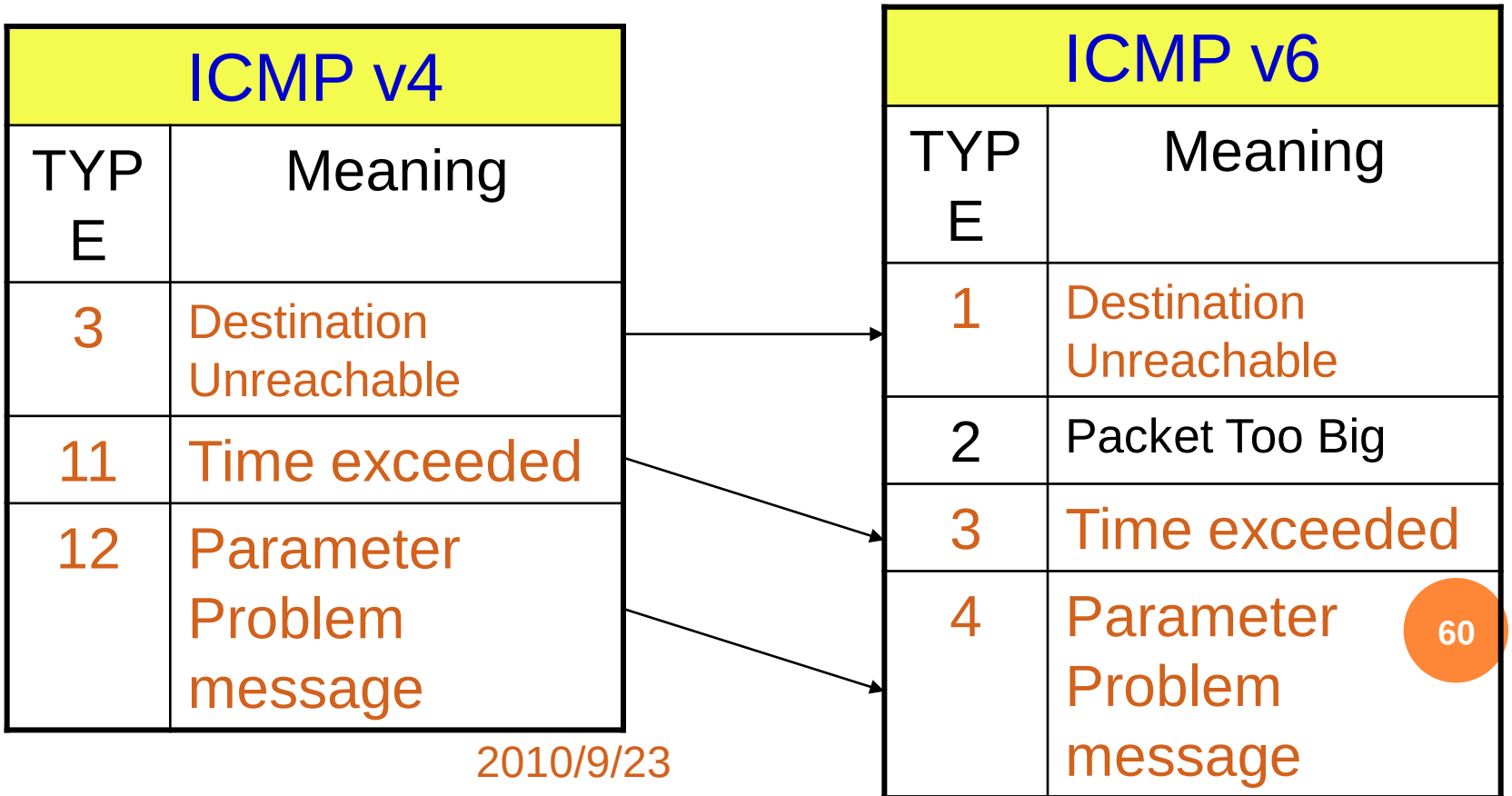
code 2 – the code is not assigned'

code 3 – address unreachable

code 4 – port unreachable

ICMPv6 Introduction(4/7)

- ICMPv4 compare with ICMPv6 for Error Messages



ICMPv6 Introduction(5/7)

- ICMPv4 compares with ICMPv6

for Information Messages

ICMP v4	
TYPE	Meaning
0	Echo Reply
4	Source Quench
5	Redirect
8	Echo Request
13	Timestamp Request
14	Timestamp Reply
15	Information Request
16	Information Reply
17	Address mask Request
18	Address mask Reply

ICMP v6	
TYPE	Meaning
128	Echo Request
129	Echo Reply
130	Multicast Listener Query
131	Multicast Listener Report
132	Multicast Done
133	Router Solicitation
134	Router Advertisement
135	Neighbor Solicitation
136	Neighbor Advertisement
137	Redirect Message
138	Router Renumbering
139	Node Information Query
140	Node Information Response
141	Inverse ND Solicitation
142	Inverse ND Advertisement Message
150	Home Agent Address Discovery Request Message
151	Home Agent Address Discovery Reply

ICMPv6 Introduction(6/7)

ICMPv4 訊息及對應的 ICMPv6 訊息

ICMPv4 訊息	對應的 ICMPv6 訊息
Destination Unreachable - Network unreachable (Type 3、Code 1)	Destination Unreachable - No route to destination (Type 1、Code 0)
Destination Unreachable - Host Unreachable (Type 3、Code 1)	Destination Unreachable - Address unreachable (type 1、Code 3)
Destination Unreachable - Protocol unreachable (Type 3、Code 2)	Paremeter Problem - Unrecognized Next Header 欄位 (Type 4、Code 1)
Destination Unreachable - Port Unreachable (Type 3、Code 3)	Destination Unreachable -Port Unreachable (Type 1、Code 4)
Destination Unreachable - Fragmentation needed and DF set (Type 3、Code 4)	Packet too Big (Type 2、Code 0)
Destination Unreachable - Communication with destination host administratively prohibited (Type 3、Code 10)	Destination Unreachable -Communication with destination administratively prohibited (Type 1、Code 1)
Time Exceeded - TTL expired (Type 11、Code 0)	Time Exceeded - Hop Limit exceeded (Type 3、Code 0)
Time Exceeded - Fragmentation timer expired (Type 11、Code 1)	Time Exceeded -Fragmentation timer expired (Type 3、Code 1)
Parameter Problem (Type 12、Code 0)	Parameter Problem (Type 4、Code 0 或 2)
Source Quench (type 4、Code 0)	在 IPv6 中未建置此訊息。
Redirect (Type 5、Code 0)	Neighbor Discovery Redirect message (Type 137、Code 0)。有關詳細資訊，請參閱 "Neighbor Discovery"。

ICMPv6 Introduction(7/7)

- ICMPv6 通訊協定還提供以下功能的架構：
  - Multicast Listener Discovery (MLD) MLD 是一組 ICMP 訊息 (3 個)，它們取代 IPv4 網際群組管理通訊協定 (IGMP) 的第 2 版，以管理子網路多點傳播成員身份。
  - Neighbor Discovery (ND) ND 是一組 ICMPv6 訊息 (5 個)，它們管理連結上節點間通訊。Neighbor Discovery 取代位址解析通訊協定 (ARP)、ICMPv4 Router Discovery 以及 ICMPv4 Redirect 訊息。
- ICMPv6 是 IPv6 建置所必需的，(RFC2463) Every node in IPv6 must implement ICMPv6。

Multicast Listener Discovery (RFC 2710)

§ MLD has three message types:

1. Multicast Listener Query – allows a router to find a specific multicast address or all that nodes have joined.
2. Multicast Listener Report – announces that someone is listening on a specific group.
3. Multicast Listener Done – lets a router know that all listeners on an address may be finished and it should send a query to make sure.

§ MLD ensures that only a small number of messages need to be sent to keep the local routers up-to-date.

Note that: multicast traffic routing is a challenge.

Neighbor Discovery (RFC 2461)

§ Replaces IPv4 ARP, ICMP Router discovery, ICMP Redirect, etc

§ ND is useful for:

- Find link-layer address of neighbors
- Find neighboring routers
- Actively keep track of neighbor reachability
- Send network information from routers to hosts
- Be useful for host autoconfiguration
- Neighbor unreachability detection (NUD)
- Duplicate address detection (DAD)

Data Structures for Neighbor Discovery

§ ND Cache
§ Destination Cache
§ Prefix List
§ Default router list

- States of items in ND Cache
  1. Incomplete
  2. reachable
  3. stale
  4. delay
  5. probe

Five ICMPv6 Types Used for Neighbor Discovery

- Router solicitation (type 133)
- Router advertisement (type 134)
- Neighbor solicitation (type 135)
- Neighbor advertisement (type 136)
- redirect (type 137)

Neighborhood Watch

- ICMP ND is an IP protocol, and can be secured with IPsec.

- **Like ARP, ND explicitly includes the link-layer addresses within body of message. This makes for easier implementation and also leaves the option of proxy ND open for situations such as Mobile IP.**

ND --- Address Resolution

- Node A is trying to connect to Node B

- **A sends out Multicast Neighbor Solicitation message** (ARP request)
  Assume both B & C receive and response
  (which is sent to translate a IPv6 unicast address into a link-layer address)

- **B responses with Unicast Neighbor Advertisement message , while C discards the message**

ND --- Neighbor Solicitations

- For address resolution, NUD, and DAD

- **When nodeA is sharing a link with nodeB, and nodeA has nodeB's IP address but no link-layer address**
- **NodeA sends out a NS to nodeB (source address is nodeA's address and detination address is nodeB's solicited node multicast address, while the target address is nodeB's unicast address**
- **NodeA's link-layer driver sends to a link-layer multicast address, nodeB's IP layer will receive the message and reply to nodeA with a neighbor advertisement message**

ND --- Neighbor Advertisements

- Sent in response to solicitations or sent unsolicited

- **Is also used for neighbor unreachable detection (NUD)**
- **Reachability detection of a neighbor – reachability confirmation is achieved by sending a neighbor solicitation unicast to a neighbor**
- **Neighbor advertisements can also be sent by a proxy node, either solicited or not.**

ND --- Router Solicitations

- For hosts to discover one or more default neighboring routers when they first attach to a link

- **router solicitations are sent to the all-routers multicast address, which is a link-local multicast address (FF02::2)**

- **the reception of a router solicitation will result in sending router advertisement from each router on-link**

ND --- Router Advertisements

- Router advertisements can be sent as a response to router solicitations or regularly in an unsolicited manner

- **When a router is solicited, the advertisement is sent to the unicast address of the soliciting node**
- **Router advertisements allow neighbors to discover the default routers' IP and link-layer addresses, in addition to link prefix and MTU**
- **Router lifetime – the time (s) during which the router can be considered as a default router**
- **Reachable time – the time (ms) that a node can assume a neighbor is reachable**
- **Retransmission timer – the frequency (ms) that address resolution of other nodes should be done**

Duplicate Address Detection (DAD)

- A has address FE80:2DD:FF:FE11:1111 and B has address FE80:2DD:FF:FE22:2222, if C is new and the same address (tentative address) as node A

- **C sends out Multicast Neighbor Solicitation message**
  Both B & C may receive and response

- **A responses with Multicast Neighbor Advertisement message**

- Note that: C sends out with source address :: and A sends to all-nodes link-local address FF02::1

Neighbor Unreachability Detection (NUD)

- NUD is way of checking that we are still in bidirectional contact with a neighbor.

- In case a neighbor has become unreachable because of a change of layer 2 address, the Neighbor Solicitation will be used to discover the new layer 2 address.

- In case a router is unavailable, then another router could be chosen.

ND --- Router Discovery (1)

- IPv6 routers routinely send out router advertisement message which includes prefix, MTU, Hop limit, … to notify nodes its router role

- Booting node multicasts RS, all local routers respond with a unicast RA message (?)

- The sending node together with all others update their configuration parameters according to the received RA message

ND --- Router Discovery (2)

- Router solicitation are sent by booting nodes to request RAs for configuring the interfaces.

RS (sending by the booting host):

ICMP Type = 133
Src = ::
Dst = All-Routers multicast Address (FF02::2)
query= please send RA

RA (responding by the local routers):

ICMP Type = 134
Src = Router Link-local Address
Dst = All-nodes multicast address (if regularly advertise ?)
Data= options, prefix, lifetime, autoconfig flag

Parameters for Router's Influence

- Specify that hosts must do stateful autoconfiguration
- Specify that hosts must do stateless autoconfiguration
- Specify the link MTU
- Specify the default hop limit
- Specify the length of time for which hosts are considered reachable

Redirect

- Host A unicasts to router A,
- If router A realizes a better next hop for host A, say router B, router A forwards the message to router B
- Router A sends redirect message to notify host A the link-local address of router B
- In the subsequent messages from host A will be sent directly to router B

Address autoconfiguration (1)

- Stateful vs Stateless autoconfiguration
- Stateless autoconfig. --- Using interface ID and prefix to construct an IPv6 address
- Stateful autoconfig. --- requires a DHCP server

Address autoconfiguration (2)

- The autoconfiguration for link-local addresses is only specified for hosts
- Routers must obtain through manual configuration

- Multicast local nodes address -- FF02::1
- Multicast local routers address -- FF02::2

Address autoconfiguration (3)

- Stateless config. – automatic generation of host identifiers (MAC to Interface ID)
  eg. 00-50-8b-c8-e6-76
  02-50-8b-c8-e6-76
  02-50-8b-ff-fe-c8-e6-76
  this is so called EUI-64 identifier
  finally, FE80:: EUI-64 forms a tentative link-local address

- When a router advertisements are received, the host performs the process of concatenating the advertised prefixes with interface id to produce other addresses.

DHCPv6

- Allowing for stateful address configuration of IPv6
- (and more …)

Uncovered IPv6-related Issues

1. Routing Protocols
2. IPv6 QoS
3. IPv6 Security (IPsec, SEND,..)
4. Integration of 3G/4G into IPv6
5. Mobile IPv6
6. Router renumbering (rfc 2894)
7. IPv6 Network Management

Small Fishes

1. Routing table growth due to the multi-homing. Solution to BGP in IPv4 and IPv6? Problem of Provider Independent (PI) and Provider Aggregate (PA) ?
2. IPv4 vs. IPv6 QoS provision. Similarly, how security is enhanced in IPv6? And, how does PMTU and fragmentation work?
3. Address auto-configuration issue (DHCP/PPP in IPv6)
4. Anycast and multicast encounters/applications always deserve attention
5. Integration of 3G/4G into IPv6
6. Mobile IPv6 and other IPv6 Network Management issues