Analyze | Assure | Accelerate™

IPv6 Test & Analysis Workshop

Presented by

Alan Way
alan.way@spirentcom.com

Agenda

- IPvx History
- Why IPv6?
- Where is IPv6
- What changed
- IPv6 basics
- IPv6 Packet Header Format
- Addressing
- Neighbor Discovery
- Extension headers
- IPv6 - IPv4 Transition
- IPv6 Testing

Sputnik Launched Oct. 4, 1957

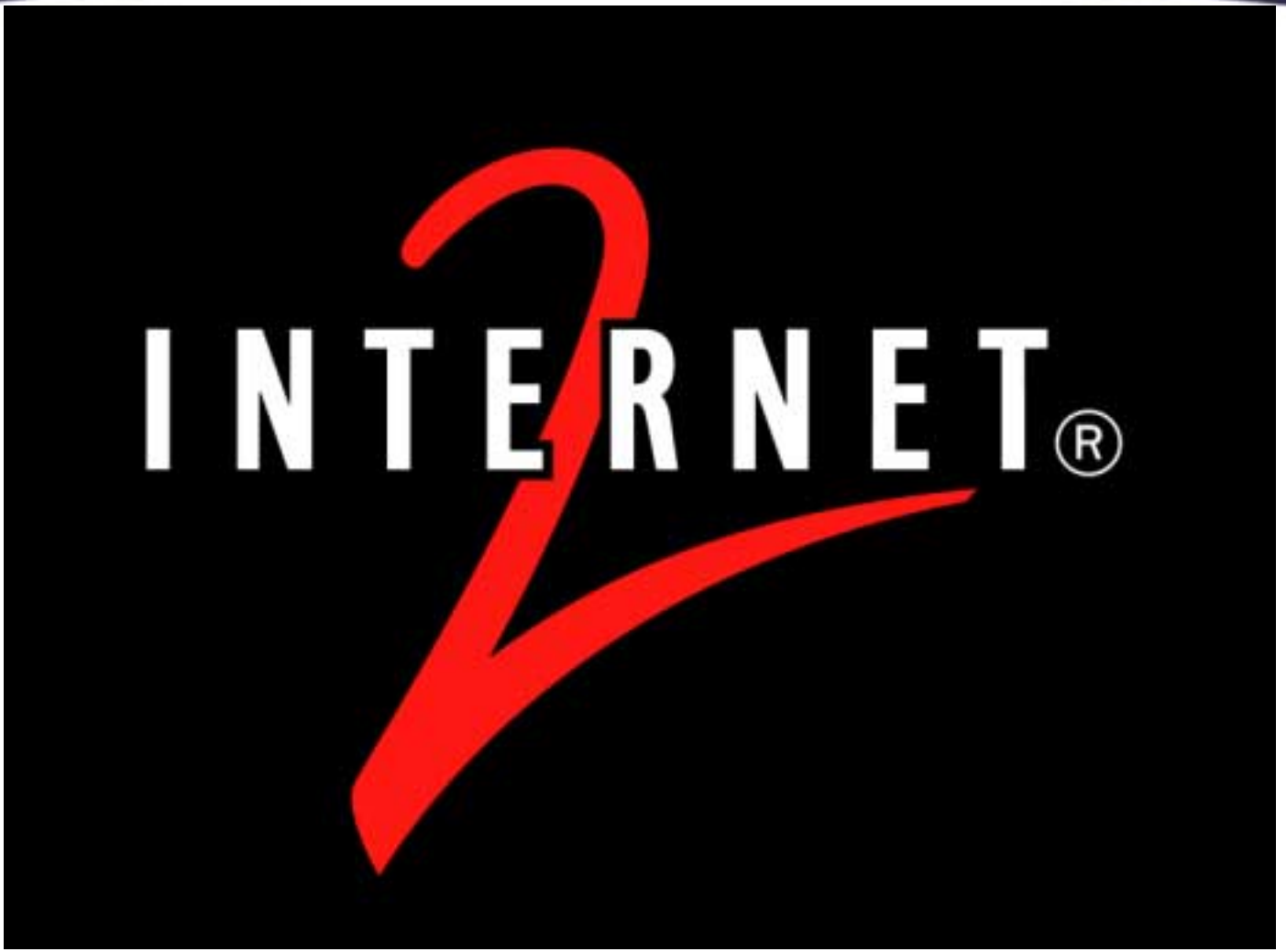
The Internet is launched

IP History

- RAND corporation proposed the basic Internet design (made public in 1964).
- Advanced Research Projects Agency Network (ARPANET) 4 nodes in place by late 1969.
- Initially used NCP. TCP/IP adopted in 1977
- Current version of IP (IPv4) has not substantially changed since 1981 with the introduction of RFC 791
- IPv6 (RFC 1883) initially released in December 1995

Where in the world is IPv6?

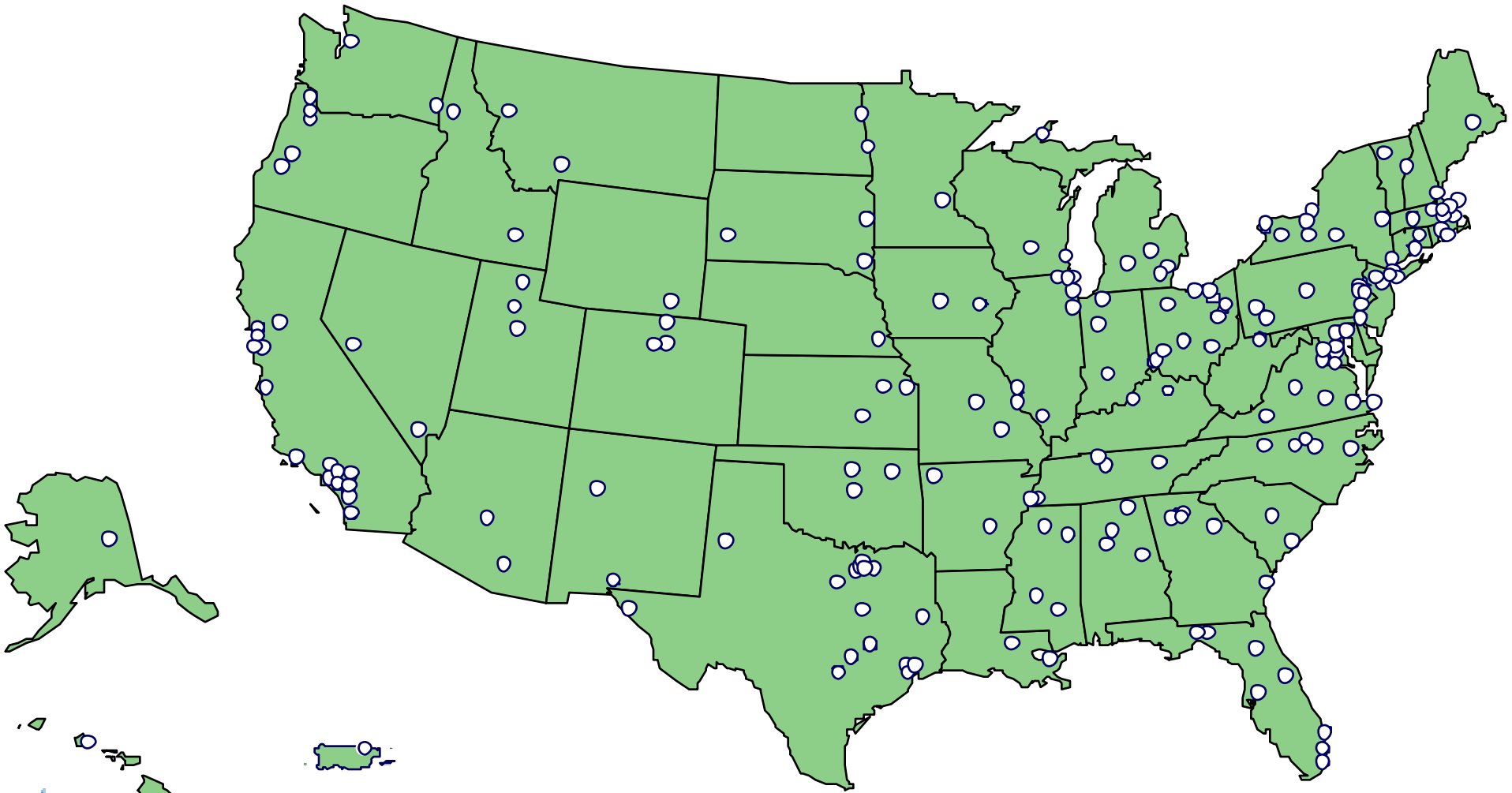
Internet2



www.internet2.edu

Internet2 Universities

198 University Members, July 2002



Internet2 Corporate Partners

Internet2 and the Next Generation Internet Initiative

Internet2
University-led

Developing education and research driven applications

Building out campus networks, gigaPoPs and inter-gigapop infrastructure

NGI
Federal agency-led

Agency mission-driven and general purpose applications

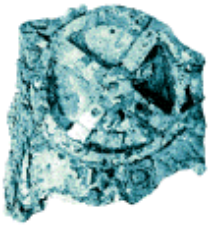
Funding research testbeds and agency research networks

Interconnecting and interoperating to provide advanced networking capabilities needed to support advanced research and education applications

Where is IPv6

- 6NET, Europe's largest Internet research project. 
- 31 project partners, including industry & academia
- Will span North America, Europe & Asia-Pacific
- Initially 9 European countries linked with 2.5Gbps
- When completed, largest & fastest IPv6 network to date
- Primarily will link Universities & Research institutions
- Will be deployed over 3 years at a cost of Eur 26.5M

6net Partners



Westfälische Wilhelms-Universität

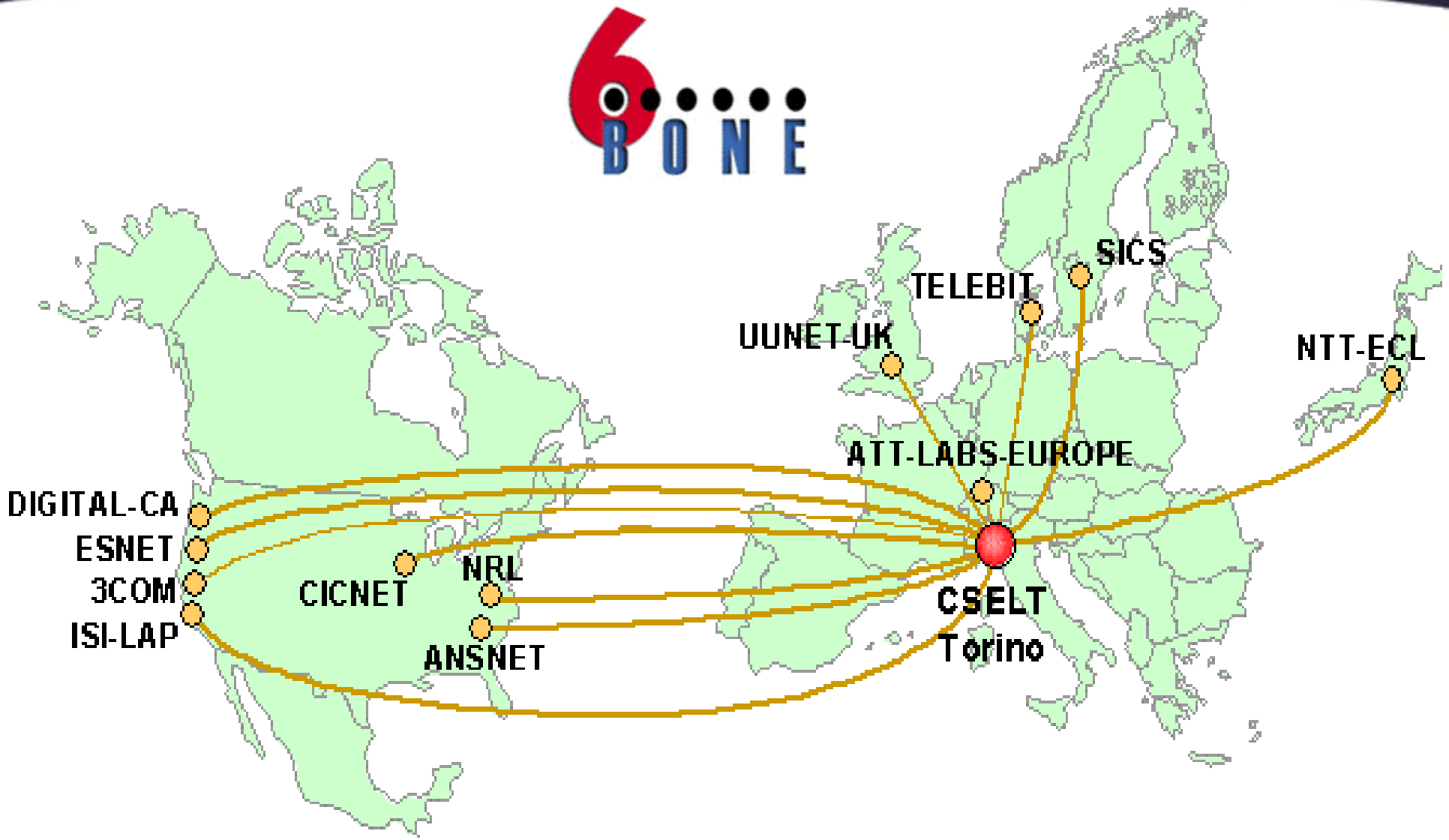
Fraunhofer Institute for Open Communication Systems

Analyze | Assure | Accelerate™

Where is IPv6

- The 6bone is an experimental IPv6 network layered on top of portions of the physical IPv4-based Internet
- The network became a reality in march 1996 with the establishment of the first tunnels between the IPv6 laboratories of G6 (France), UNI-C (Denmark) and WIDE (Japan).
- The experimental activities carried out inside the 6bone are coordinated by the IETF in order to provide feedback to various IETF IPv6-related activities and to IPv6 product developers based on test bed experience.

Where is IPv6



Why IPv6 ?

- Biggest problem with IPv4 is address space
- Network Address Translation (NAT) can be used
- NAT breaks the golden rule. It alters the data between source and destination
- Can limit services such as web hosting and IPsec
- Required for GPRS and UMTS
- U.S.A has majority of IPv4 addresses, followed by Europe the rest of the world have very few addresses

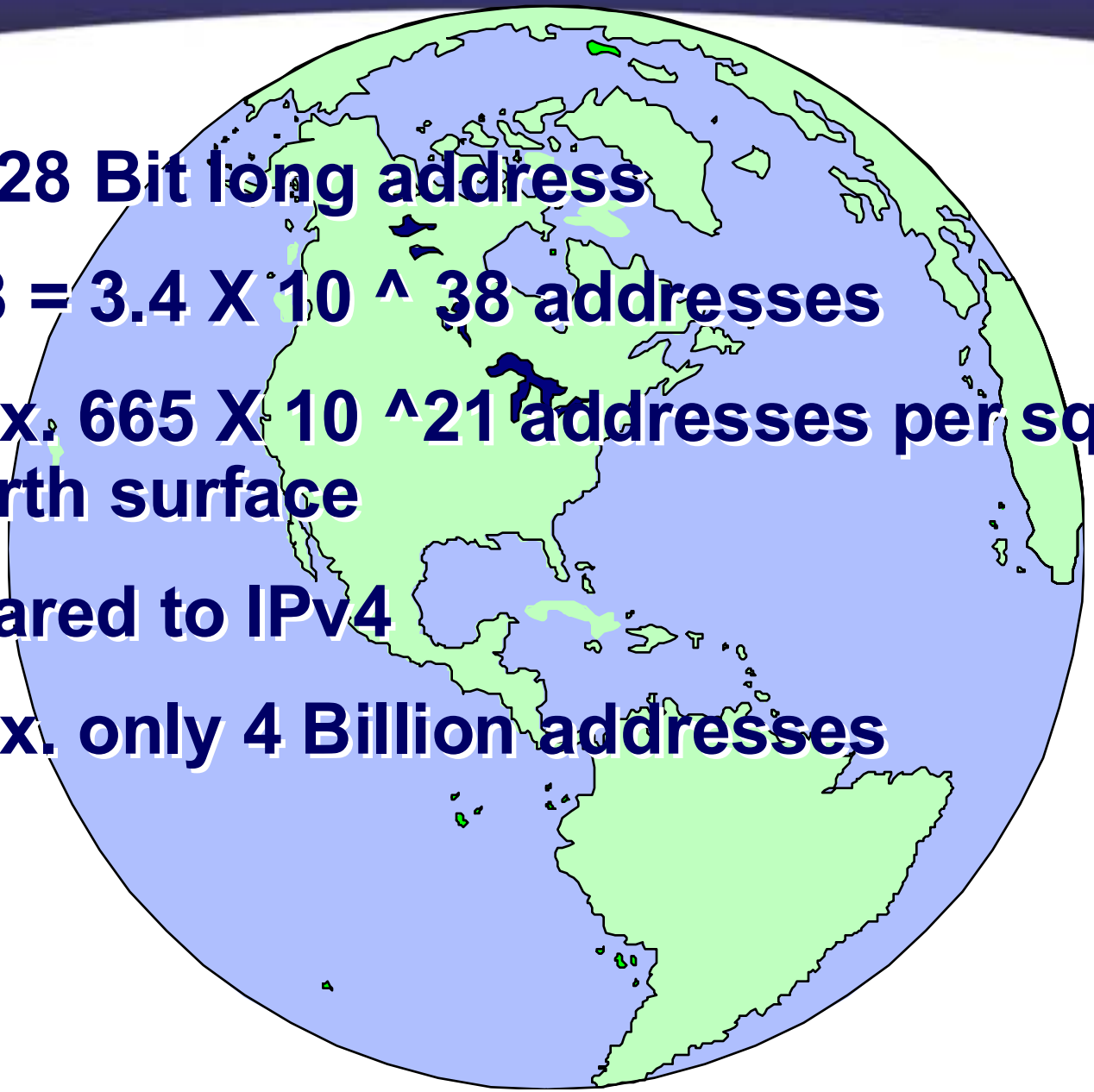
What Changed

- Address space increased from 32 bits to 128
- Built in IP security (IPsec is part of IPv6)
- Fixed length header. Optimized for hardware implementation
- Improved support for QoS, Multicast and Mobile IP
- Support for domestic appliances

Why such a large address space

- Internet is Growing:
  320 M in 2000, 550 M by 2005
- Mobile Phone Usage:
  405 M in 2000 over 1 B by 2005
- Billions of Internet appliances, always on
- MIT, Xerox, Apple each have more address space then the whole of China

The scale of change



- IPv6 128 Bit long address
- 2^128 = 3.4 X 10 ^ 38 addresses
- Approx. 665 X 10 ^21 addresses per sq.m of the earth surface
- Compared to IPv4
- Approx. only 4 Billion addresses

- IPv5 never really existed
- Used to identify experimental non-IP real time protocol
- Called ST
- Never widely used
- RFC 1819

IPv6 Basics

RFC's

- [RFC 1719] A Direction for IPng.
- [RFC 1726] Technical Criteria for Choosing IP The Next Generation (IPng).
- [RFC 1752] The Recommendation for the IP Next Generation Protocol.
- [RFC 1809] Using the Flow Label Field in IPv6.
- [RFC 1881] IPv6 Address Allocation Management.
- [RFC 1887] An Architecture for IPv6 Unicast Address Allocation.
- [RFC 1888] OSI NSAPs and IPv6.
- [RFC 1981] Path MTU Discovery for IP version 6.
- [RFC 2126] ISO Transport Service on top of TCP (ITOT).
- [RFC 2170] Application REQuested IP over ATM (AREQUIPA).
- [RFC 2185] Routing Aspects Of IPv6 Transition.
- [RFC 2292] Advanced Sockets API for IPv6.
- [RFC 2373] IP Version 6 Addressing Architecture.
- [RFC 2374] An IPv6 Aggregatable Global Unicast Address Format.
- [RFC 2375] IPv6 Multicast Address Assignments.
- [RFC 2401] Security Architecture for the Internet Protocol.
- [RFC 2450] Proposed TLA and NLA Assignment Rules.
- [RFC 2452] IP Version 6 Management Information Base for the Transmission Control Protocol.

RFC's

- [RFC 2454] IP Version 6 Management Information Base for the User Datagram Protocol.
- [RFC 2460] Internet Protocol, Version 6 (IPv6) Specification.
- [RFC 2461] Neighbor Discovery for IP Version 6 (IPv6).
- [RFC 2462] IPv6 Stateless Address Autoconfiguration.
- [RFC 2464] Transmission of IPv6 Packets over Ethernet Networks.
- [RFC 2465] Management Information Base for IP Version 6: Textual Conventions and General Group.
- [RFC 2467] Transmission of IPv6 Packets over FDDI Networks.
- [RFC 2470] Transmission of IPv6 Packets over Token Ring Networks.
- [RFC 2471] IPv6 Testing Address Allocation.
- [RFC 2472] IP Version 6 over PPP.
- [RFC 2473] Generic Packet Tunneling in IPv6 Specification.
- [RFC 2474] Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers.
- [RFC 2475] An Architecture for Differentiated Services.
- [RFC 2491] IPv6 over Non-Broadcast Multiple Access (NBMA) networks.

RFC's

- [RFC 2492] IPv6 over ATM Networks.
- [RFC 2497] Transmission of IPv6 Packets over ARCnet Networks.
- [RFC 2507] IP Header Compression.
- [RFC 2508] Compressing IP/UDP/RTP Headers for Low-Speed Serial Links.
- [RFC 2526] Reserved IPv6 Subnet Anycast Addresses.
- [RFC 2529] Transmission of IPv6 over IPv4 Domains without Explicit Tunnels.
- [RFC 2553] Basic Socket Interface Extensions for IPv6.
- [RFC 2590] Transmission of IPv6 Packets over Frame Relay Networks Specification.
- [RFC 2675] IPv6 Jumbograms.
- [RFC 2711] IPv6 Router Alert Option.
- [RFC 2732] Format for Literal IPv6 Addresses in URL's.
- [RFC 2765] Stateless IP/ICMP Translation Algorithm (SIIT).
- [RFC 2766] Network Address Translation - Protocol Translation (NAT-PT).
- [RFC 2767] Dual Stack Hosts using the "Bump-In-the-Stack" Technique (BIS).
- [RFC 2780] IANA Allocation Guidelines For Values In the Internet Protocol and Related Headers.

RFC's

- [RFC 2874] DNS Extensions to Support IPv6 Address Aggregation and Renumbering.
- [RFC 2893] Transition Mechanisms for IPv6 Hosts and Routers.
- [RFC 2928] Initial IPv6 Sub-TLA ID Assignments.
- [RFC 3041] Privacy Extensions for Stateless Address Autoconfiguration in IPv6
- [RFC 3053] IPv6 Tunnel Broker.
- [RFC 3056] Connection of IPv6 Domains via IPv4 Clouds.
- [RFC 3111] Service Location Protocol Modifications for IPv6.
- [RFC 3142] An IPv6-to-IPv4 Transport Relay Translator.
- [RFC 3146] Transmission of IPv6 Packets over IEEE 1394 Networks.
- [RFC 3178] IPv6 Multihoming Support at Site Exit Routers.

IPv6 Packet Header Format

Ethernet II Encapsulation

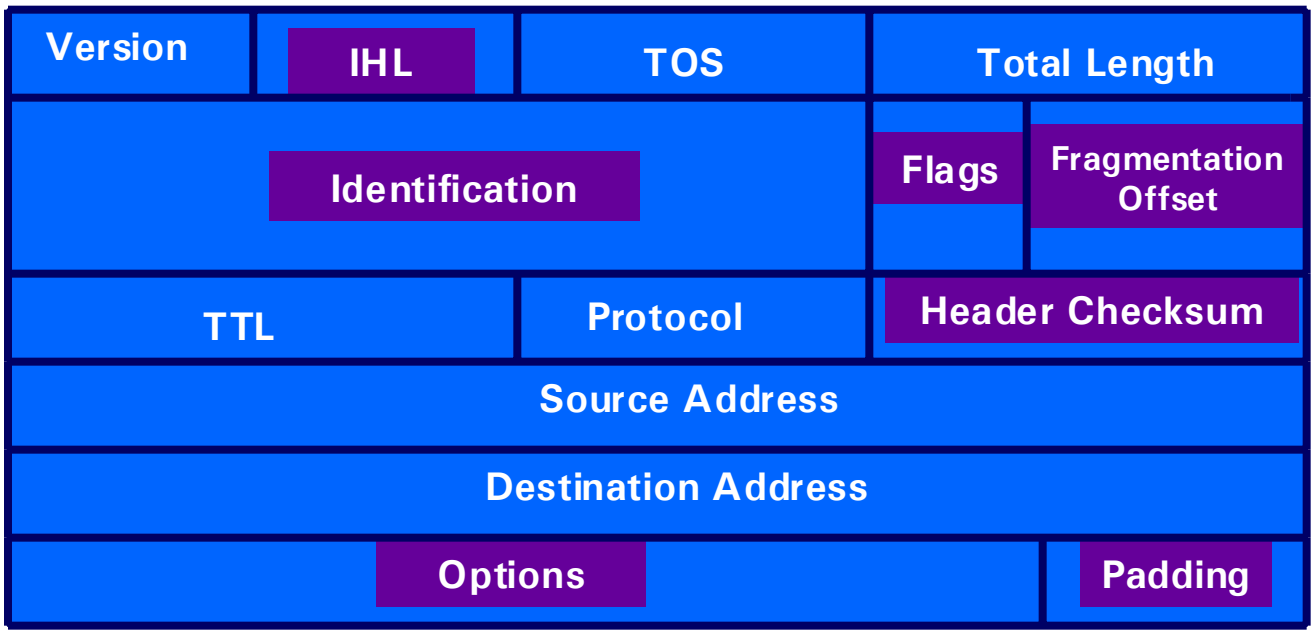
LINK Layer Frame

- **IPv4 is indicated by EtherType 0x800**
- ***IPv6 Packet size from 46 - 1,500 Bytes***

Header Changes IPv6-IPv4

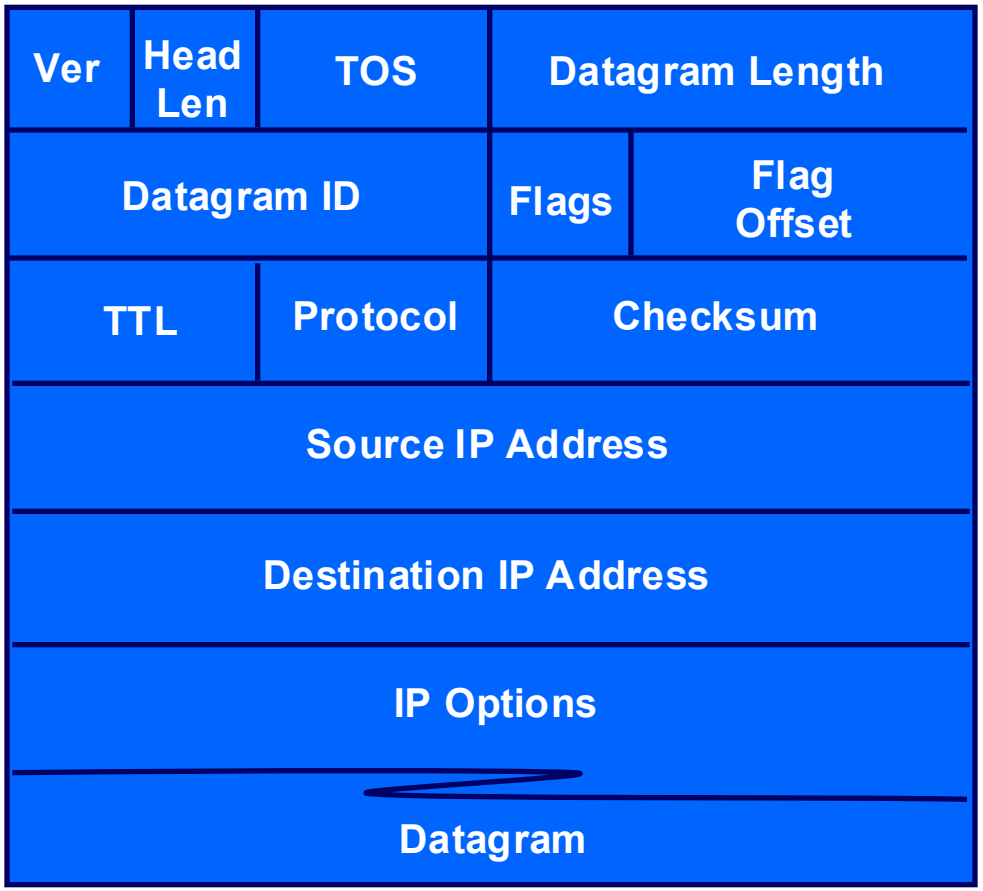
- TTL (Time To Live) = Hop Limit
- Protocol = Next header
- Precedence & TOS = Traffic Class
- Addresses increased from 32 bits to 128 bits

IPv4 Packet Header Format



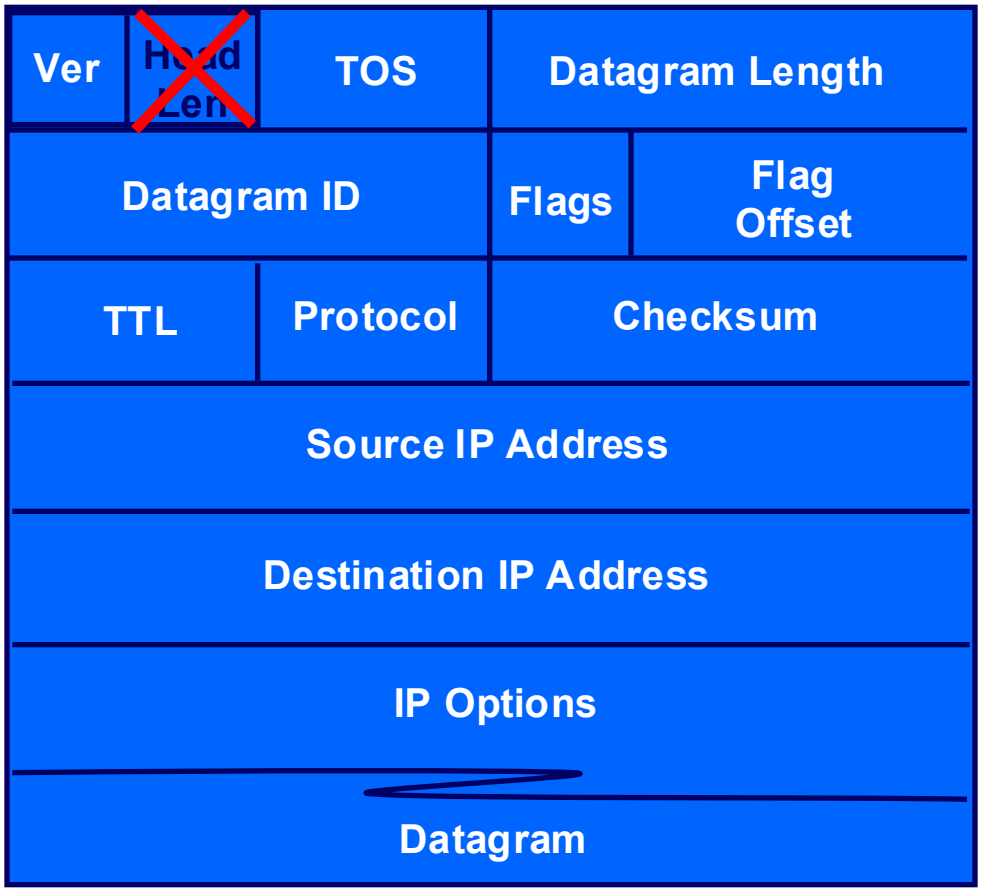
Shaded fields are not included in IPv6 header

Protocol Changes



Here is the traditional IPv4 Header. ***Watch the IPv6 Transformation!***

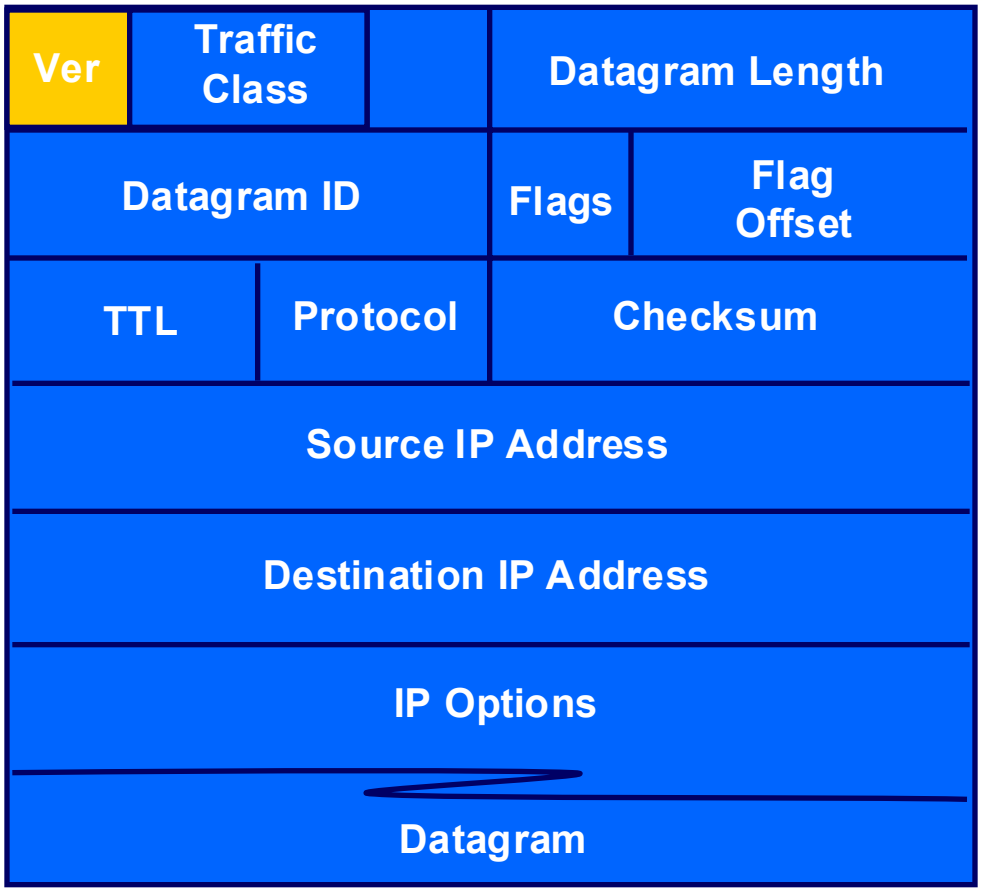
Protocol Changes



IPv4s "options" necessitated the Header Length field, as headers could be anywhere from 20 to 60 bytes.
Now there is no need.

Version Field Remains as-is, and header length field is gone.
IPv6 headers are all 40 bytes.

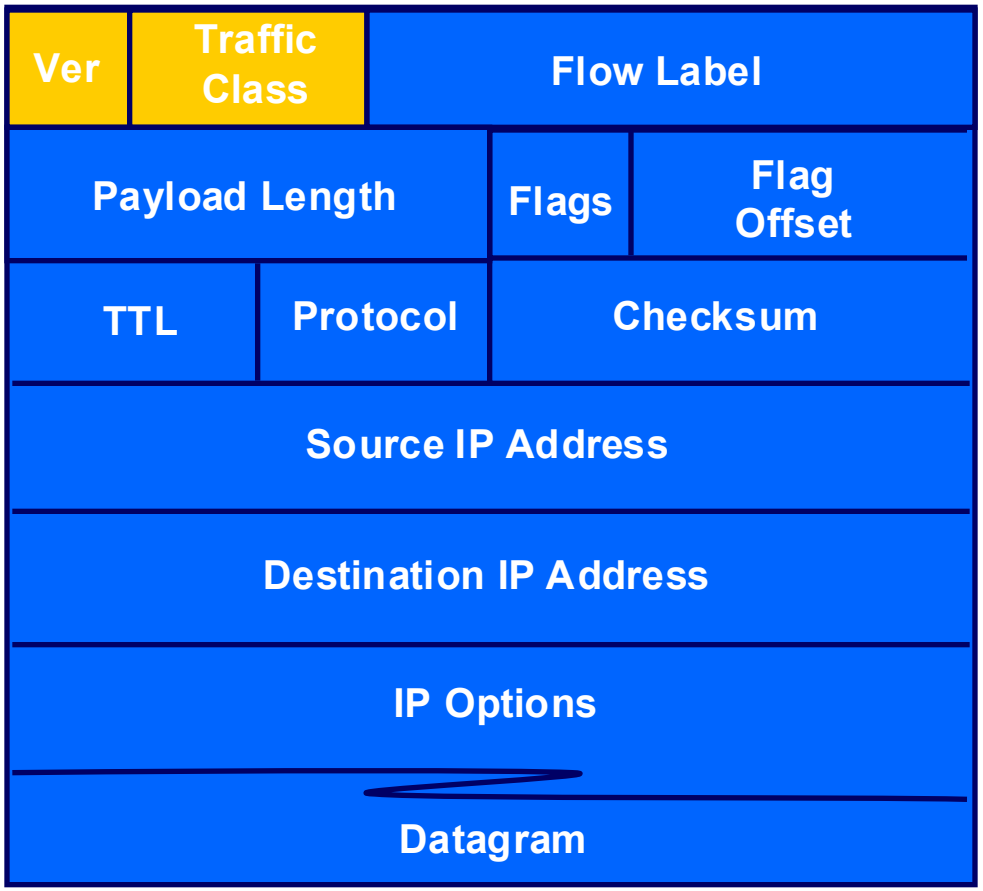
Protocol Changes



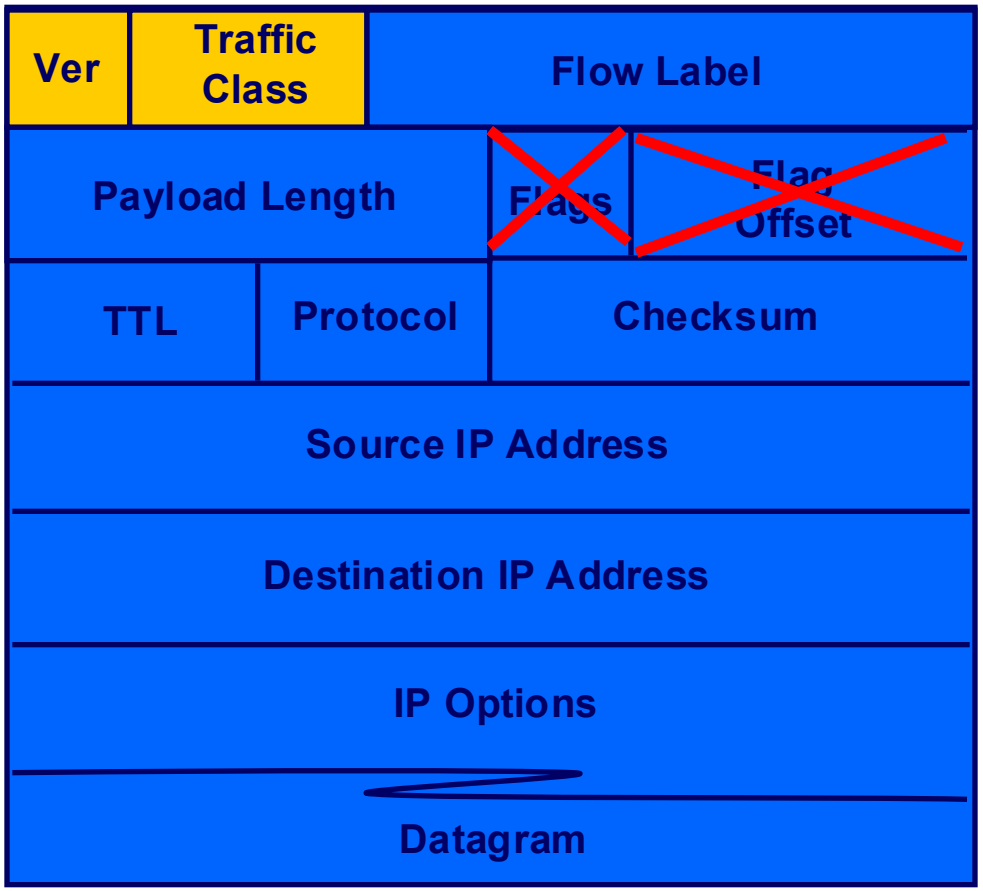
Traffic Class

Default value of this field is set to zero. It appears that the same DiffServ as IPv4 will be used.

Protocol Changes

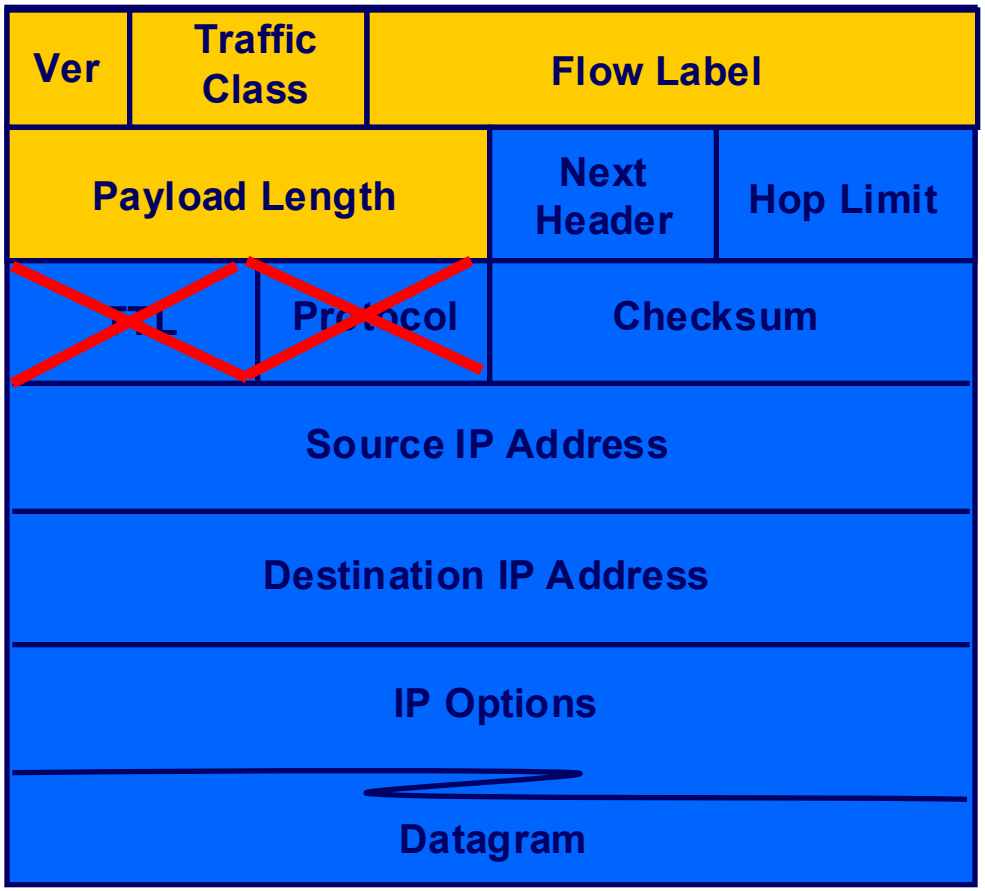


Protocol Changes



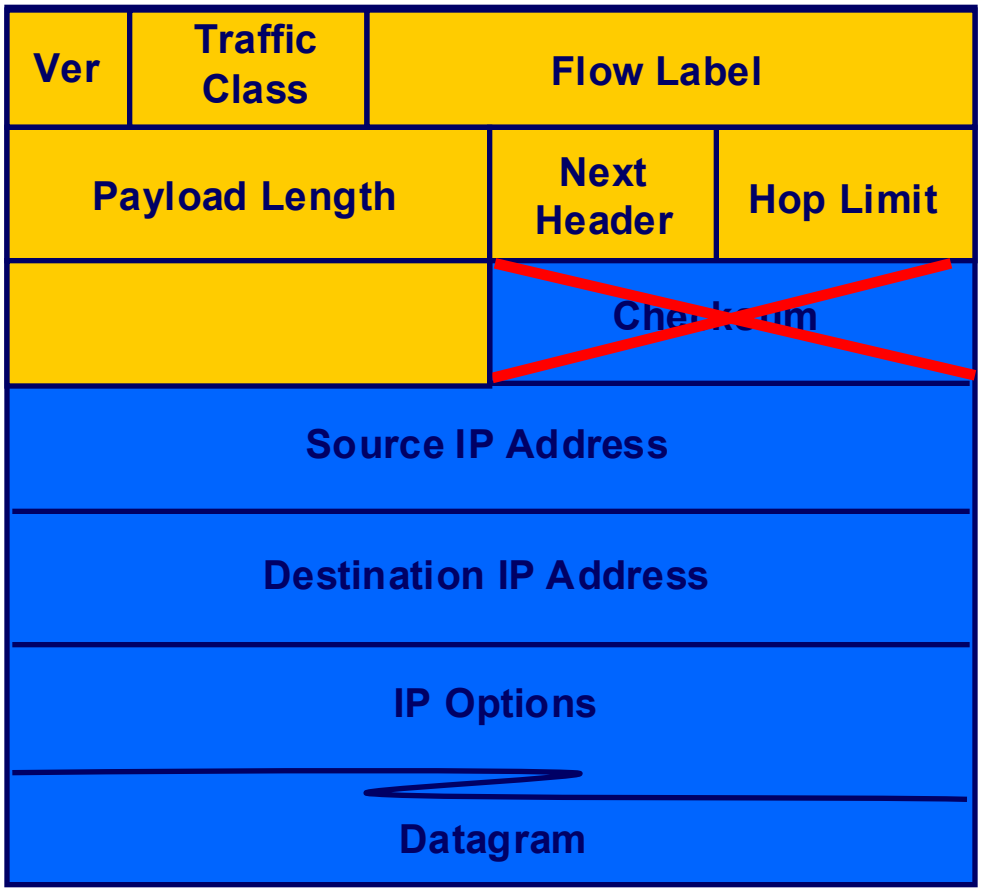
The “Flag” and “offset” were eliminated. IPv4 fragmentation was considered processor intensive.

Protocol Changes



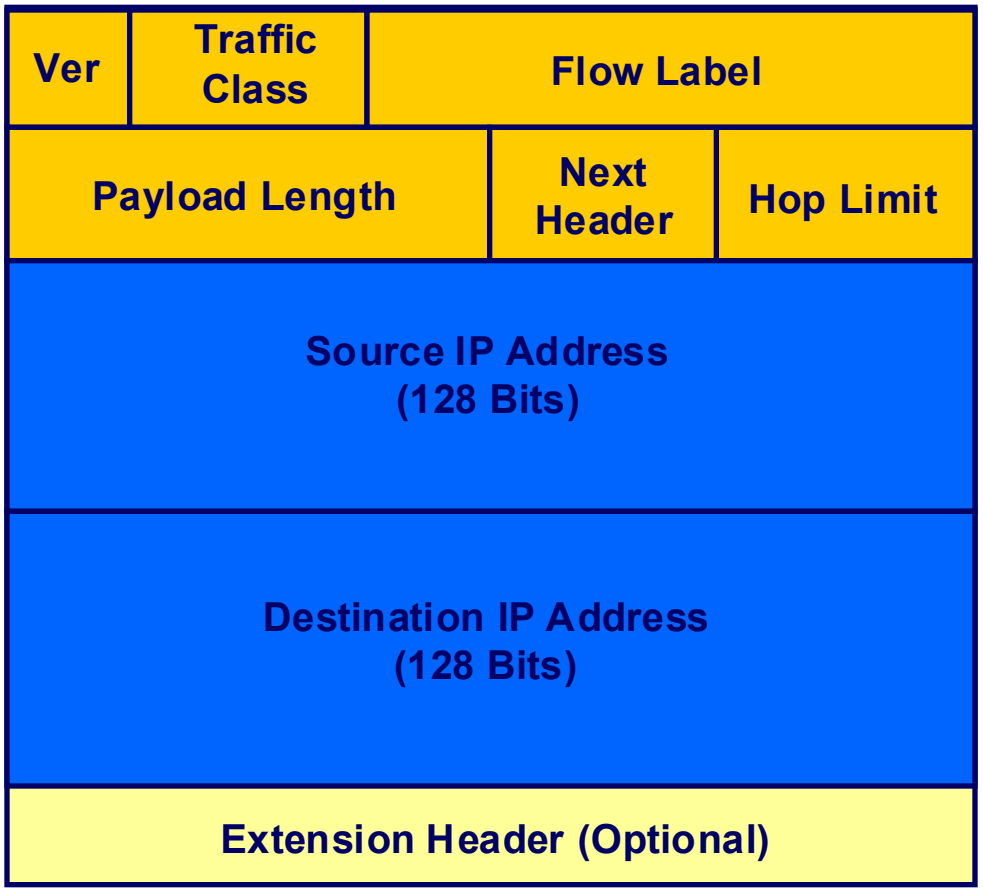
The “Next Header” field can indicate which type of extension header follows the IPv6 header. This overcomes the need for the IPv4 “Protocol” field. TTL is replaced by “Hop Limit.”

Protocol Changes



Since upper layer protocols like TCP and UDP calculate their own checksums, the IP header checksum was considered unnecessary. If content authentication is desired, the Authentication header is available in IPv6.

Protocol Changes



Finally, the main reason for IPv6's existence is the 128 bit addressing scheme. Extension headers are available for a number of different functions, but are not considered part of the IPv6 header itself.

IPv6 Addressing

IPv6 Address Syntax RFC 2373

- IPv6 Address in Binary form

00100001110110100000000011010011000000000000000000010111100111011
00000010101010100000000011111111111111100010100010011100010110 10

- The 128-bit address is divided along 16-bit boundaries:

0010000111011010	0000000011010011	0000000000000000	0010111100111011
0000001010101010	0000000011111111	1111111000101000	1001110001011010

- Each 16-bit block is converted to hexadecimal and delimited with colons:

  21DA:00D3:0000:2F3B:02AA:00FF:FE28:9C5A

- Leading zeros may be removed within each block:

  21DA:D3:0:2F3B:2AA:FF:FE28:9C5A

IPv6 Address Syntax

Compressing Zeros

- A long sequence of zeros may be simplified by "::"

  Known as a double-colon.

Example:

FF02:0:0:0:0:0:0:2 *is equal to* FF02::2

- Zero compression can only be used for a contiguous series of 16-bit blocks.

Example:

FF02:30:0:0:0:0:0:5 can't be expressed as FF02:3::5

IPv6 Address Syntax

- To determine how many 0 bits are represented by the "::". Subtract the number of blocks in the compressed address from 8 and then multiply by 16.
- Example
  FF02::2 The number of bits represented by "::" is 96.
  96=(8-2)X16.
- Zero compression can only be used once in a given address.

IPv6 Address Syntax

IPv6 Prefix

- Indicates the fixed part of the address. Same as CIDR in IPv4.
- Example:
  21DA:D3::/48 is a route prefix.
  21DA:D3:0:2F3B::/64 is a subnet prefix.

IPv6 Address Syntax

- When in mixed IPv4 and IPv6 environments, can be written as x:x:x:x:d.d.d.d where “d” is the actual decimal value of the IP address.

  (IPv4 compatible) 0:0:0:0:0:0:13.211.45.138 or
  (IPv4 mapped) 0:0:0:0:0:FFFF:13.211.45.138

- **Compatible addresses are used by devices that intend to tunnel IPv6 packets through IPv4 routers. Mapped addresses (All Fs in the 6th 16-bit grouping) are used by IPv6 devices sending to nodes *only* supporting IPv4.**

IPv6 Address Types

Unicast

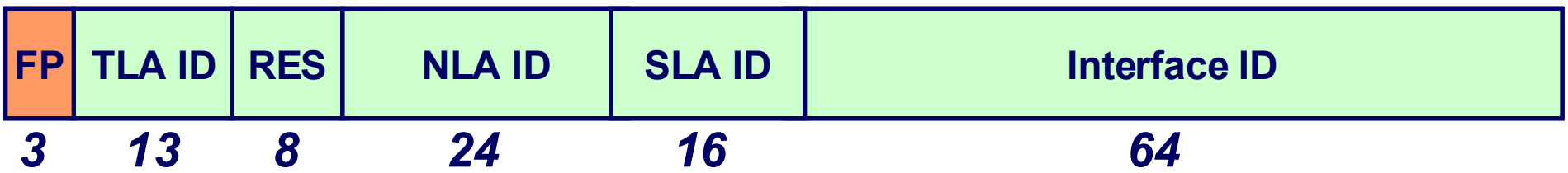
An identifier for a single interface. A packet sent to a unicast address is delivered to the interface identified by that address.

- Aggregatable global unicast addresses
- Link-local addresses
- Site-local addresses
- Special addresses

IPv6 Addressing Hierarchy RFC 2450

Aggregatable Global Unicast Addresses

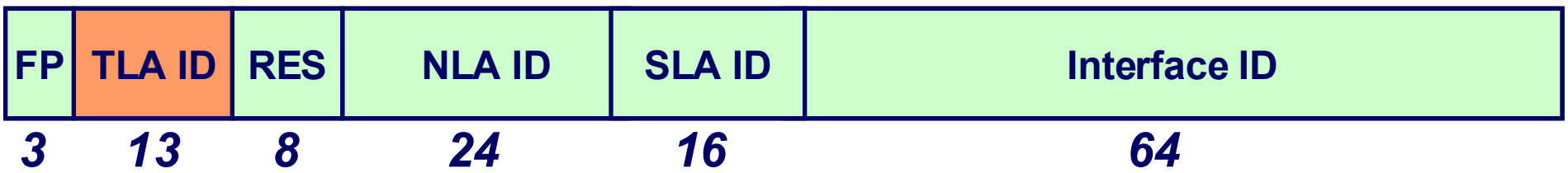
- *Format Prefix* is used to indicate where an address belongs in the IPv6 address space.
  - **001 = Aggregatable Global Unicast**
  - **111 = Multicast OR Private**
  - **More will be assigned as the protocol matures**



IPv6 Addressing Hierarchy

Aggregatable Global Unicast Addresses

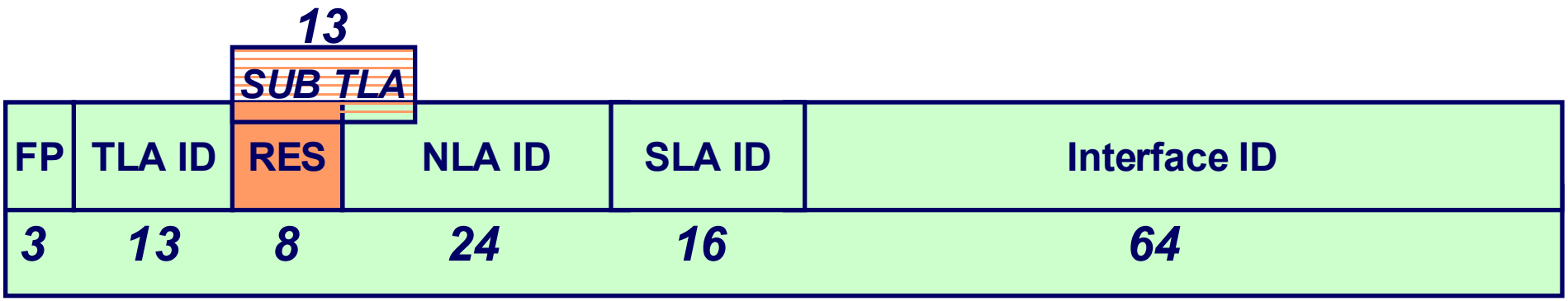
- The *Top Level Aggregation IDentifier* contains the highest level routing information for the address. This space will typically be assigned to transit providers, not leaf sites. At 13 bits, this assures that there can only be 8192 different top-level routes in the internet.



IPv6 Addressing Hierarchy

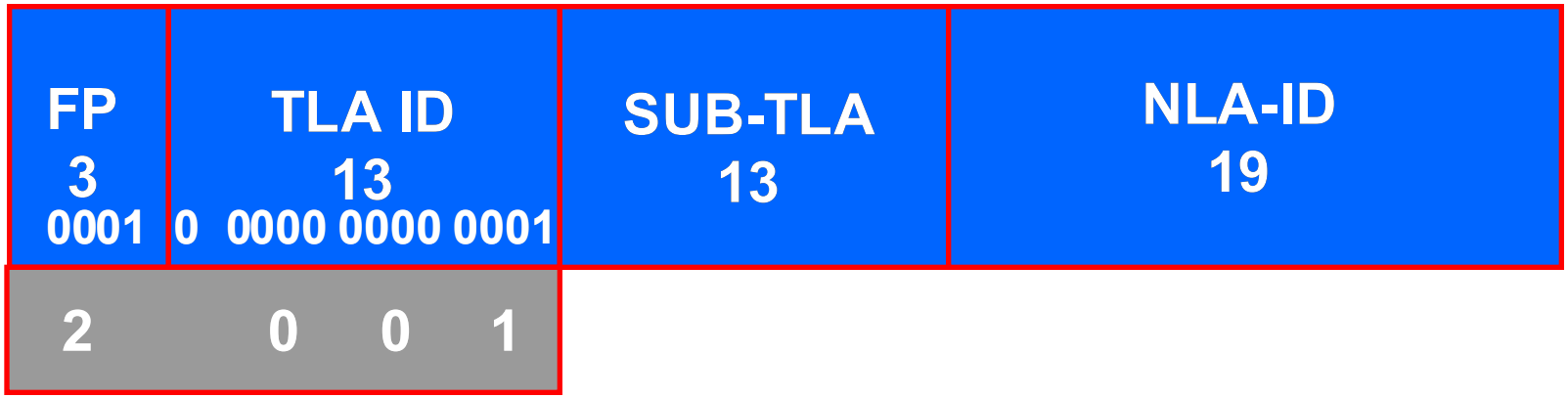
Aggregatable Global Unicast Addresses

- These 8 bits are *RES*erved for future use.
- RFC 2450 proposes the first addresses allocated will have TLA ID 0x0001as a “litmus test” of sorts. The *RES*erved field and 5 bits of the NLA ID will be used as a “Sub TLA” identifier. Registries who can utilize 90% of the NLA IDs under their Sub TLA ID will be granted their own TLA ID.



http://www.ietf.org/rfc/rfc2450.txt?number=2450

RFC 2450 Interim Rules

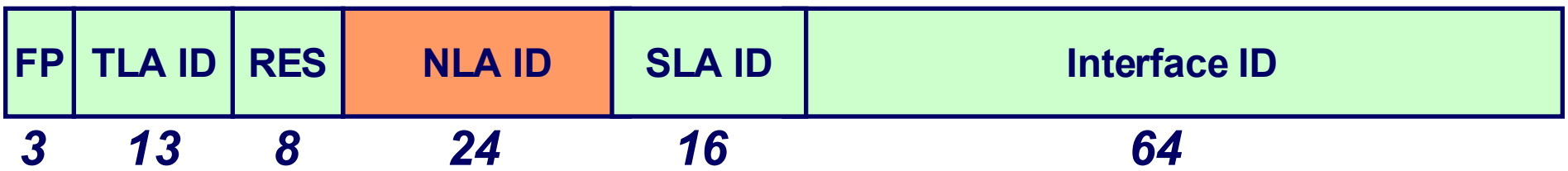


- FP = 001 = Format Prefix
- TLA ID = 0x0001 = Top-Level Aggregation Identifier
- Initially many addresses will start with 2001
- 6bone starts with 3FFE

IPv6 Addressing Hierarchy

Aggregatable Global Unicast Addresses

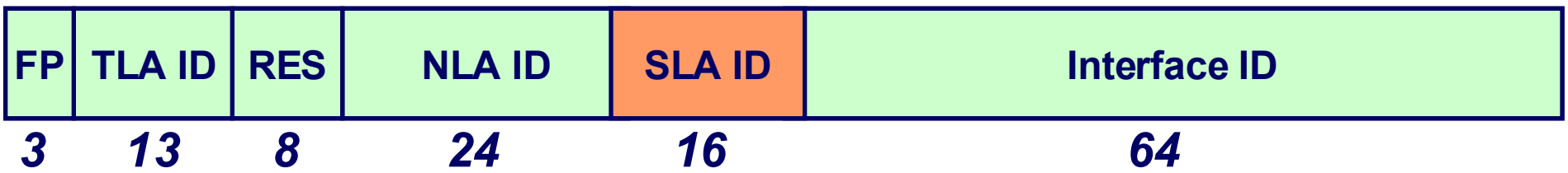
- The *N*ext *L*evel *A*ggregation *ID*entifier will be used by the organizations that control TLA IDs. These transit providers will be allowed to create their own address hierarchy out of these 24 bits.



IPv6 Addressing Hierarchy

Aggregatable Global Unicast Addresses

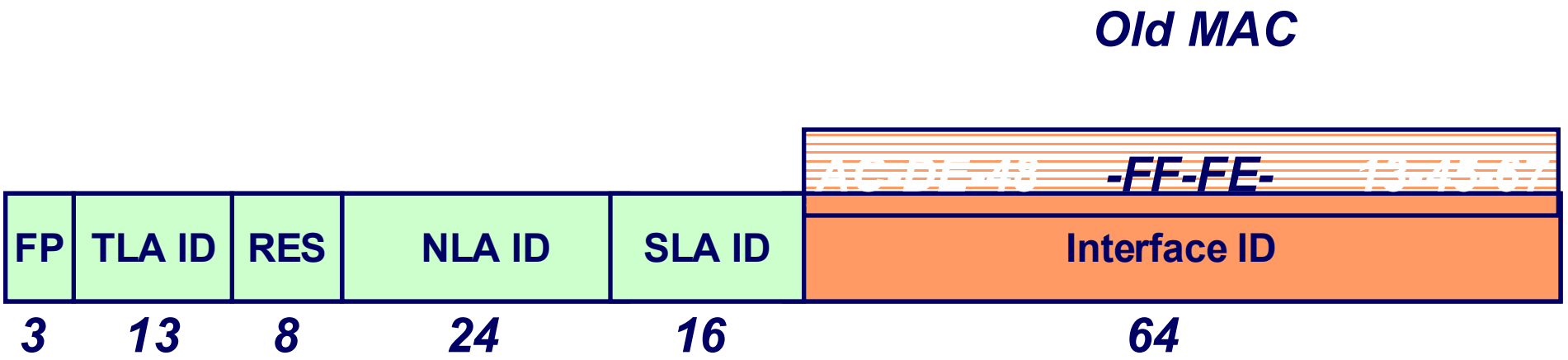
- The *Site Level Aggregation IDentifier* will be allocated to organizations for their internal network structure. The 16 available bits allow for as many as 65,535 subnets.



IPv6 Addressing Hierarchy

Aggregatable Global Unicast Addresses

- V6 host addresses are recommended (not mandated!) to be built from so-called EUI64 addresses. EUI64 addresses are -- as the name says -- 64-bits wide, and derived from MAC addresses of the underlying network interface. For example, with Ethernet, the 6-byte (48-bit) MAC address is filled with the hex bits "fffe" in the middle -- the MAC address



http://standards.ieee.org/regauth/oui/tutorials/EUI64.html

Anycast

RFC 2461

Anycast

An identifier for a set of interfaces. A packet sent to an anycast address is sent to the “nearest” one based on the routing protocols’ measure of distance.

- DNS is one example of this. The host sending the message doesn’t care which DNS server gets the message, it just wants a reply.

Multicast IPv6 Addresses RFC 2373

- An identifier for a set of interfaces. A packet sent to a multicast address is delivered to all identified by that address.

  - Only the Transient (T) flag has been defined. When the low order bit is set to 0 this indicates that it is a well known address assigned by IANA
  - T = 0 = Permanently assigned Multicast address
  - Group ID identifies the multicast group, either permanent or transient, within the given scope

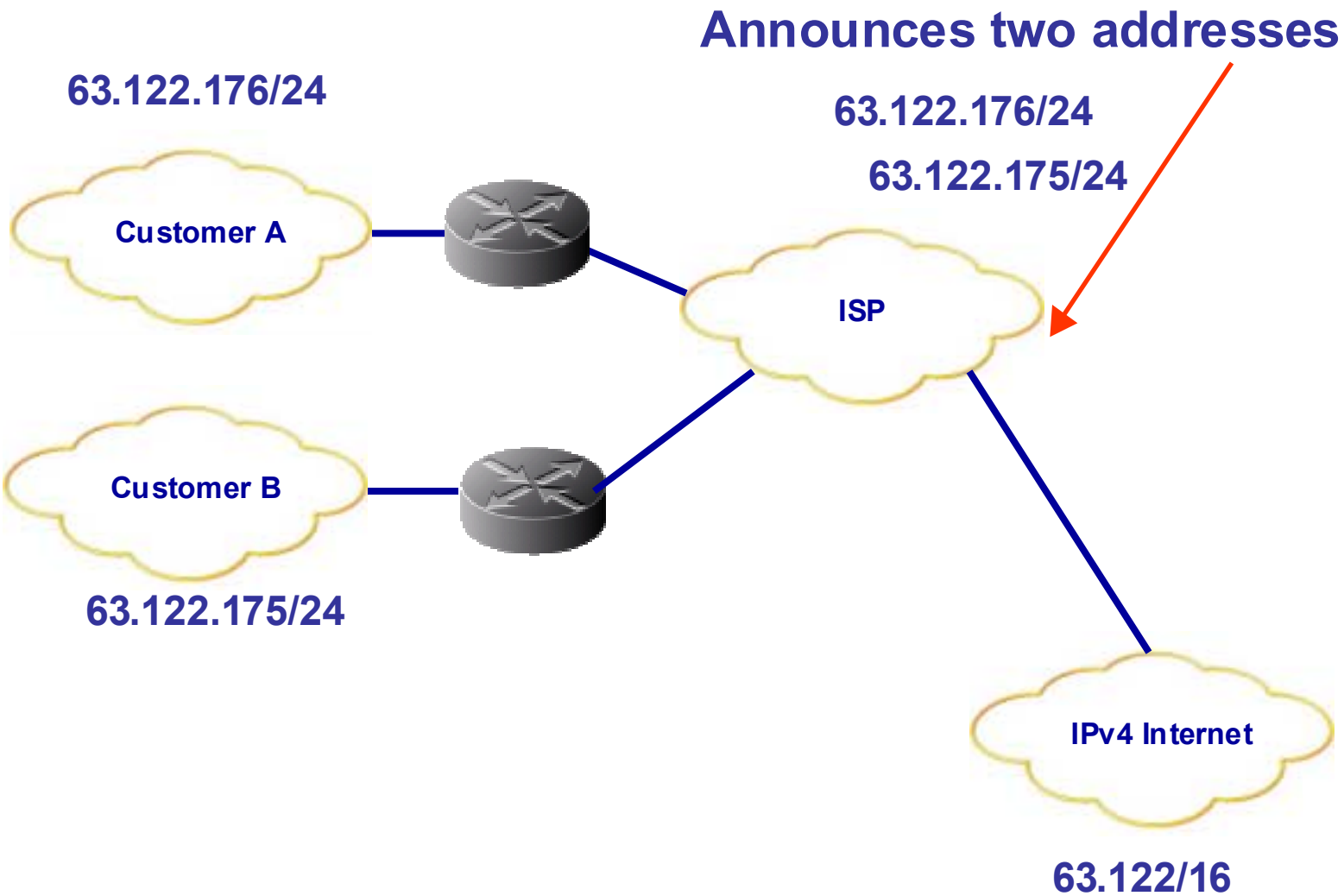
Multicast IPv6 Addresses

- Scope
- Indicates the scope of the IPv6 internetwork for which the multicast traffic is intended.

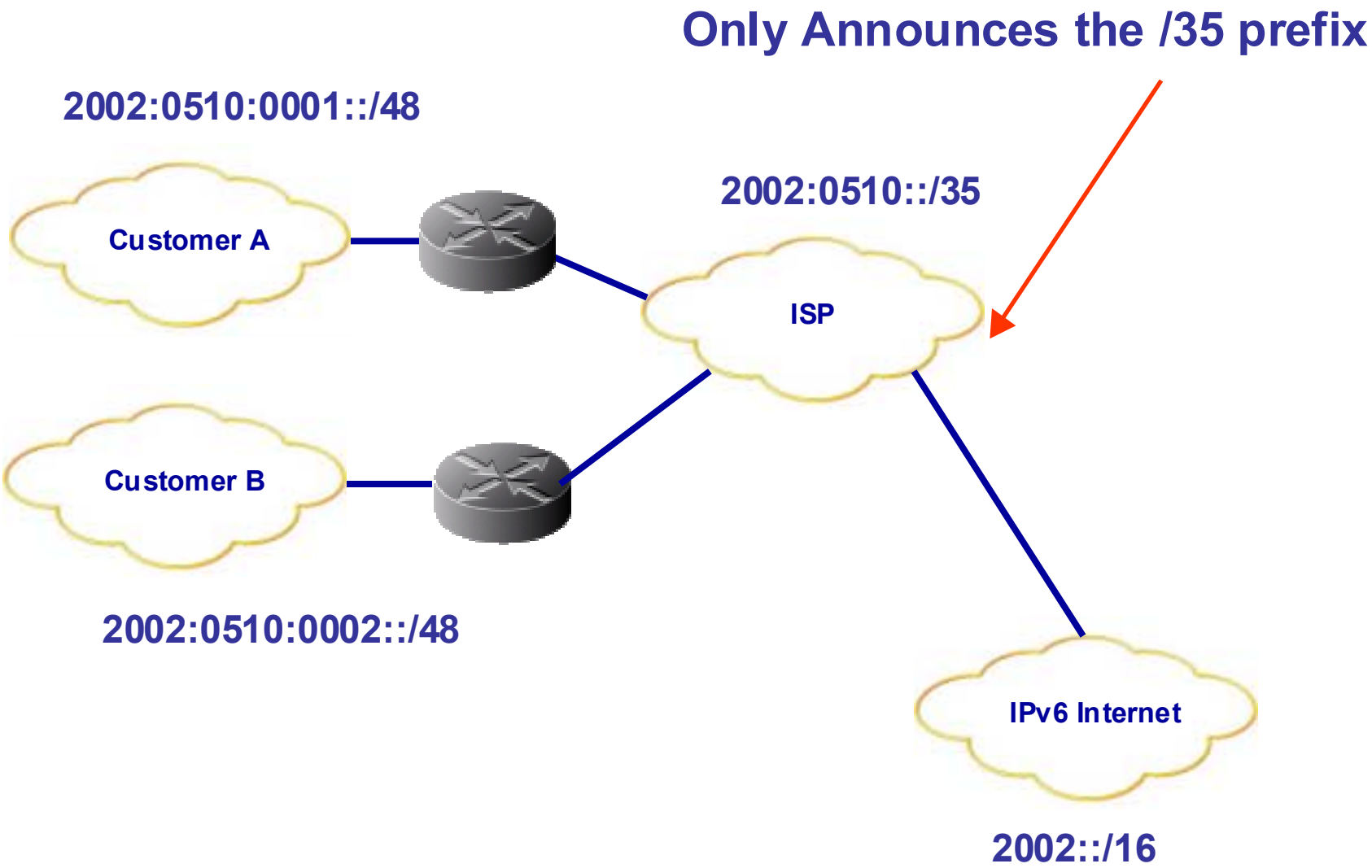
Value	Scope
0	Reserved
1	Node-Local Scope
2	Link-Local Scope
5	Site-Local Scope
8	Organization-Local Scope
E	Global Scope
F	Reserved

Traffic with the Multicast address FF02::2 has a link - local scope. Targeted for all routers

IPv4 Prefix Use

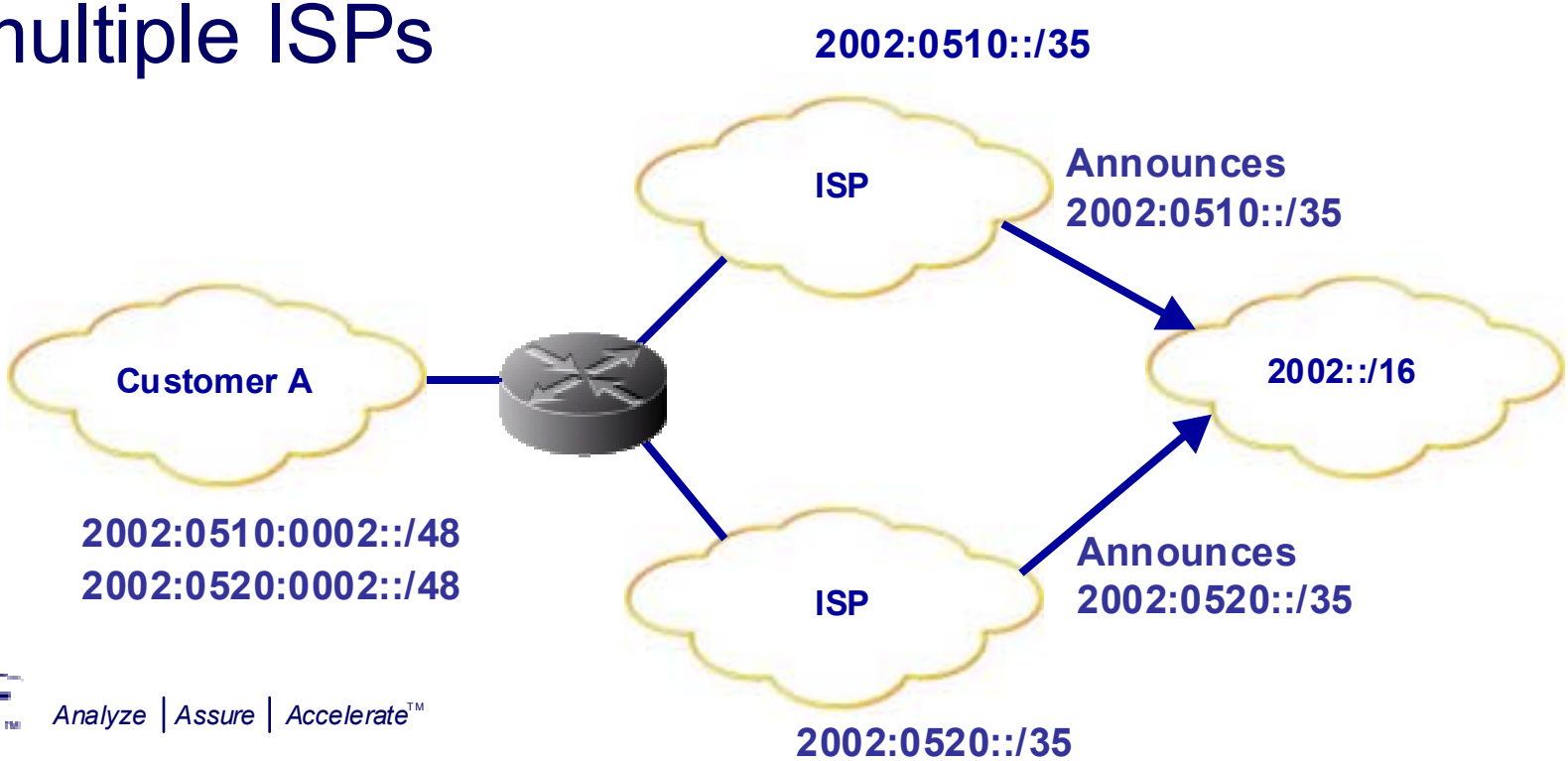


IPv6 Prefix Aggregation



IPv6 Multihoming

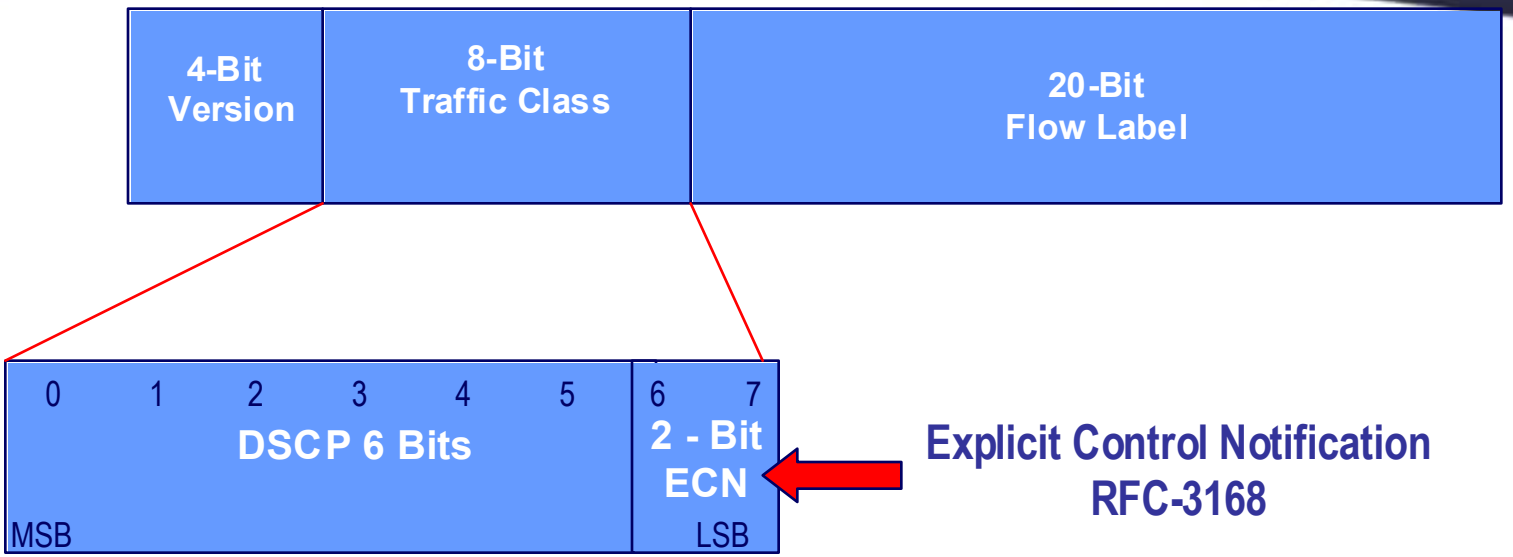
- Multiple IPv6 prefixes can be assigned to networks and hosts.
- Having multiple prefixes assigned to a network makes it easy for that network to connect to multiple ISPs



QOS Using Traffic Class

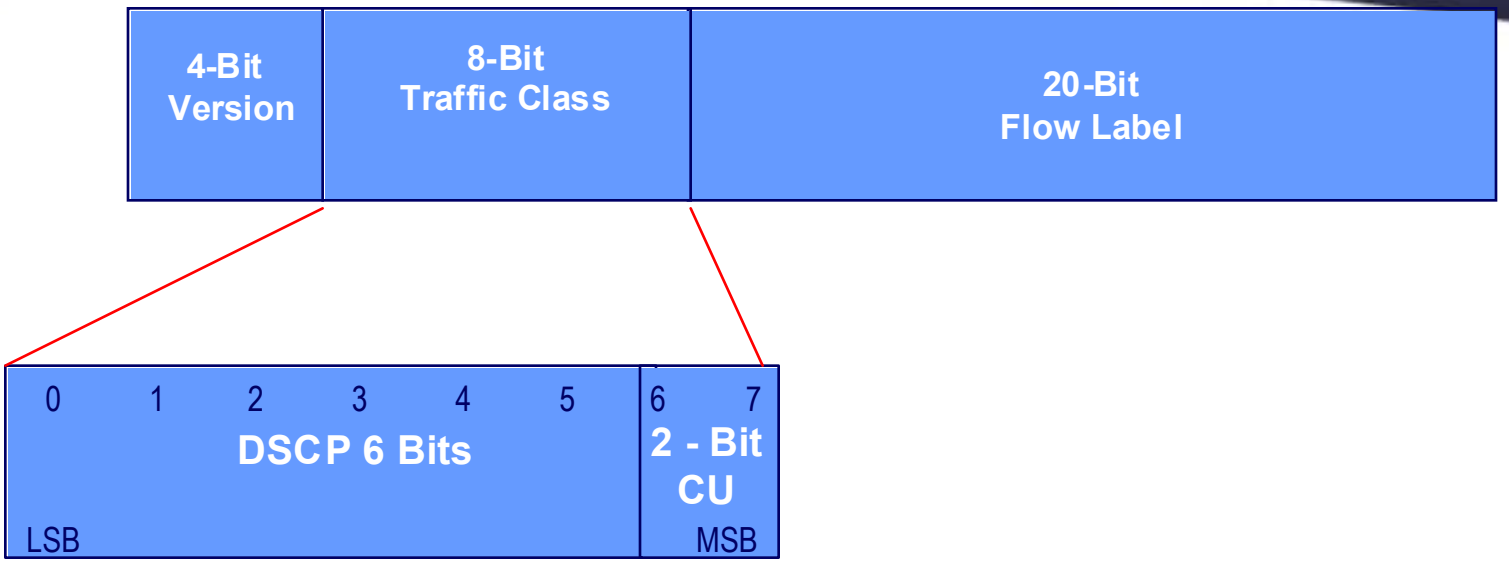
(Same as IPv4)

IP Differentiated Services RFC-2474



- Known as "DS" field. "Differentiated Services"
- Class Selector Code Points = Bits 0, 1 & 2
- Six Bits are used as a code point (DSCP)
- Provides 64 distinct code points.
- Bits 3, 4 & 5 must = 0 (For RFC-2474)

RFC-2597 Assured Forwarding (AF)

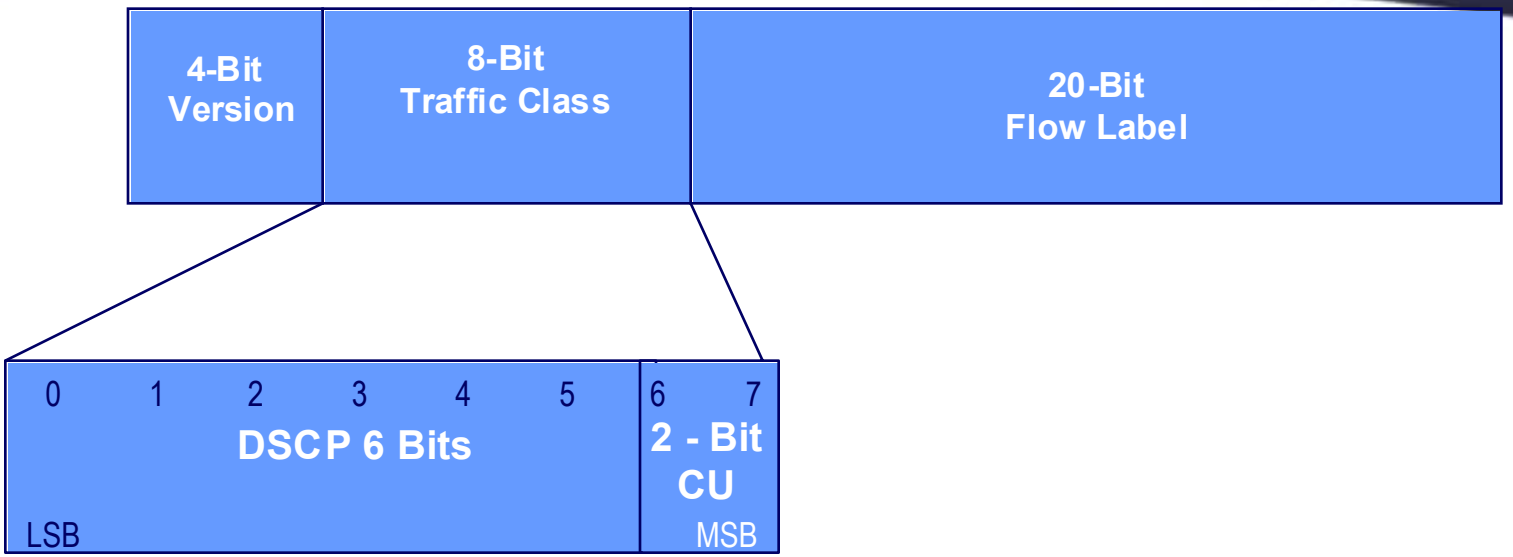


- Four AF classes are defined
- Each DS node allocates resources for each AF
- Each class is is marked with one of three drop precedence

RFC-2597 Assured Forwarding (AF)

Precedence	Class 1	Class 2	Class 3	Class 4
Low Drop	001010	010010	011010	100010
Medium Drop	001100	010100	011100	100100
High Drop	001110	010110	011110	100110

RFC-2598 Expedited Forwarding (EF)



- Low loss, Low latency, Low jitter, Assured bandwidth
- Appears as a "virtual leased line" (VLL)
- Also known as "Premium service"
- Code point value for this service = 101110

IPv4 Vs IPv6 Efficiency

- IPv4 header includes a checksum which must be computed by each intervening node on a per packet basis.
- IPv6 resigned this time consuming and costly processing mechanism because most higher level protocols have their own checksum control mechanism.
- IPv4 specification, each router along the transmission path must process the variable length Option Field, again, on a per packet basis even though an option might be effectively used only by the end hosts.
- IPv6, the new concept of an ordered linked list of Extension Headers ensures that routers process only the options necessary for correct operation.

QOS and the Flow Label

- Flow Label Not fully defined
- Draft-ietf-ipv6-flow-label-02.txt
- It appears to be well suited to RSVP type applications.

QOS and the Flow Label

- IPv4 has no implicit support for flows. Thus, intervening routers rely on transport protocol or application level information to identify flows. The fact that a router which is supposed to process data only at the network layer, according to the OSI reference model, requires information from the transport or application protocol (i.e. socket ports) to map packets on to their reserved resources, introduces what is known as the Layer Violation Problem.
- Another disadvantage caused by the dependency on transport or application protocol information is that IP-level security techniques, such as IPSEC , cannot be used in conjunction with RSVP. These mechanisms might encrypt the entire transport header hiding the port numbers of data packets from intermediate routers.

The Benefits of the Flow Label

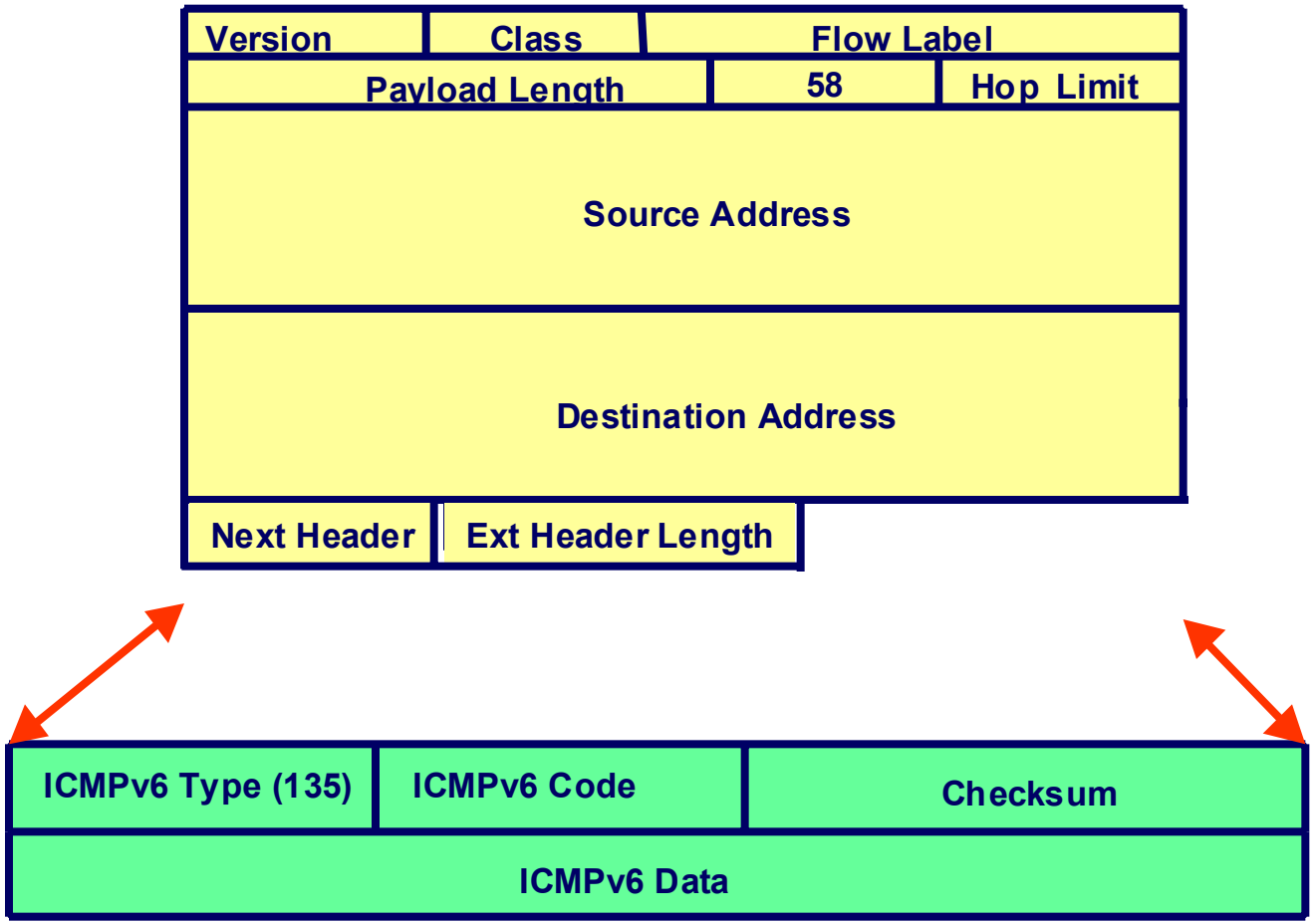
- Routers need to perform packet classification only for packets with a non-zero flow labels; and second, the processing time of the IPv6 header, especially the extension headers, is greatly reduced due to the fact that all packets from the same flow must have identical extension headers. As a result, routers along a path have to process the headers only on a per flow basis rather than a per packet basis.

IPv6 Neighbor Discovery Protocol

Neighbor Discovery

- Replaces IPv4 ARP, plus new features
- Uses Internet Control Message Protocol (ICMPv6) messages
- Used to:
  - Find link- layer address of neighbor
  - Find neighboring routers
  - Actively keep track of neighbor reachability
  - Send network information from routers to hosts
  - Protocol used for host autoconfiguration

Neighbor Discovery (RFC 2461)



Neighbor Discovery

How does it work?

Fast Facts On Neighbor Discovery

- Any node supporting IPv6 and multicast should support neighbor discovery.
- Uses the all nodes multicast FF02::1 or all routers multicast FF02::2. ← ***ICMP messages can only be transmitted if the host knows the MAC of the destination. This is solved by using a multicast transmission***
- Both hosts and routers use ND to determine the link-layer addresses for neighbors known to reside on attached links and to quickly purge cached values that become invalid.
- Hosts use ND to find neighbor routers willing to forward packets on their behalf.
- Verifies reachability of a neighbor
- Keeps track of neighboring routers

Duplicate Address Detection

- Before the link-local address can be assigned to an interface and used, however, a node must attempt to verify that this "tentative" address is not already in use by another node on the link. Specifically, it sends a Neighbor Solicitation message containing the tentative address as the target. If another node is already using that address, it will return a Neighbor Advertisement saying so. If another node is also attempting to use the same address, it will send a Neighbor Solicitation for the target as well. The exact number of times the Neighbor Solicitation is (re)transmitted and the delay time between consecutive solicitations is link-specific and may be set by system management.
- If a node determines that its tentative link-local address is not unique, autoconfiguration stops and manual configuration of the interface is required. To simplify recovery in this case, it should be possible for an administrator to supply an alternate interface identifier that overrides the default identifier in such a way that the autoconfiguration mechanism can then be applied using the new (presumably unique) interface identifier. Alternatively, link-local and other addresses will need to be configured manually.

Duplicate Address Detection

Neighbor Solicitation Message

Unspecified Address = :: (source)	Tentative Address = B (Destination)

ICMPv6 Type =135
Src = Tentative Address
Dst = Tentative Address

If address already exists, a Neighbor Advertisement will come back.

FE02::1 (Multicast)	Source Address = B

ICMPv6 Type =136
Src = B
Data =Link layer address of B

Neighbor Discovery

Neighbor Solicitation Message (RFC 2461) Step 1

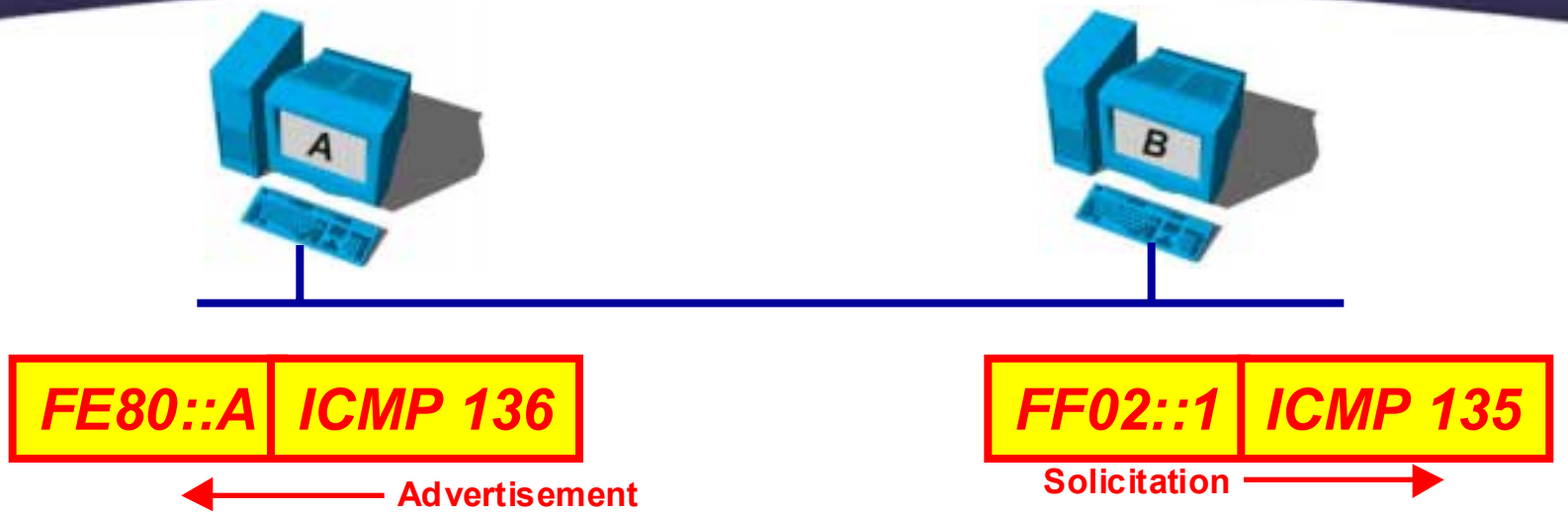
FF02::1	ICMP 135

Solicitation →

ICMPv6 Type =135
Src = A
Dst = Solicited-node multicast of B
Data =Link layer address of A
Query = what is your link address?

Neighbor Discovery

Neighbor Advertisement Message (RFC 2461) Step 2



ICMPv6 Type =135
Src = A
Dst = Solicited-node multicast of B
Data =Link layer address of A
Query = what is your link address?

ICMPv6 Type =136
Src = B
Dst = A
Data =Link layer address of B

Neighbor Discovery

Neighbor Solicitation Message (RFC 2461)

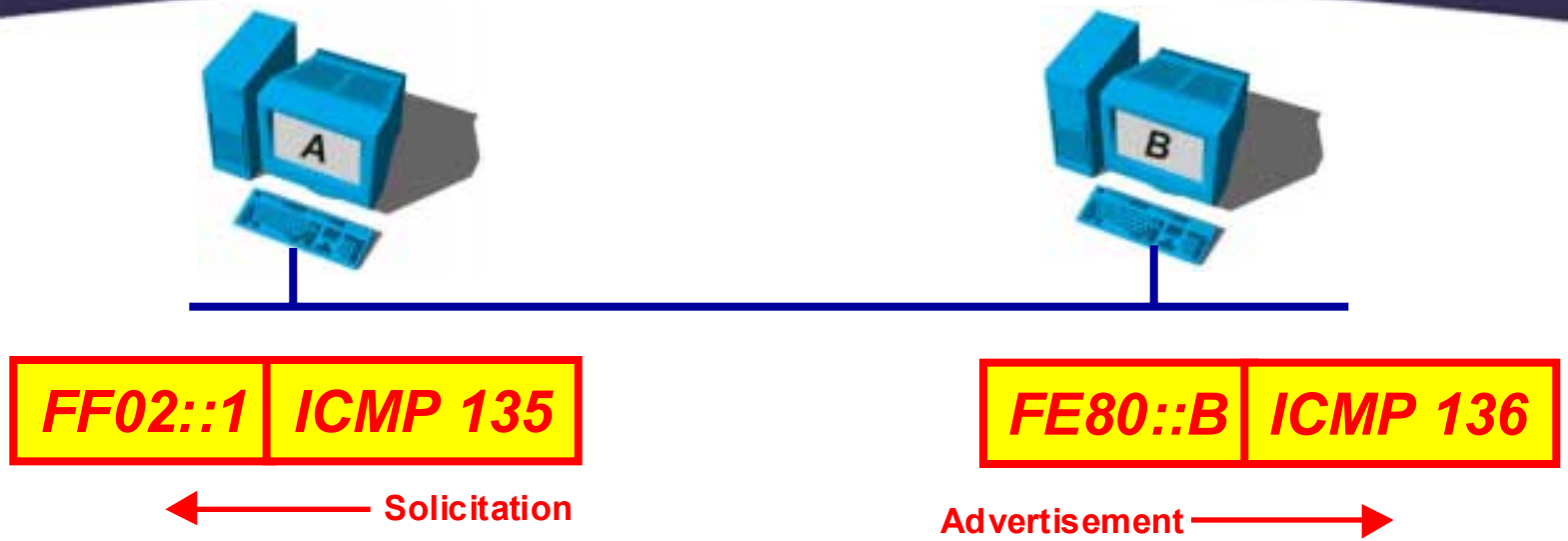
Step 3

FF02::1	ICMP 135

Solicitation

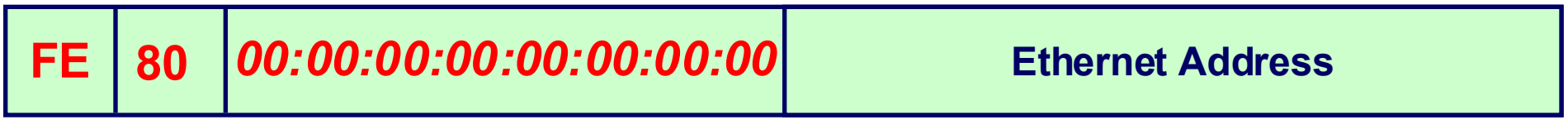
Neighbor Discovery

Neighbor Advertisment Message (RFC 2461) Step 4



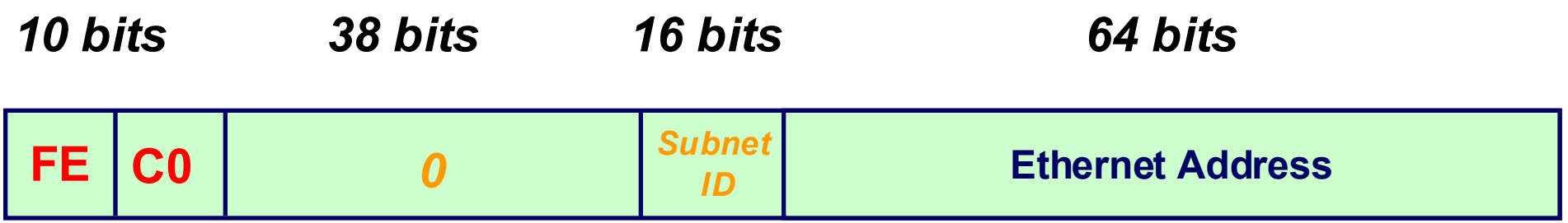
A and B can now exchange packets on this link
They are in the "REACHABLE" state.

Link Local Address



- The Link Local Address is the Ethernet address appended to the IPv6 address.
- Link- Local addresses are designed to be used for addressing on a single link for the purpose such as auto-address configuration, neighbor discovery, or when no routers are present.
- Routers must not forward any packets with Link-Local source or destination addresses to other links.

Site Local Address



- Site- Local addresses are designed to be used for addressing inside of a site without the need for a global prefix.
- Routers must not forward any packets with Site-Local source or destination addresses outside of the site.

Router Discovery

How does it work?

Unsolicited Router Advertisement

- ICMPv6 Type = 134
- Src = Router
- Dst = all nodes multicast address (FF02::1).

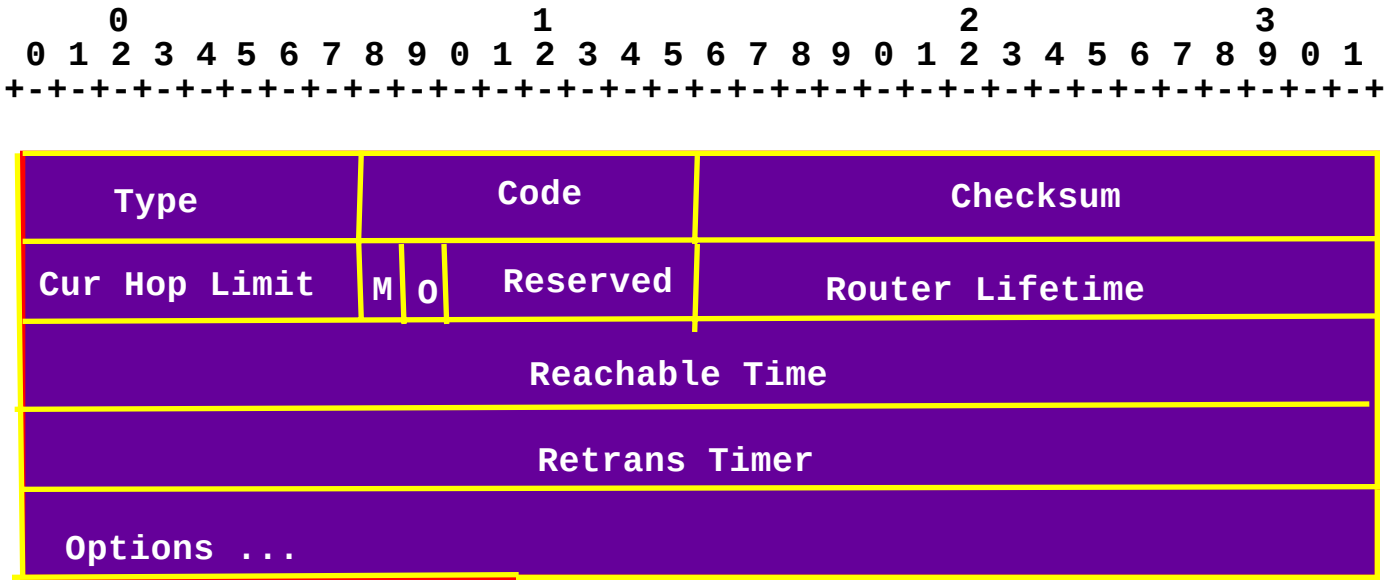
Unsolicited Router Advertisement

- ICMPv6 Type = 134
- Src = Router
- Dst = all nodes multicast address (FF02::1).

Router advertisement messages typically include the following:

- On - link IPv6 prefixes that nodes on the local link can use to automatically configure their IPv6 addresses.
- Lifetime information for each prefix included in the advertisement.
- Default router information (whether the router sending the advertisement should be used as a default router and, if so, the amount of time (in seconds) the router should be used as the default router).
- Additional information for hosts, such as the hop limit and MTU a host should use in packets that it originates.

Format for Router Advertisement Message



Format for Router Advertisement Message

Type	08 bit	134
Code	08 bit	0
Checksum	16 bit	
Cur Hop Limit	08 bit	
Managed Address Configuration flag	01 bit	
Other Stateful Configuration flag	01 bit	
Reserved	06 bit	
Router Lifetime	16 bit	
Reachable Time	32 bit	
Retrans Timer	32 bit	
Options	xxxxx	

Stateless Autoconfiguration

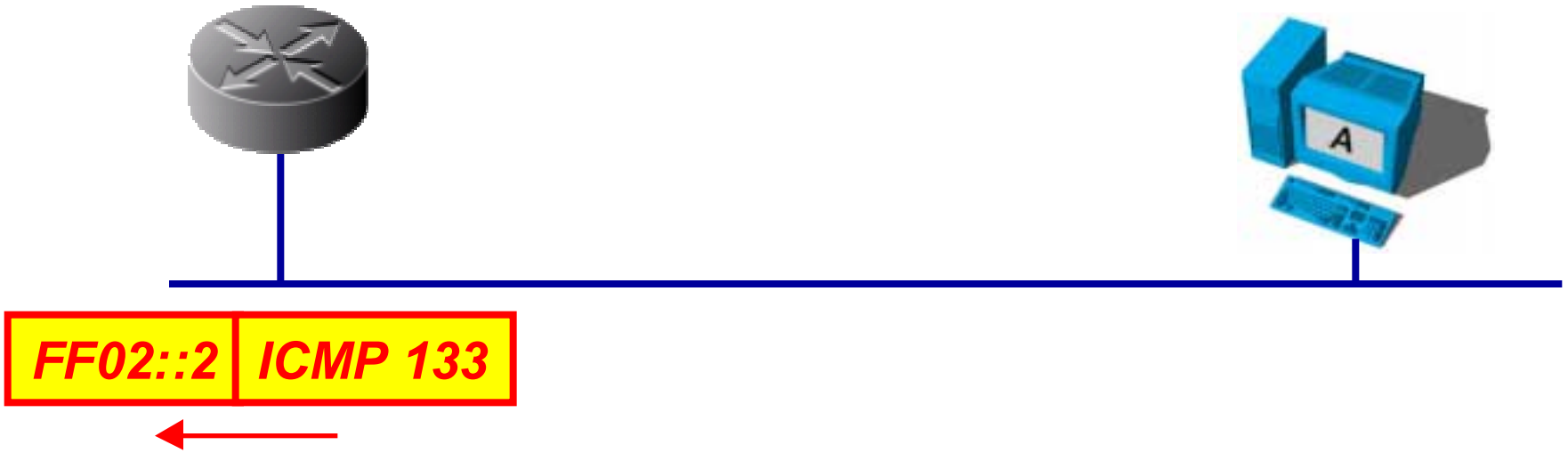
- Only used by hosts. Since hosts use information sent in router advertisements, the routers must be configured by other means.
- The stateless approach is used when a site is not particularly concerned with the exact addresses hosts use, so long as they are unique and properly routable.
- The stateful approach is used when a site requires tighter control over exact address assignments.
- Both stateful and stateless address autoconfiguration may be used simultaneously.
- The site administrator specifies which type of autoconfiguration to use through the setting of appropriate fields in Router Advertisement messages
- A host forms a link-local address by appending its interface identifier to the link-local prefix.

Router Discovery

- Each router periodically multicasts a Router Advertisement packet announcing its availability.
- A host receives Router Advertisements from all routers, building a list of default routers.
- Routers generate Router Advertisements frequently enough that hosts will learn of their presence within a few minutes, but not frequently enough to rely on an absence of advertisements to detect router failure; a separate Neighbor Unreachability Detection algorithm provides failure detection

Router Discovery
Router Solicitation Message

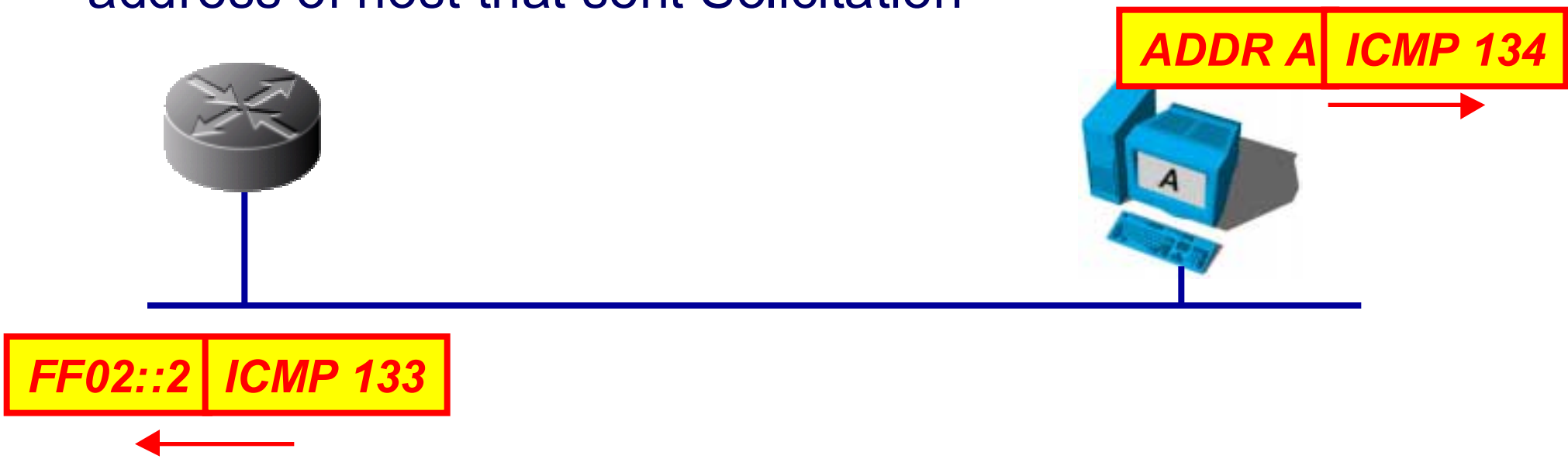
- ICMPv6 Type = 133
- Src = Host link local address
- Dst = all routers multicast address



Router Advertisment

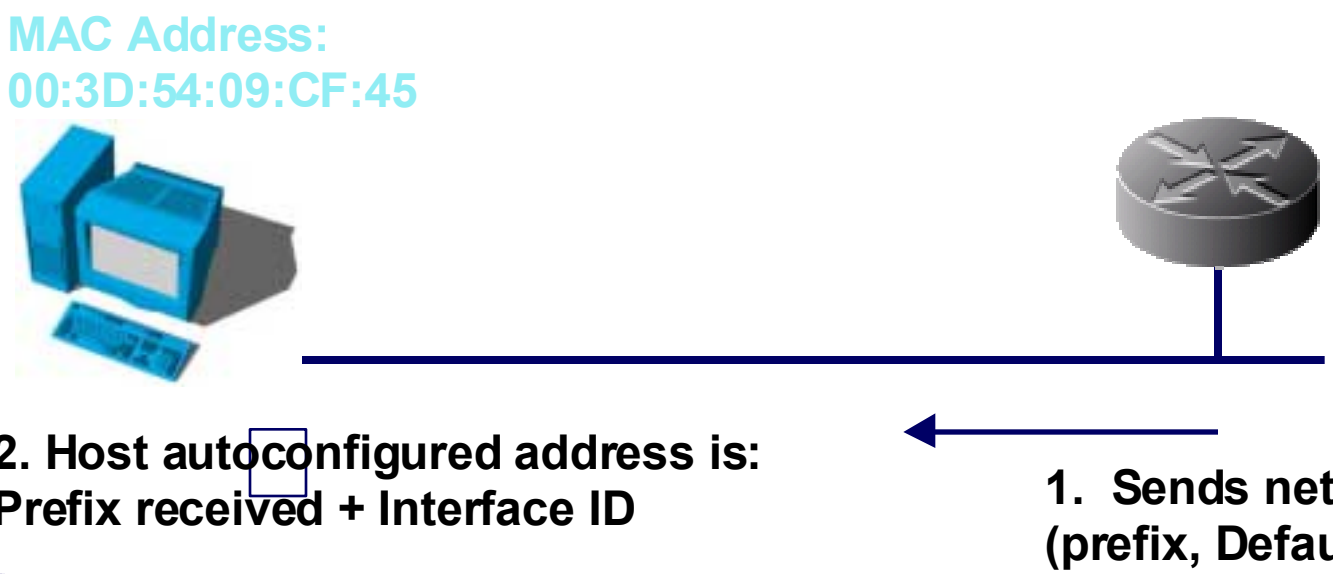
Router Advertisment Message

- ICMPv6 Type = 134
- Src = Router
- Dst = all nodes multicast address (FF02::1) or Unicast address of host that sent Solicitation



Stateless Autoconfiguration

- No need for DHCP
- Simplified Renumbering
- Router notifies hosts of supported prefix(es) using the "options" area of a Neighbor Advertisement packet.



Redirects: Better Router or On Link

BETTER ROUTER:

- If the selected router knows (via a routing protocol, for example) that a different router has a lower cost for the target address OR If the selected router knows (via a routing protocol, for example) that a different router has a lower cost for the target address, the router will send a redirect with the better router's IP address (and hopefully link-layer address) in the Target Address field

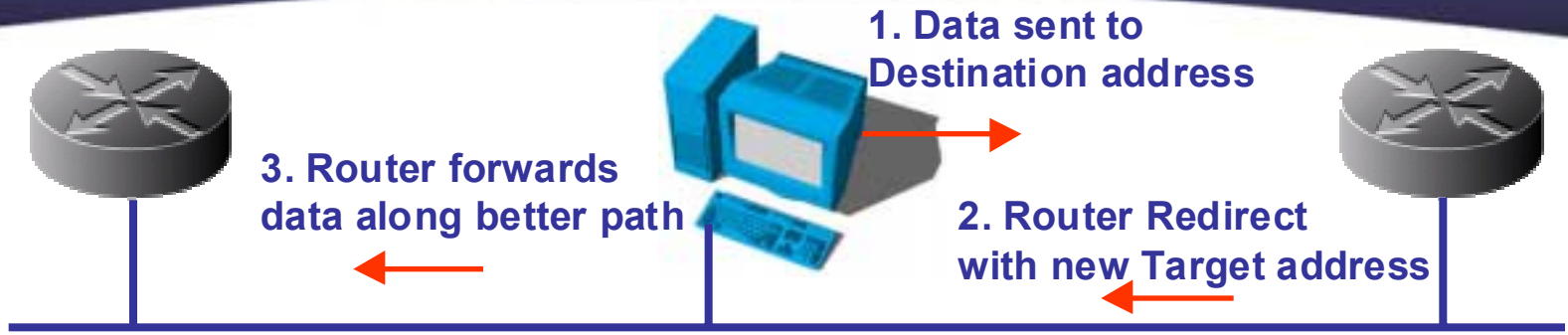
ON LINK:

- Sometimes the router knows that the target is on-link. To notify the sender, it will generate a redirect with the ICMP Target address and Destination address the same.
- In either case, the redirect is advisory. The router will forward the frame anyway

Redirects

- Target Address An IP address that is a better first hop to use for the ICMP Destination Address. When the target is the actual endpoint of communication, i.e., the destination is a neighbor, the Target Address field MUST contain the same value as the ICMP Destination Address field. Otherwise the target is a better first-hop router and the Target Address MUST be the router's link-local address so that hosts can uniquely identify routers.
- Redirects occur if
  - There may be a better router than the one chosen.
    - OR
  - If more than one router is on-link

Redirect Message



Type	Code	Checksum
Reserved		
Target Address		
Destination Address		
Options……		

- Routers send Redirect packets to inform a host of a better first-hop node

The Basic Algorithm

1. If there is no entry for that neighbor in the cache, the host should send a neighbor solicitation message. The neighbor is then added to a new cache line whose status is set to incomplete

2. If there is already an entry for that neighbor, but its status is incomplete, the host should wait for the completion of the procedure to learn the media address and send the packet

IPv6 Extension Headers

Fast Facts On Extension Headers

- Made it possible to streamline the IPv6 Header to a fixed 40 bytes.
- Extensions Include
  - Hop-By-Hop Options Header
  - Destination Options Header
  - Routing Header
  - Fragment Header
  - Authentication Header
  - Encapsulation Security Payload (ESP) Header

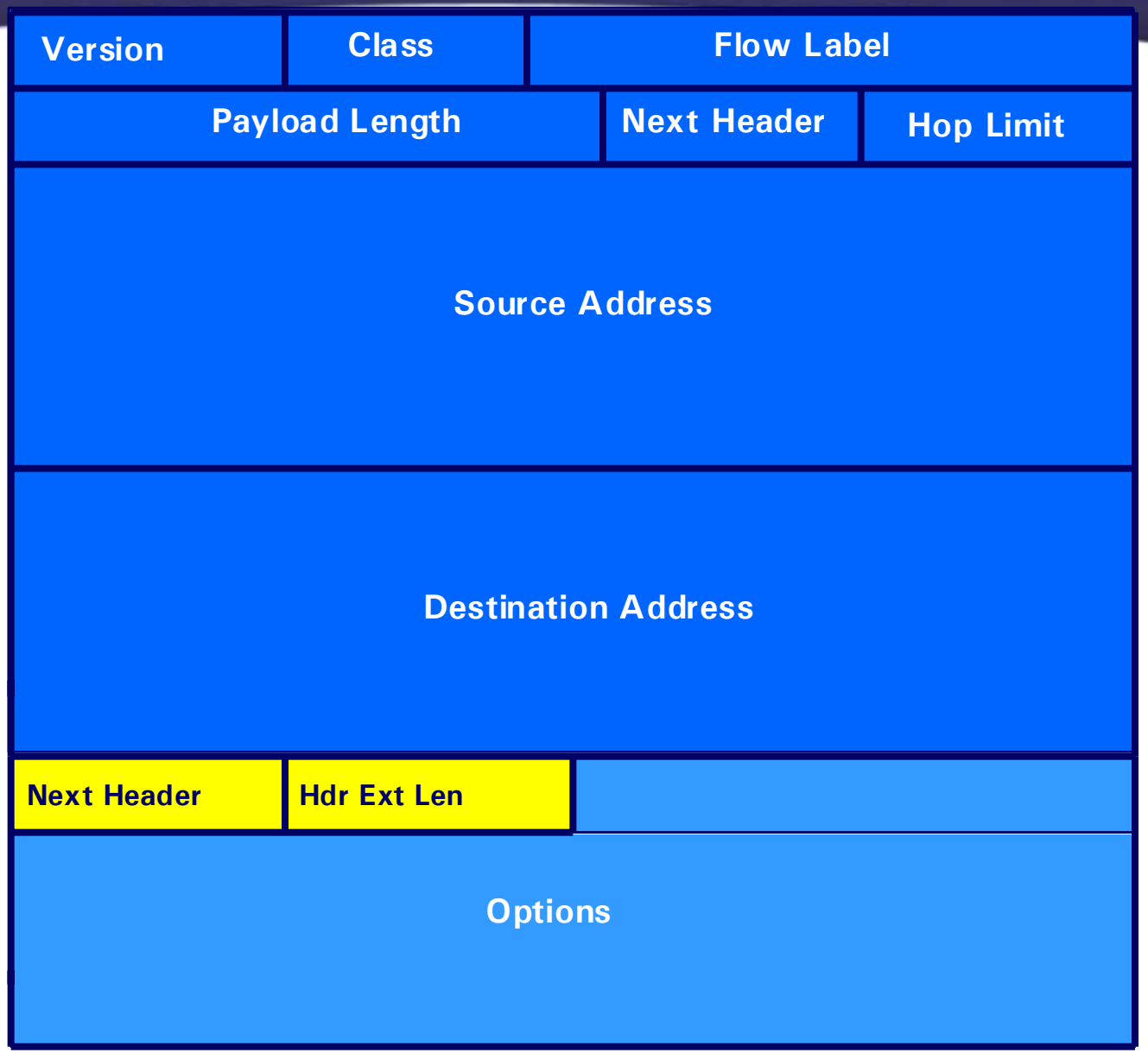
Extension Headers Explained

- Hop-By-Hop Options Header
  - Must be read by every node from source to destination.
- Destination Options Header
  - Processed by the destination node only. This header is very flexible and will be used as a basis to define future headers.
- Routing Header
  - Header contains a list of nodes which must be traversed enroute to destination. Each node in list processes the header & changes destination address to next router in the list.

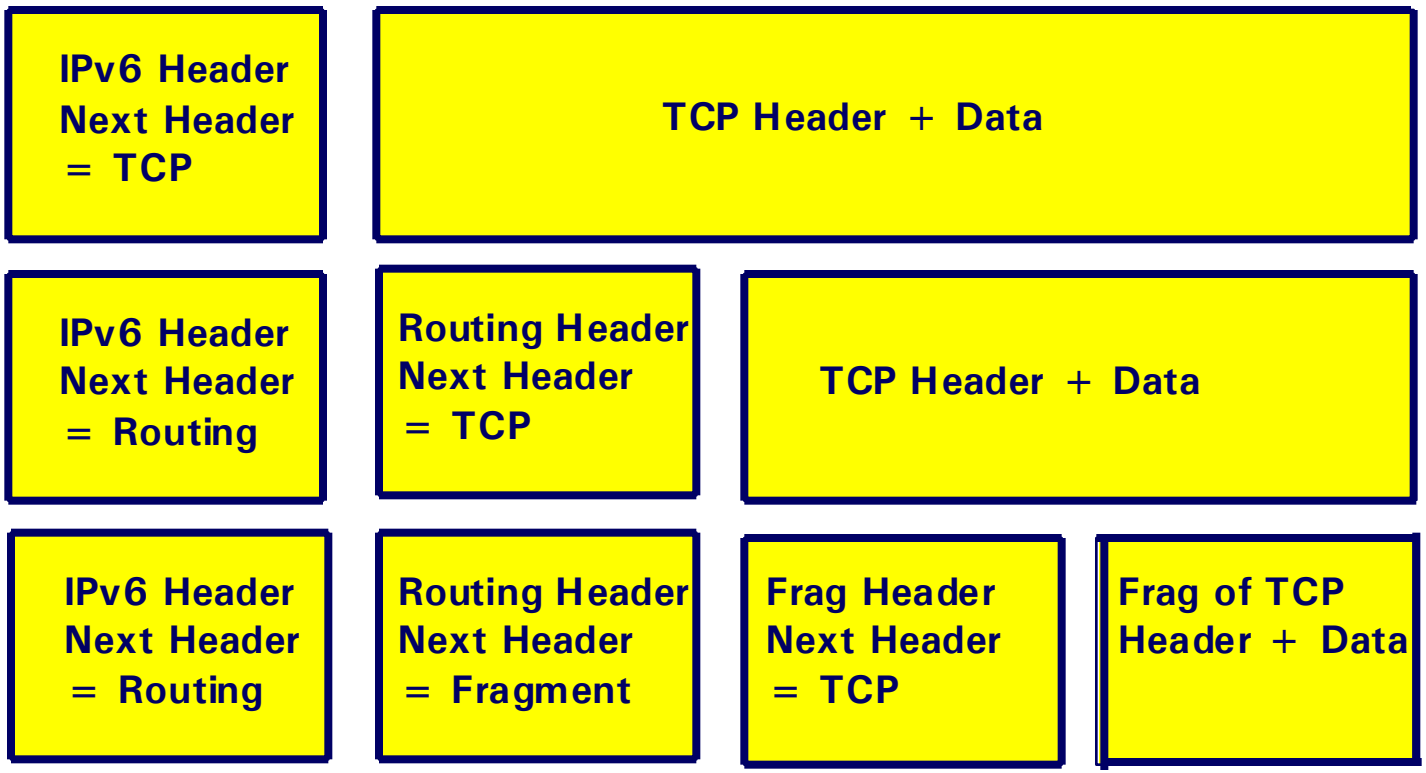
Extension Headers Explained

- Fragment Header
  - Only added by the source node. Like IPv4, contains fragment offset, more fragments, and ID fields.
    - **Fragment Header Note:** IPv6 Specs state that networks should have a minimum MTU of 1280 byte. Sources should be configured to transmit packets at that size or smaller. In addition, those nodes should also implement Path MTU discovery to detect links that can handle larger packets, and take advantage of the larger packet sizes.
- Authentication Header
  - Carries content verification data, links the source with the contents of the datagram, uses keys to ensure message integrity, and uses a sequence number field to protect against replay attacks.
- Encapsulation Security Payload (ESP) Header
  - Provides many of the same services as the authentication header, but adds encryption functionality.

Extension Headers

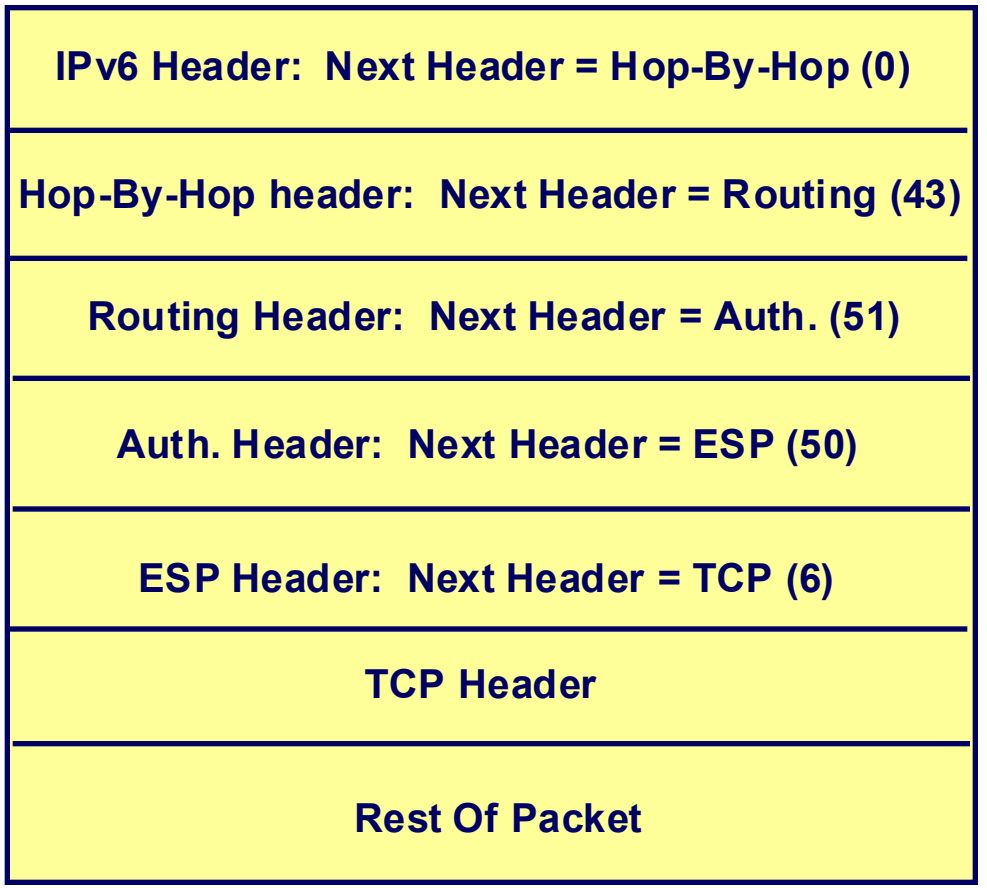


Extension Headers

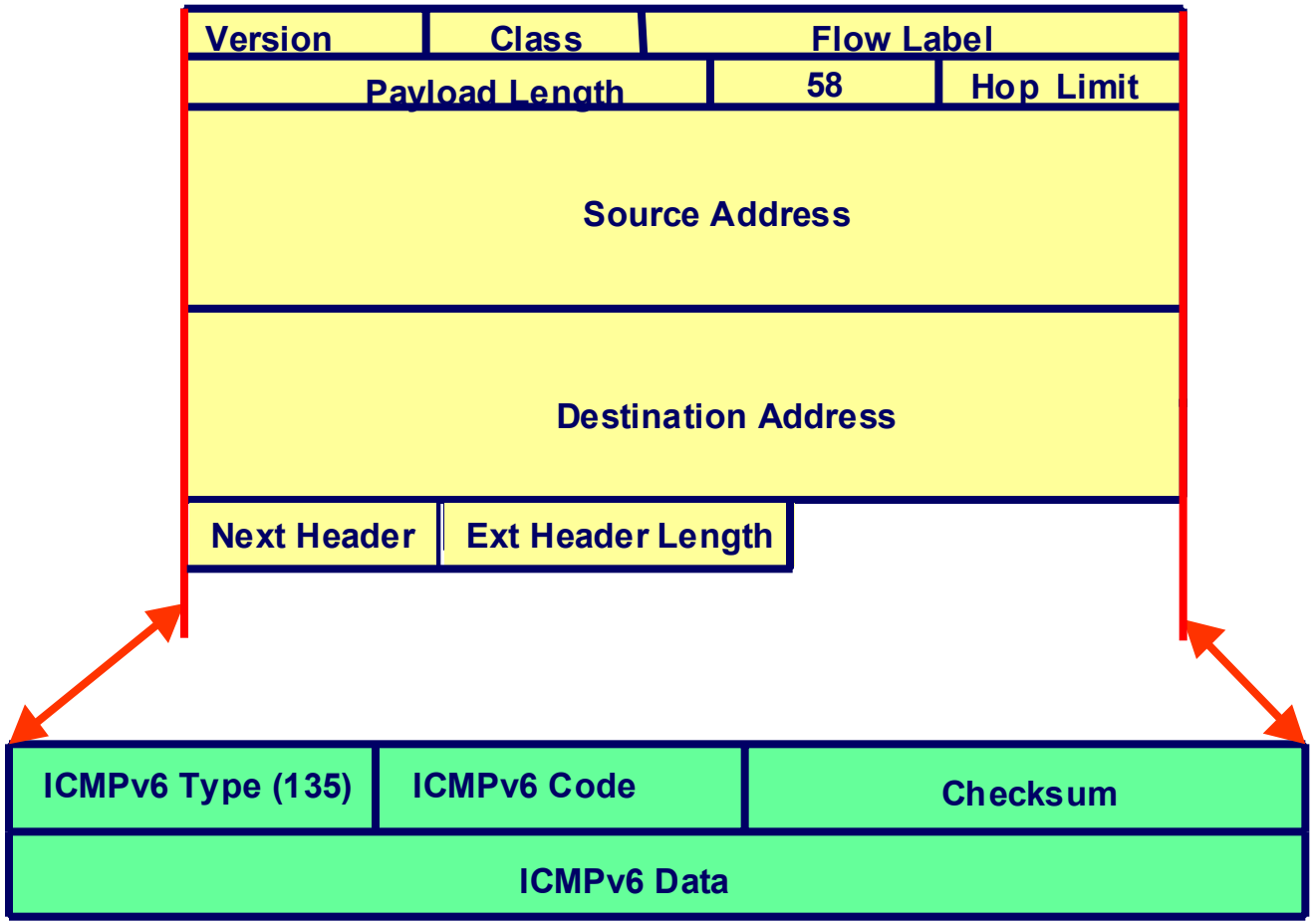


Extension Headers

- There is a Next Header field in each extension header, which ultimately indicates the L4 protocol. All numbers are from RFC 1700 - Assigned Numbers.
  - Headers can be “stacked” or “daisy chained”. The “next Header” field indicated either which extension header will be next, or indicated the upper layer protocol.



Neighbor Discovery (RFC 2461)



Hop-By-Hop Extension

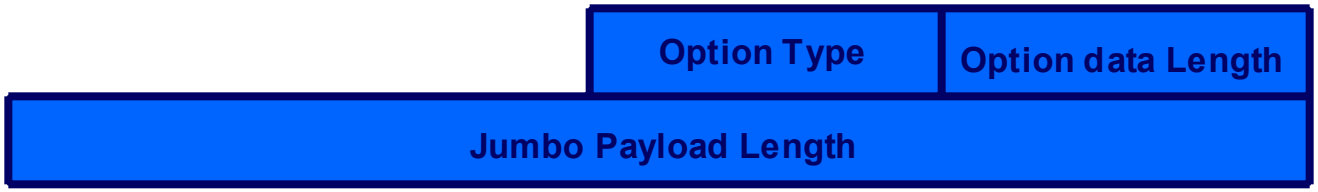
Jumbograms

RFC- 2675

- Jumbograms support packets up to 4GB in length.
- More efficient transfers with fewer interrupts tothe communicating hosts.
- IPv6 provides only 16 bits for the length field in the header
- Hop-by-Hop Options header needs to be set to 0.

Jumbograms

RFC - 2675



- Option Type = C2 (8 bit)
- Option Data Length = 4 (8 bit)
- Jumbo Payload Length = 32 bit

Length of IPv6 Packet
Not including Packet Header
Includes Extensions
Must be greater then 65,535

Router Alert Option RFC 2711

- Informs the router that the contents of this datagram is of interest to the router and to handle any control data accordingly.
- Protocols such as RSVP are required to use this option.
- Very efficient, packets not requiring further inspection are passed straight through.

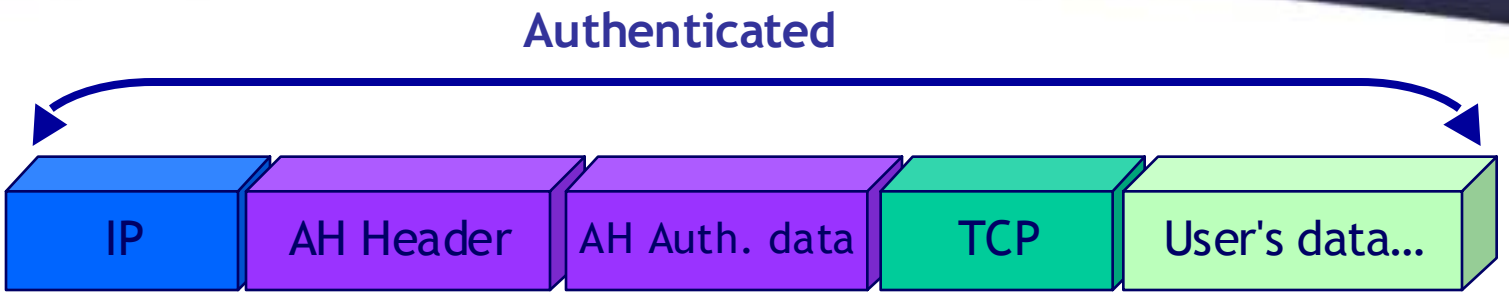
Router Alert Option Syntax

- If the Value field is set to 1 this indicates the Datagram contains an RSVP message.

Security
AH Header and ESP Header

- IPSec is defined as part of IPv6
- Essentially identical in both IPv4 and IPv6
- IPv6 can be deployed end to end, whereas, Typically IPSec in IPv4 is deployed between border routers
- Uses Authentication Extension Header and Encapsulating Security Payload

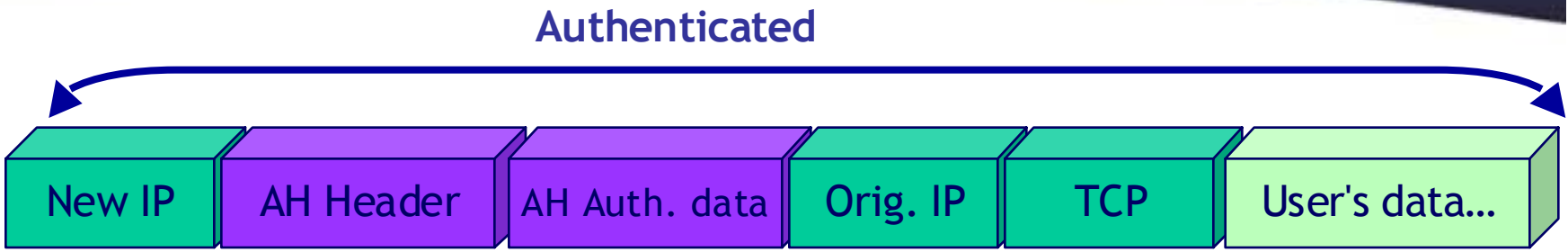
AH in Transport Mode



- Only IP's immutable fields are authenticated
  - Version #
  - Payload length
  - Next header (51 in this case)
  - Source and Destination IP addresses

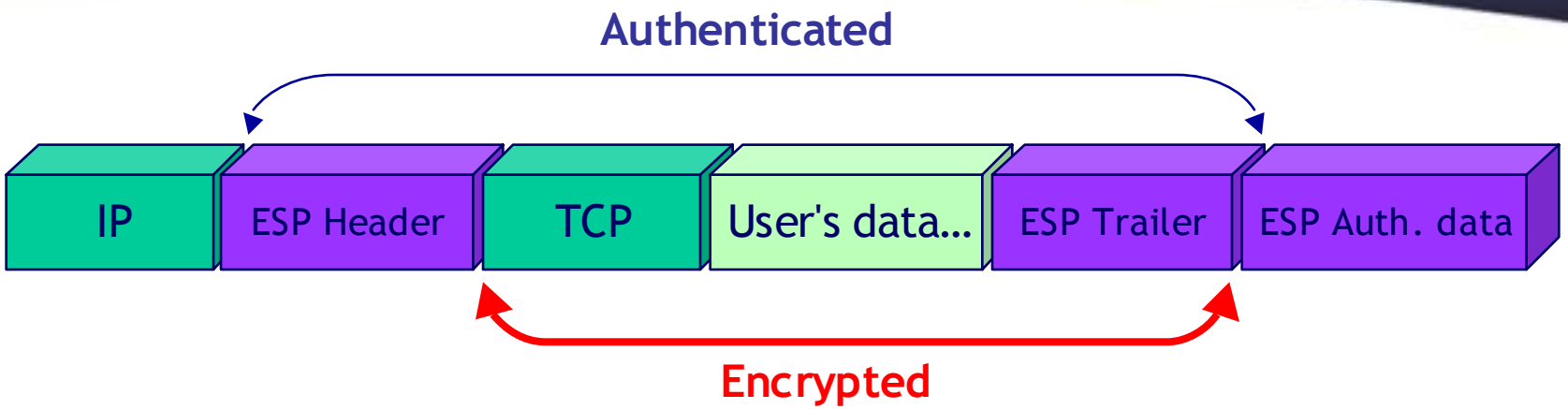
- AH's Authentication data filed is set to zero during the authorization process

AH in Tunnel Mode



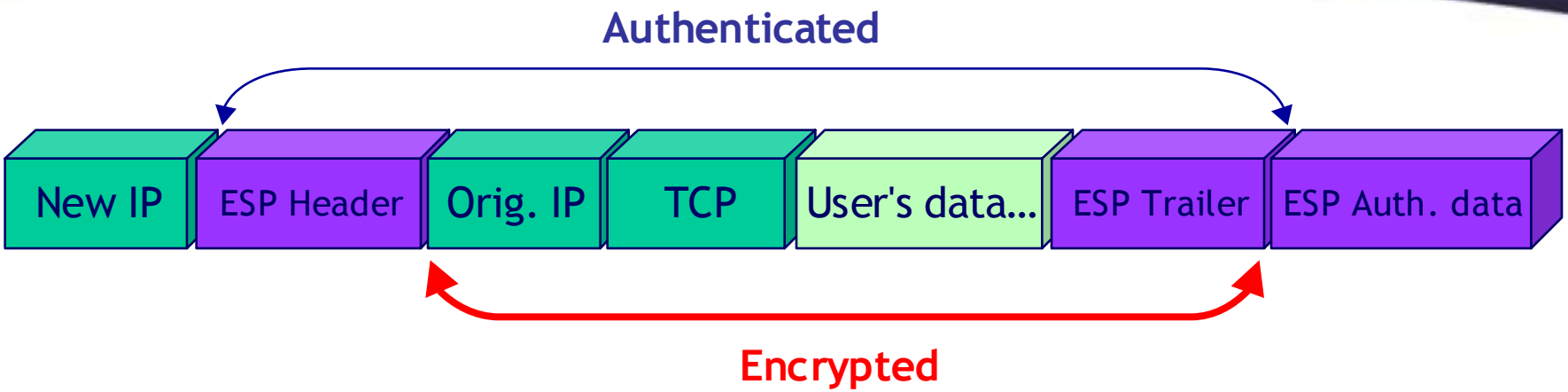
- Only IP's immutable fields in the New IP header are authenticated.
- AH's Authentication data filed is set to zero during the authorization process

ESP in Transport Mode

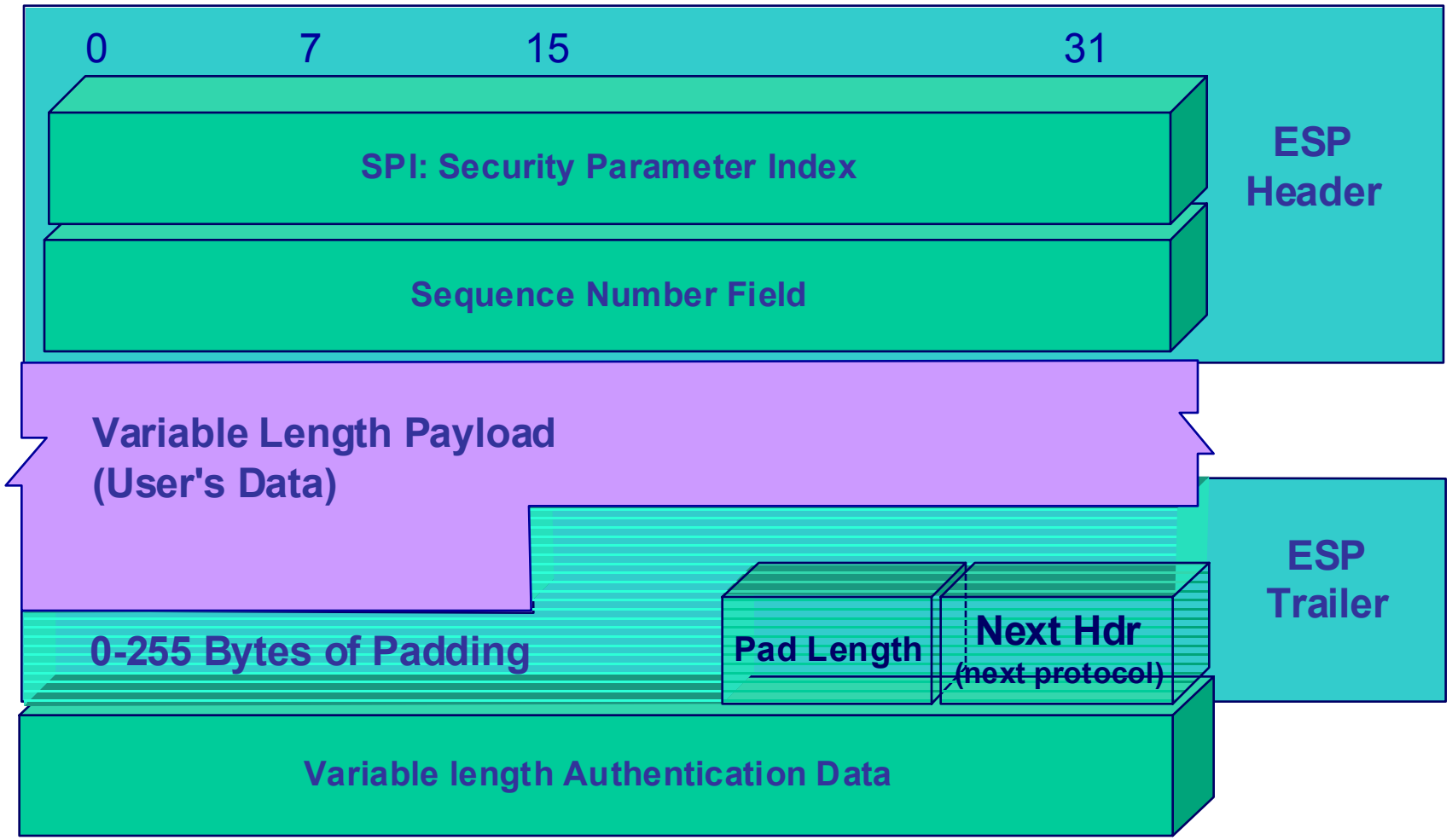


- The ESP Header contains the SPI and Sequence numbers which are not encrypted
- ESP trailer contains the padding, pad length and next header fields

ESP in Tunnel Mode

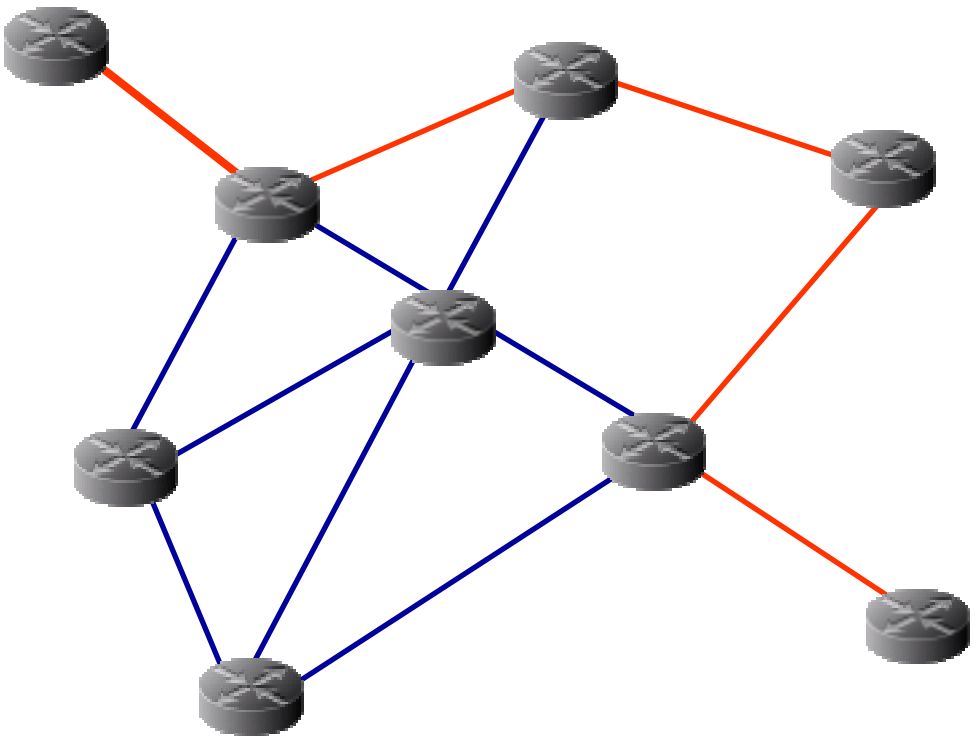


ESP Header/Trailer Format

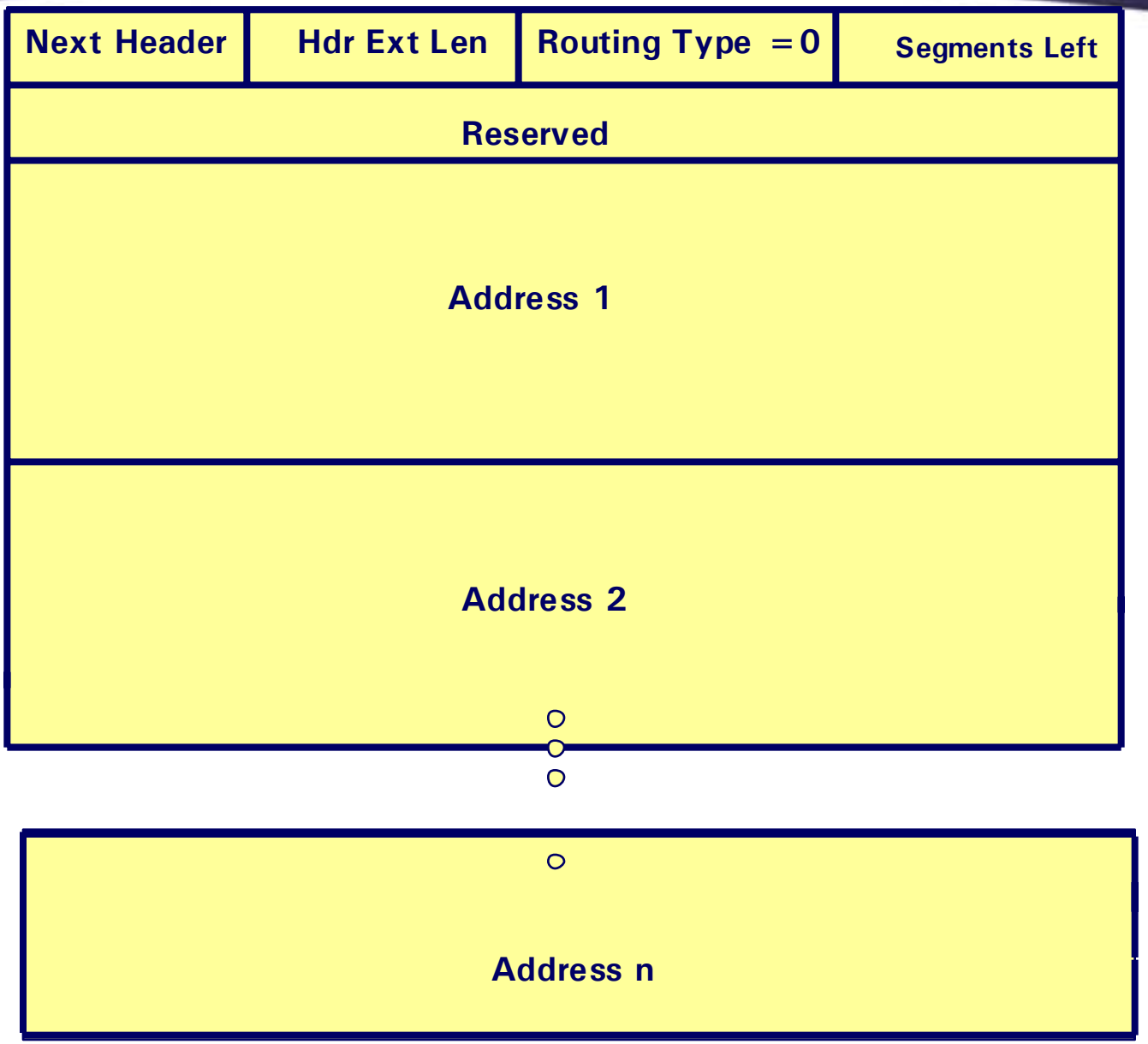


Routing Extension

- Defines the exact path to take
- Similar to traffic engineering in MPLS



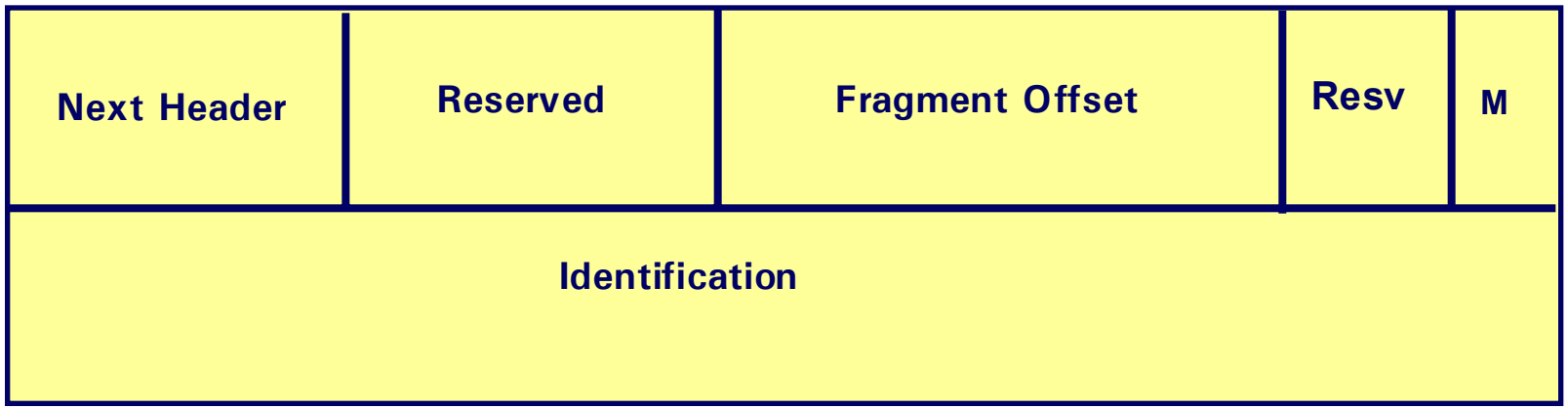
Routing Extension



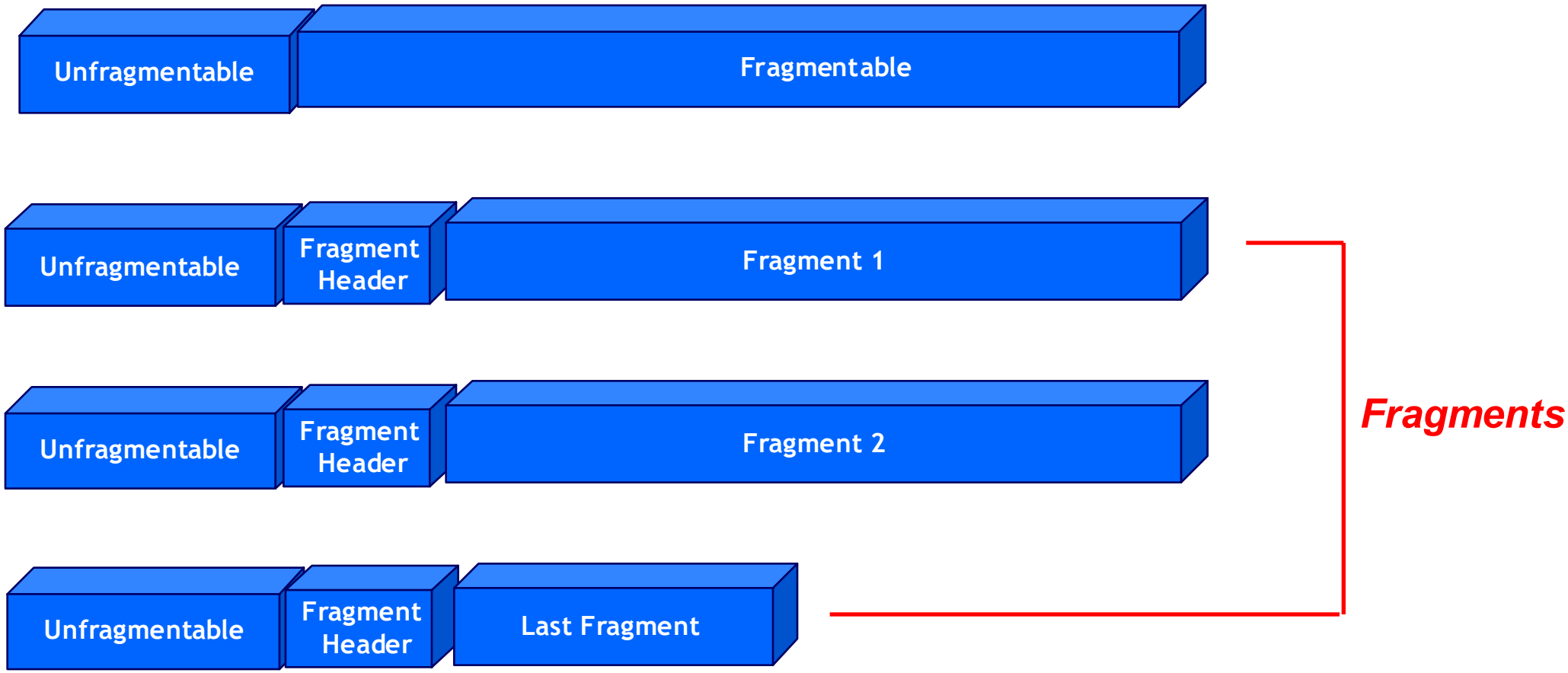
Fragmentation Header

- Fragmentation is handled by the packet source
- Fragmentation is not handled by the router
- IPv4 mMTU is 64 Bytes, IPv6 mMTU is 1280 Bytes
- IPv4 MTU is 1580 BytesIPv6 MTU is 65,535 Bytes
- Path MTU discovery in IPv6 allows the host to dynamically discover and adjust to differences.

Fragmentation Header



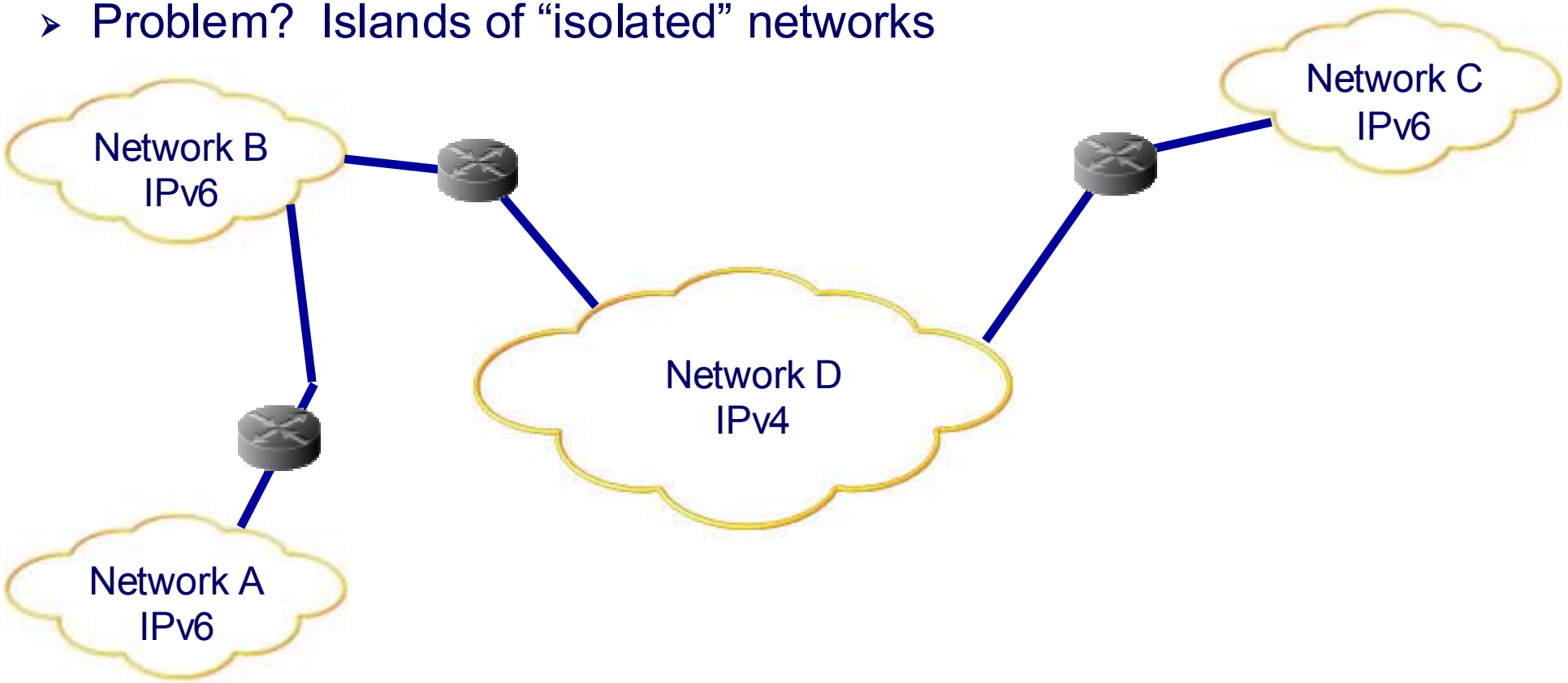
Fragmentation Header



Transition Issues

IPv4/IPv6 Transition

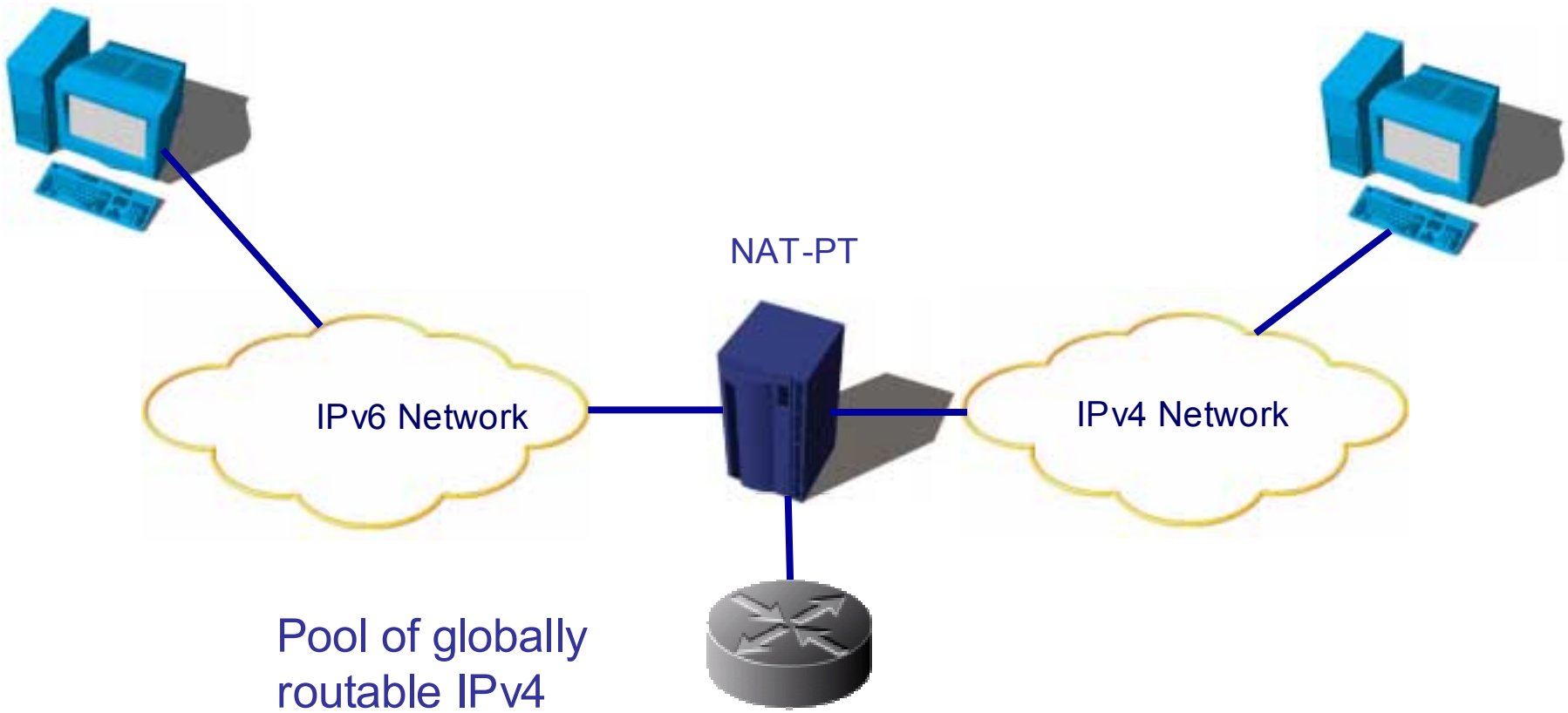
- How to inter-operate IPv4/IPv6
- Without Tunneling
  - Problem? Islands of "isolated" networks



Network Address Translator/ Protocol Translator

- NAT-PT Provides two operations:
  1) Protocol translation, due to the difference between the IPv4 & IPv6 header. As some data cannot be translated, this is a best effort.
  2) Address translation, due to the difference in address lengths.
- IPv4 address translation to IPv6 is relatively easy. The NAT-PT adds an IPv6 prefix to the upper 96 bits of the IPv6 address. The IPv4 address is added as the lower 32 bits of the address.
- IPv6 address translation to IPv4 is achieved using the NAT-PT IPv4 pool. Each IPv4 pool address can be used as an alias for an IPv6 address. Replies that arrive at NAT-PT from the IPv4 network are translated back to the associated IPv6 address.

NAT-PT



Pool of globally routable IPv4 addresses

Bump-In-the-Stack

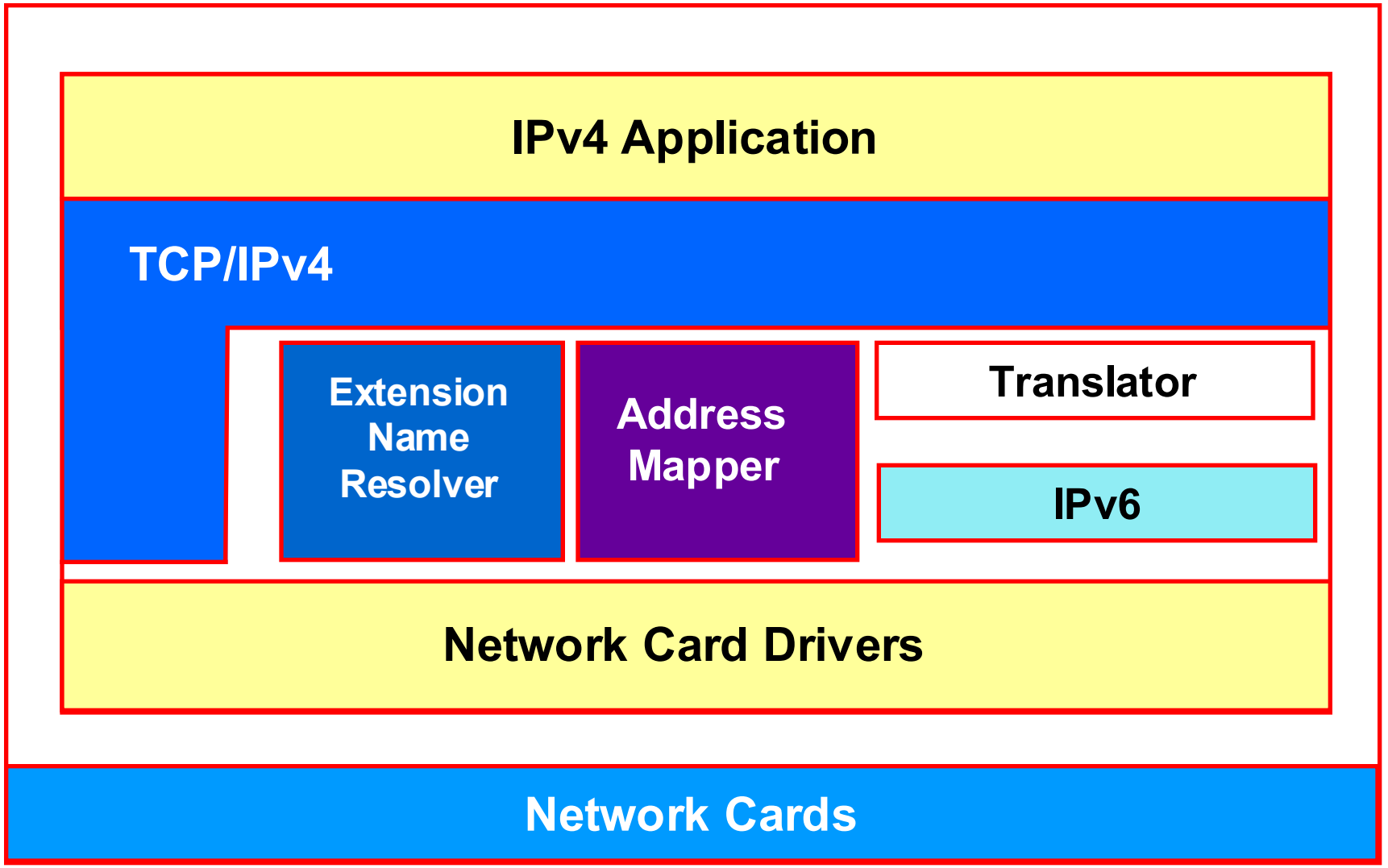
Bump-In-the-Stack (BIS)

RFC 2767

Bump-In-the-Stack

- Allows an IPv4 only application to communicate with an IPv6 only host
- Three modules are added to the IPv4 stack
  - Translator, Converts IPv4 application Packets to IPv6 Packets
  - Extension Name Resolver, If only an IPv6 address is available a request is made to the Address mapper to assign a corresponding IPv4 address.
  - Address Mapper, maintains a spool of IPv4 addresses. Also, it maintains a table which consists of pairs of an IPv4 address and an IPv6 address.

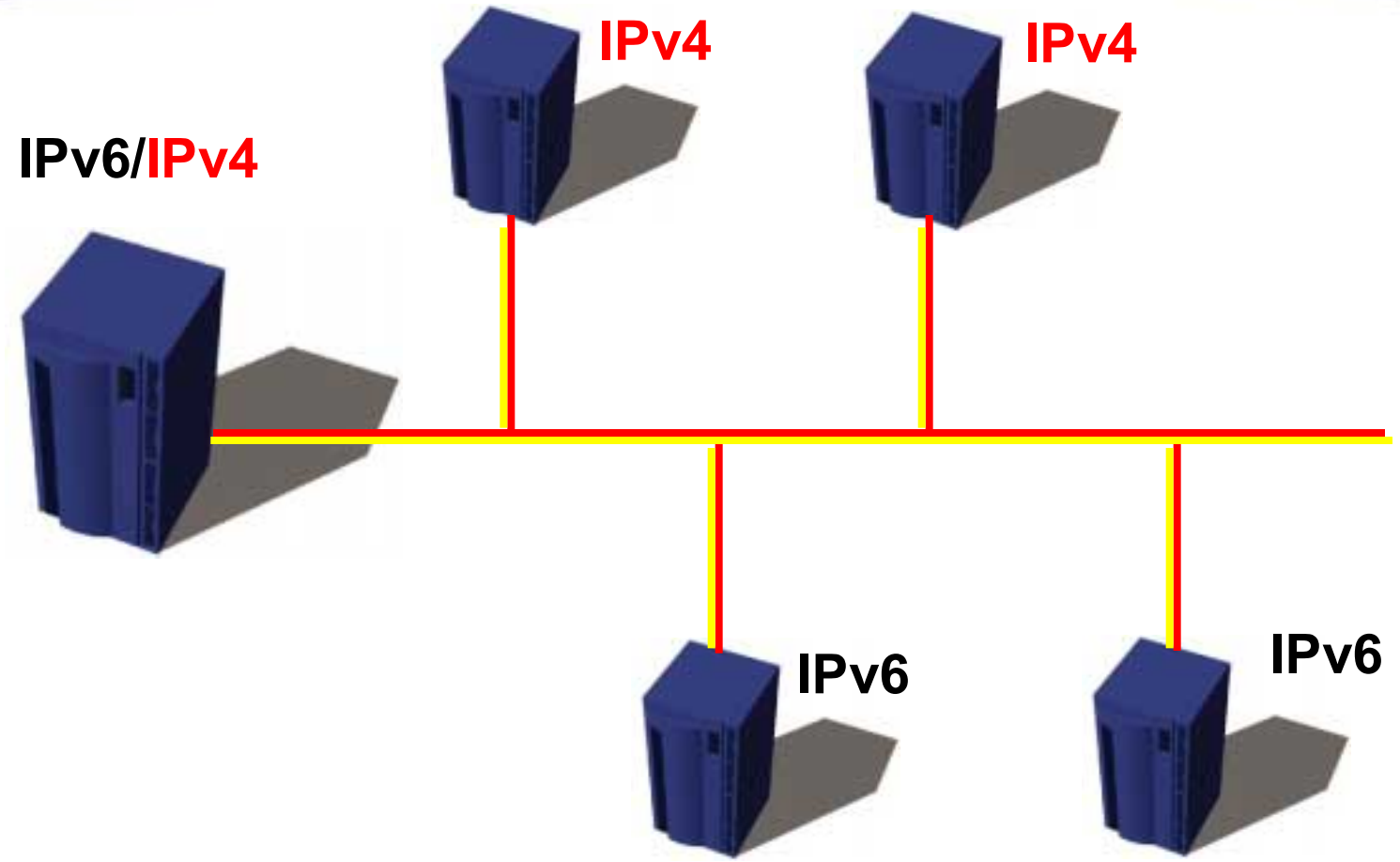
Bump-In-the-Stack



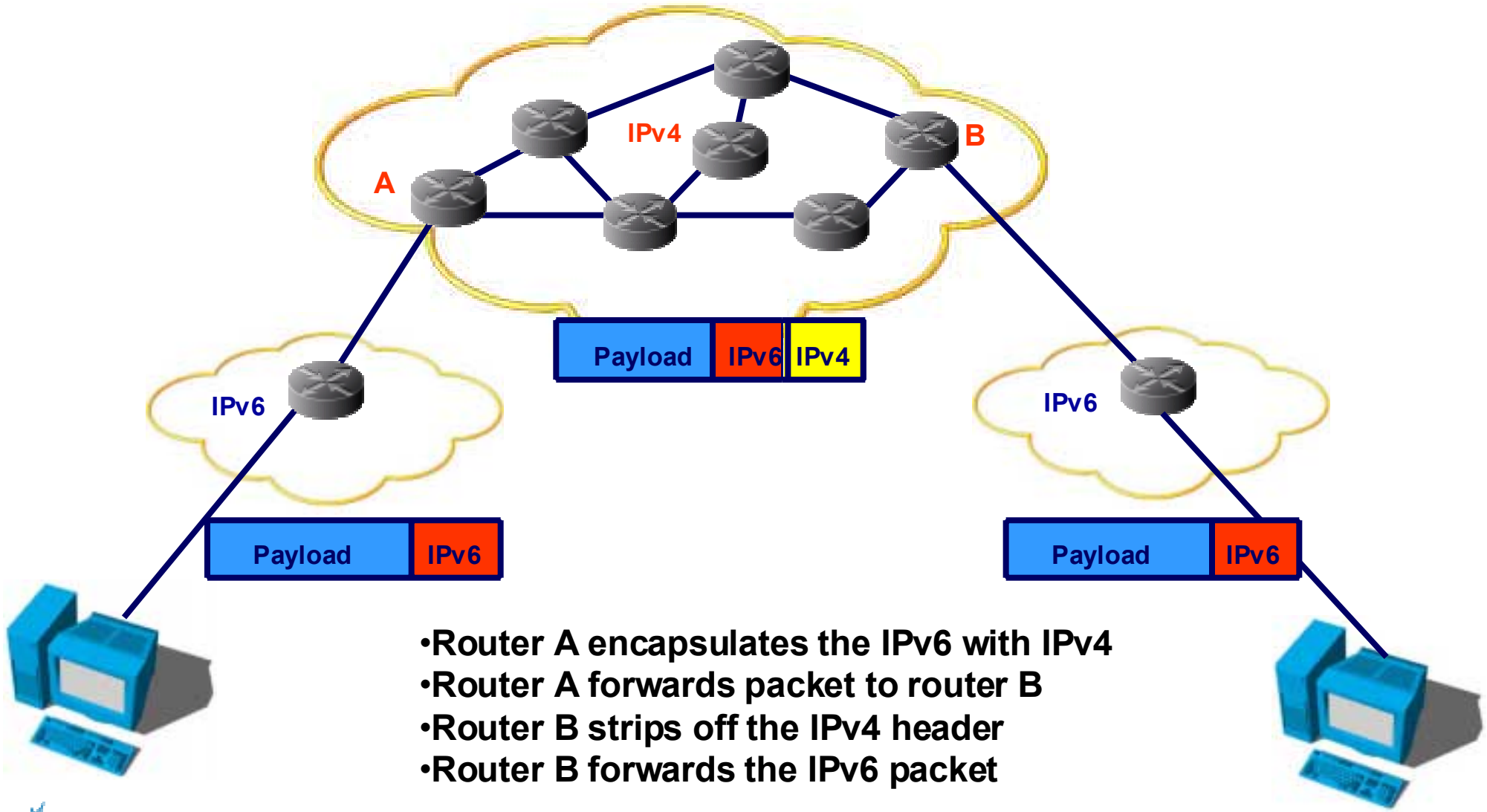
Dual Stack RFC 2893

Dual Stack

Dual Stack RFC 2893



IPv6 Tunneling: Basic Theory



- Router A encapsulates the IPv6 with IPv4
- Router A forwards packet to router B
- Router B strips off the IPv4 header
- Router B forwards the IPv6 packet

IPv6 Tunneling Format

The packet is created with a standard IPv4 header

IPv4

Version	IHL	TOS	Total Length
Identification		Flags	Fragmentation Offset
Time To Live	Protocol 41		Header Checksum
Source Address			
Destination Address			
Options			Padding
IPv6 Header & Payload			

IPv6

6over4 RFC 2893

Transition Method "6over4"

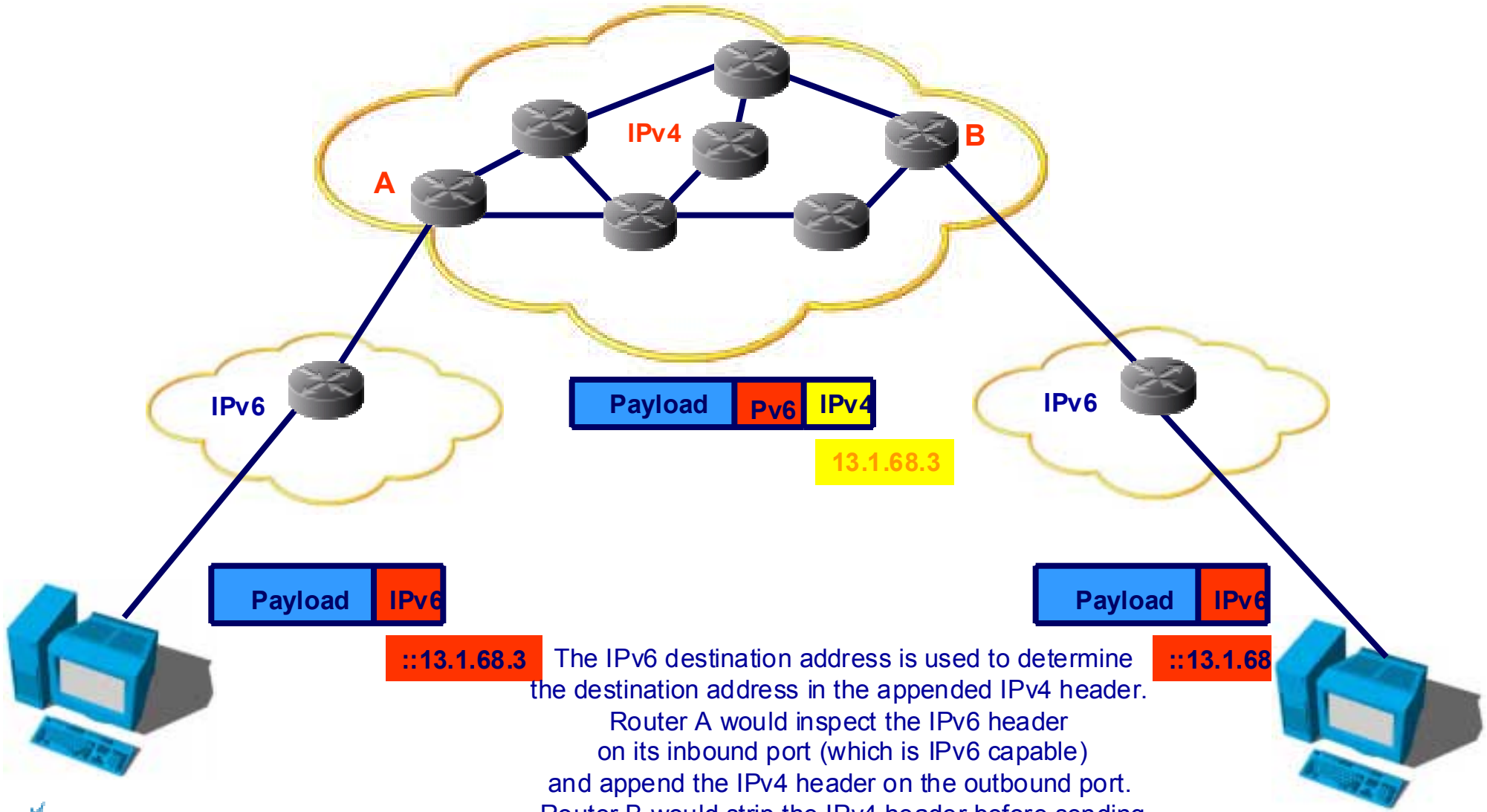
RFC 2893

- This RFC specifies two methods for creating tunnels
- Configured and Automatic
- Known as “6over4”

Tunneling Method 1: Automatic

- IPv4 Compatible IPv6 v6[v4]
- Can be converted between IPv4 and IPv6
- 0:0:0:0:0:0:13.1.68.3 or ::13.1.68.3
  - In this case, the IPv6 address of the tunneling node (in many case, the router) is an IPv4 compatible address - an address with 96 bits of leading zeroes followed by the 32-bit IPv4 address.

Tunneling Method 1: Automatic



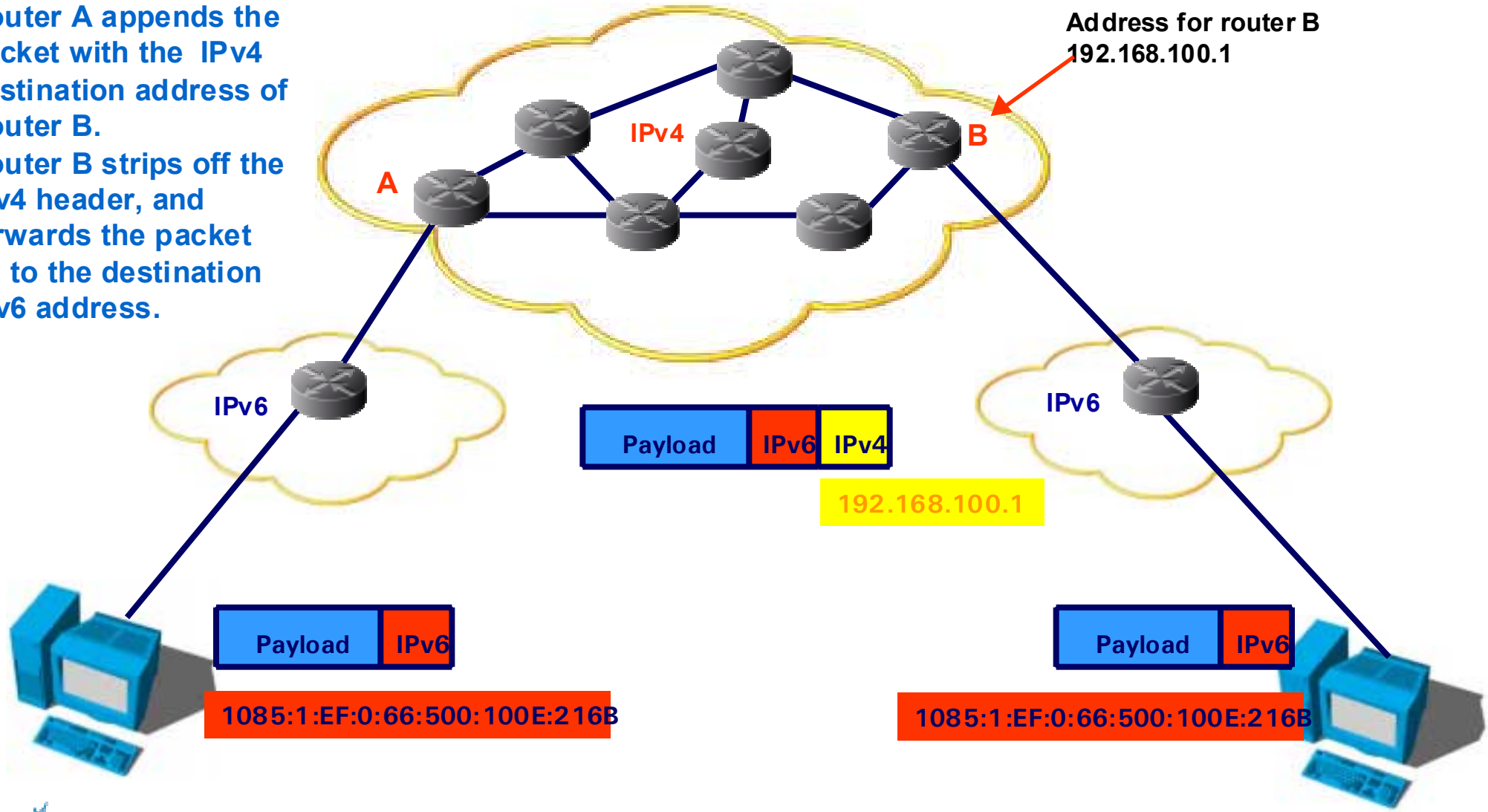
The IPv6 destination address is used to determine the destination address in the appended IPv4 header. Router A would inspect the IPv6 header on its inbound port (which is IPv6 capable) and append the IPv4 header on the outbound port. Router B would strip the IPv4 header before sending the packet on to the destination IPv6 network.

Tunneling Method 2: Configured

- Used when a native IPv6 address is used.
- The router will be configured with a destination IPv6 gateway that can be reached with an IPv4 address.
- In this case, the tunnel end points must acquire the destination IPv4 addresses through some other mechanism (like DHCPv4), instead of creating a “compatible” address.

Tunneling Method 2: Configured

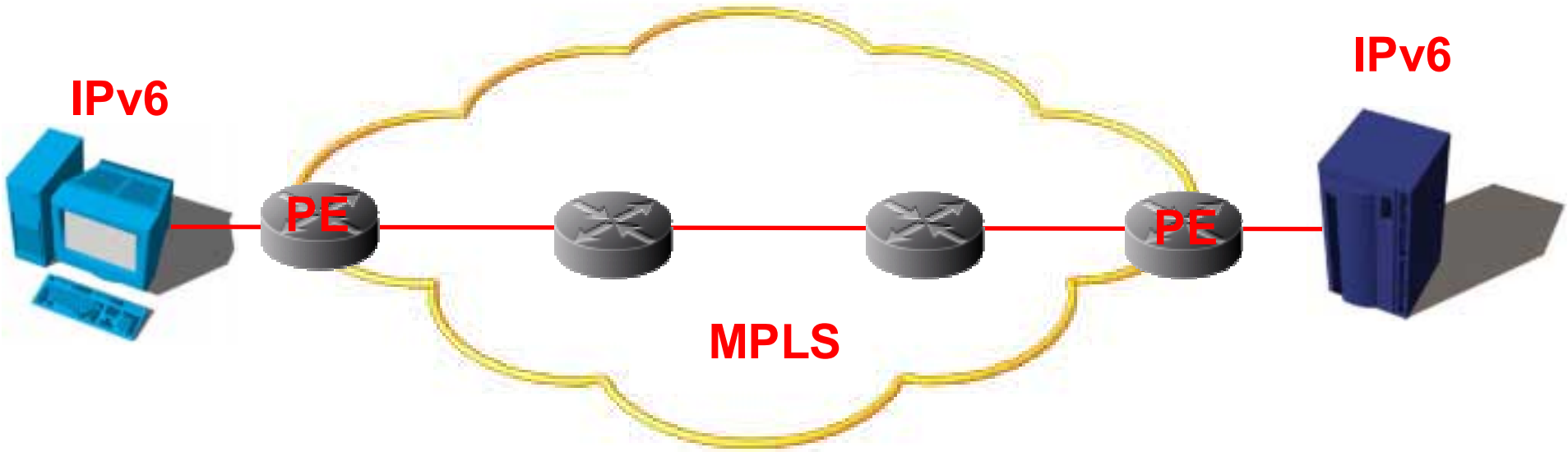
Router A appends the packet with the IPv4 destination address of Router B.
Router B strips off the IPv4 header, and forwards the packet on to the destination IPv6 address.



6PE

Tunneling IPv6 over MPLS

Tunneling IPv6 over MPLS



- The Provider Edge Router (PE) needs to be dual stack
- The existing MPLS network uses IPv4 to distribute labels
- No change required to existing MPLS network

RFC 3056

Transition Method "6to4"

6to4

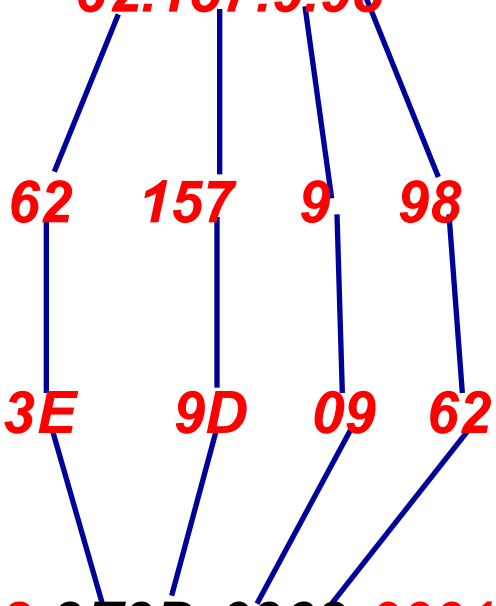
- Applicability: interconnection of isolated IPv6 domains over an IPv4 network
- Automatic establishment of the tunnel
  - No explicit tunnels
  - By embedding the IPv4 destination address in the IPv6 address
  - Under the 2002::/ 16 reserved prefix. (2002::/ 16 = 6to4)
- Gives a full /48 to a site based on its external IPv4 address
  - IPv4 external address embedded: 2002:< ipv4 ext address>::/ 48
  - Format: 2002:< ipv4add>:< subnet>::/ 64

6to4 Prefix Format

The Prefix could be shown as: *2002:V4ADDR::/48*

6to4 Address Conversion

Your IPv4 Address *62.157.9.98*



Decimal:

Hex:

Your IPv6 Address: *2002:3E9D:0962:0001::1*

6to4 Prefix

80 bit Address Space

Who should support 6to4?

- Egress router:
  - Implements 6to4
  - Must have a reachable external IPv4 address
- Often configured using a loopback interface address
  - Is a dual- stack node
- Individual nodes:
  - Nothing needed for 6to4 support. 2002 is an "ordinary" prefix that may be received from router advertisements
  - Doesn't need to be dual- stack

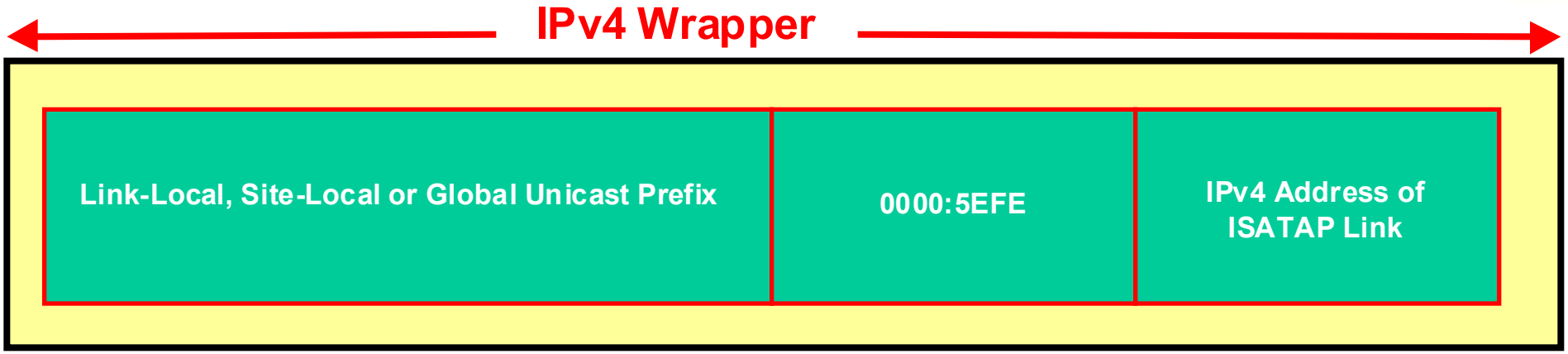
Issues with 6to4:

- Bound to the IPv4 external address:
  - If egress router changes its IPv4 address, then it means that you need to renumber the full IPv6 internal network
  - Only one entry point (no easy way to have multiple network entry points for redundancy)

Intra-Site Automatic Tunnel Addressing Protocol

- ISATAP
- Described in “draft-ietf-ngtrans-isatap-0x.txt”
- Hosts create ISATAP addresses through autoconfiguration mechanisms
- Used in conjunction with “6to4”
- ISATAP can be used for communication between IPv6/IPv4 nodes on an IPv4 network

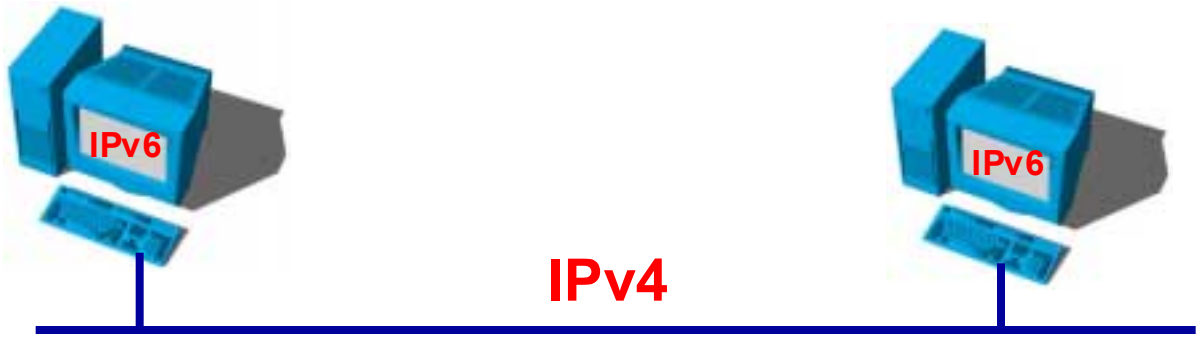
ISATAP Address Format



FP *3* *001*	*TLA* *13* *0x0002*	*IPv4 Address of destination 6to4 router* *32*	*SLA ID* *16*	*Interface ID* **0000:5EFE & IPv4 Link Address** *64*

- The 32 bits preceding the IPv4 address are the IANA OUI (00-00-5E-FE)
- ISATAP can be used in conjunction with “6to4”

ISATAP IPv6 on IPv4

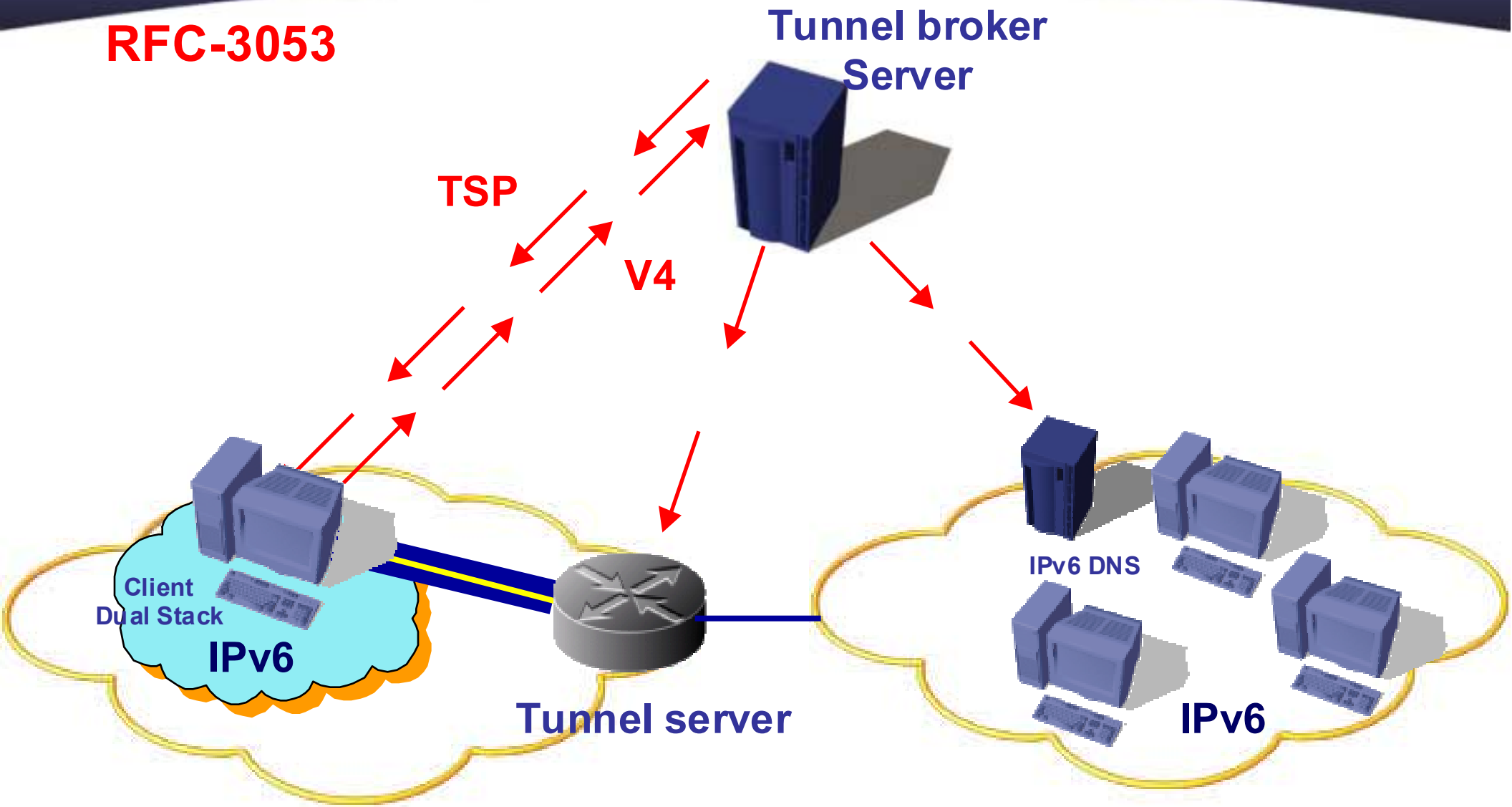


- Designed for users of IPv6 in an IPv4 infrastructure
- ISATAP, IPv6 treats the IPv4 inter-network as a link layer

RFC-3053

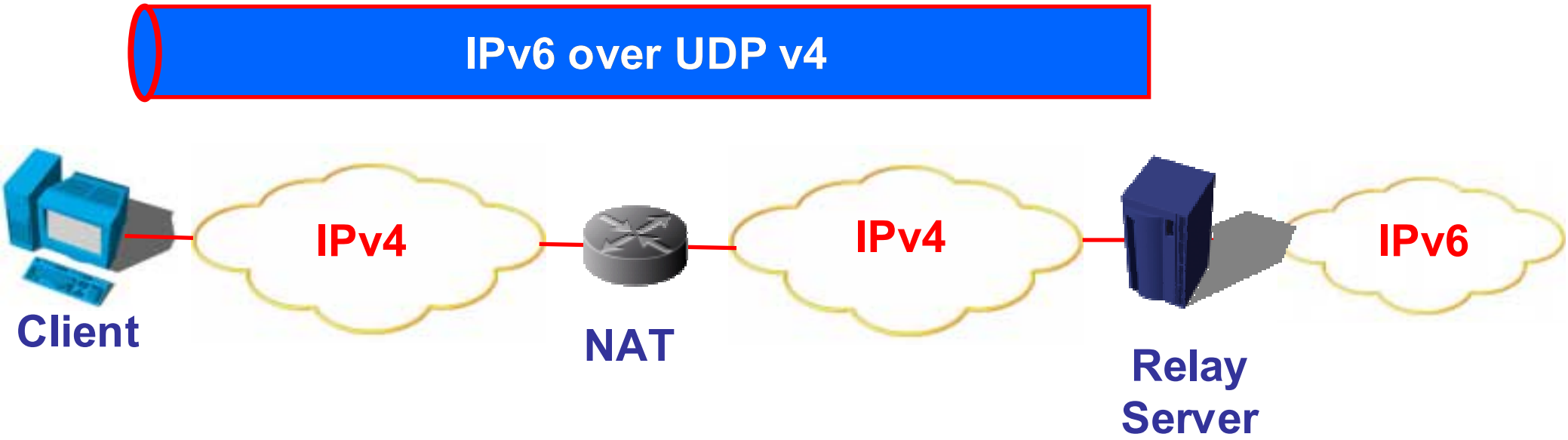
Tunnel Broker

Tunnel Broker



Shipworm (Teredo)

- Provides IPv6 Connectivity across one or more NATs
- Tunnels IPv6 over UDPv4 through the NAT
- Use of new address format



Shipworm Message Format

IPv6 Performance Testing

Spirent IPv6 Test Equipment

SB6000/600

AX-4000

Spirent IPv6 Test Solutions

• IPv6 performance testing	SmartFlow
• IPv6 transition testing	AX-4000/SmartFlow
• IPv6 Conformance testing	AX-4000
• IPv6 Router testing	TRT
• IPv6 QoS	SmartFlow
• Basic IPv6 Testing	SmartWindow
• IPv6 Scripted Testing	Spirent Connect

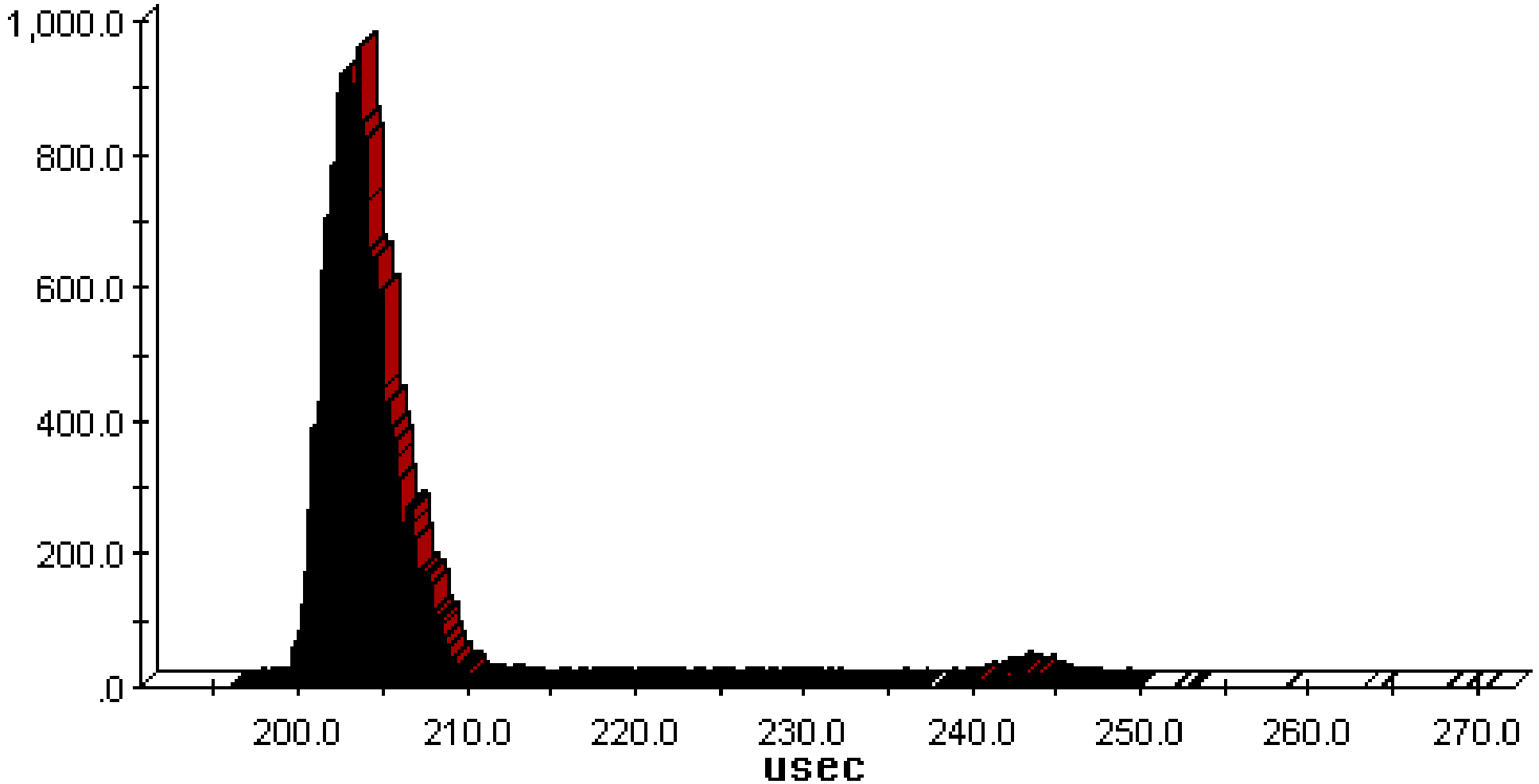
IPv6 Testing

- Basic throughput IPv4 only, single stack.

IPv4 Traffic Generation

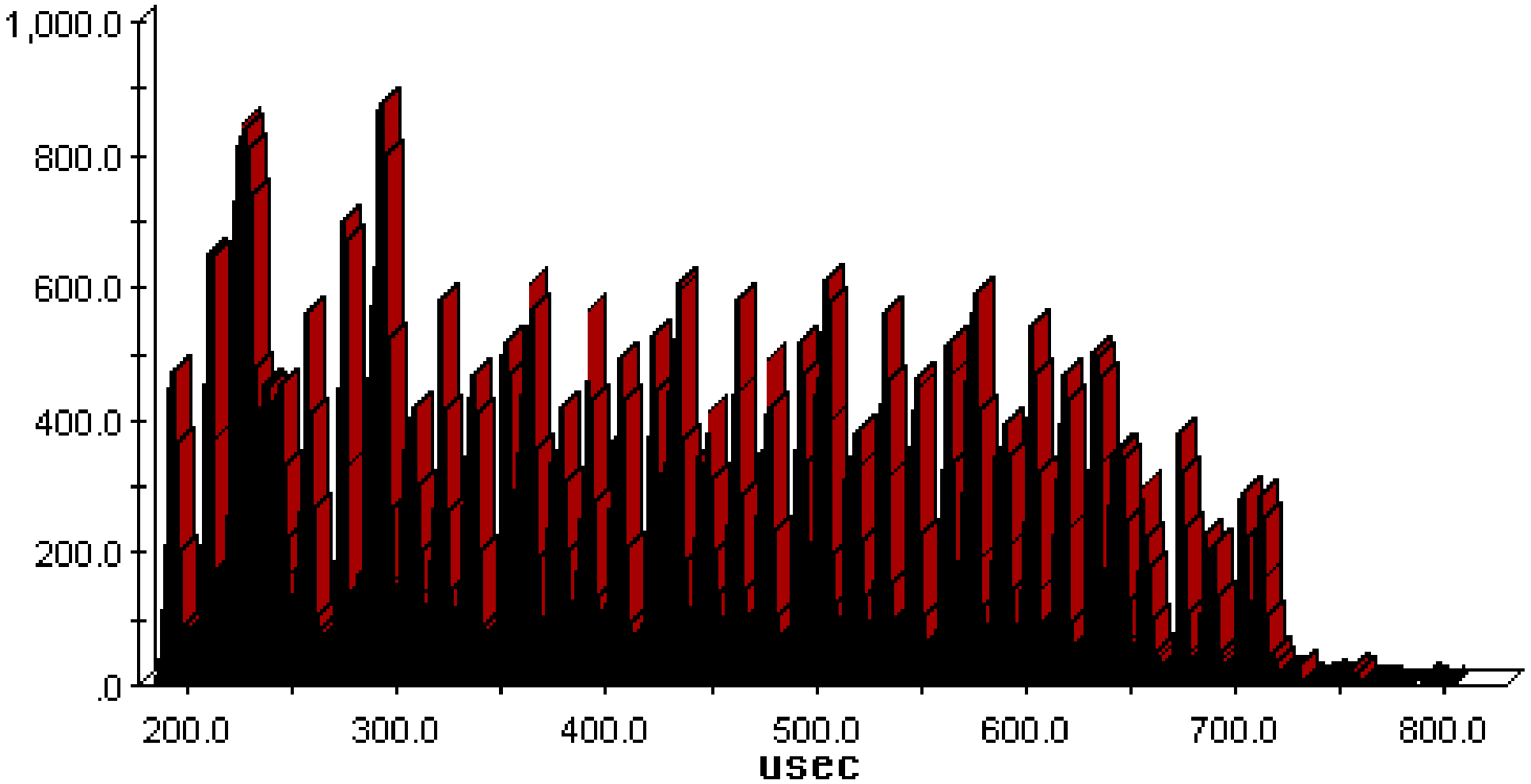
- **Two quick tests at 10% and 100% load (HDX)**
- **Reference for basic router performance**

IPv4 Latency



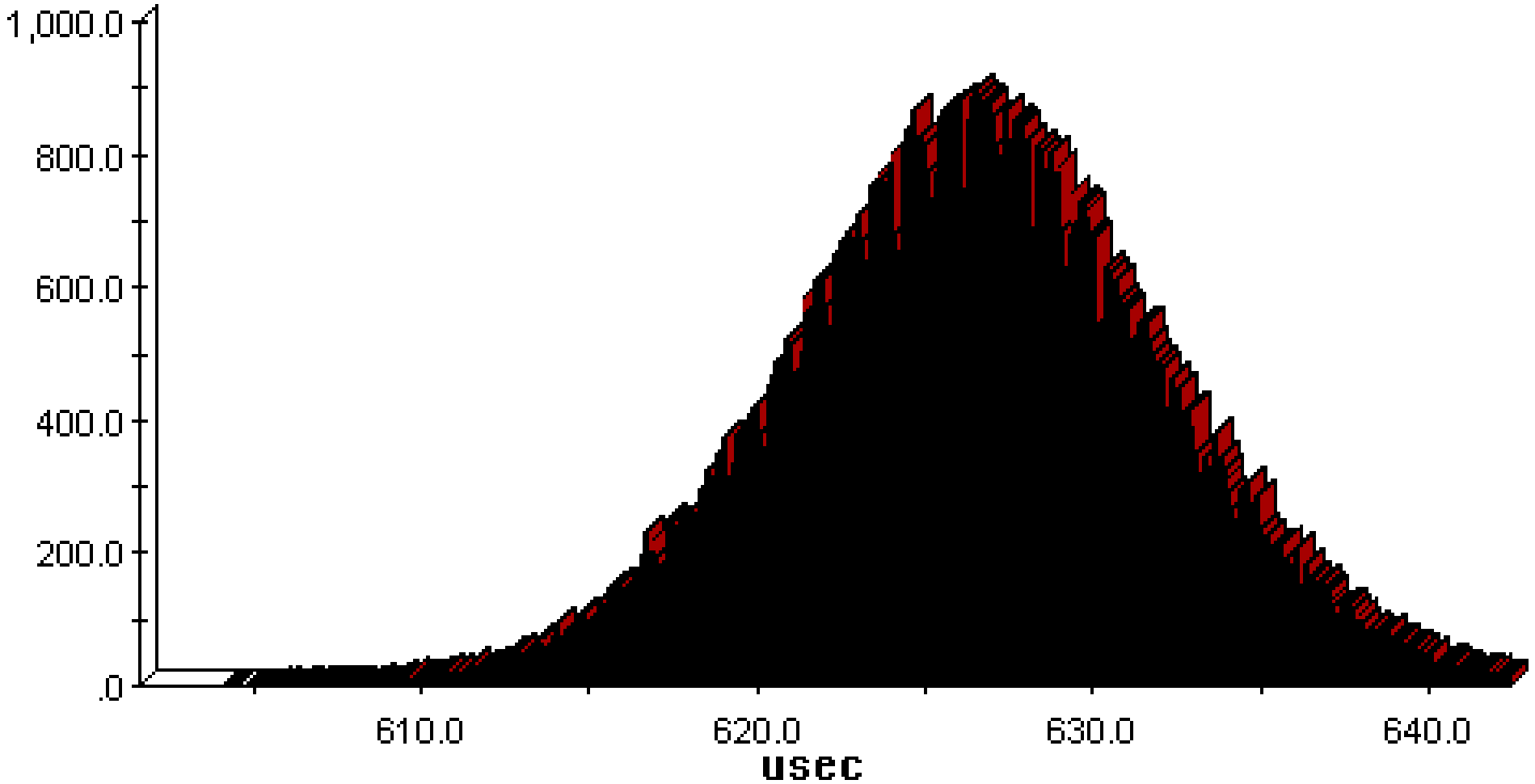
IPv4 10% Load (10Mbps) with 90 byte packets

IPv4 Latency



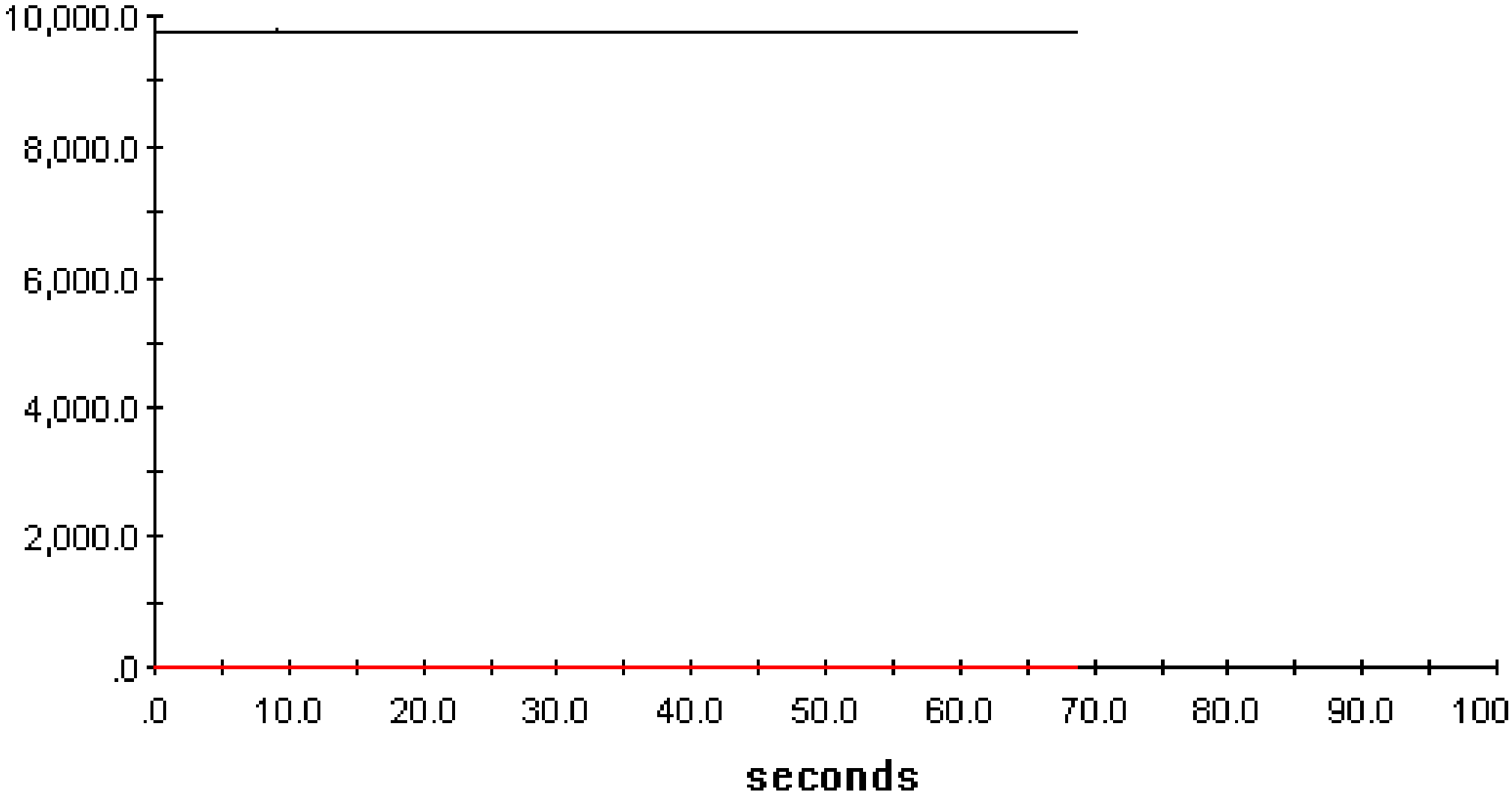
IPv4 50% Load (10Mbps) with 90 byte packets

IPv4 Latency



IPv4 100% Load (10Mbps) with 90 byte packets

IPv4 Packet Loss



No packet loss shown at 100% load

IPv4 Testing Conclusions

- No packet loss at 100% load
- Latency from 200 us – 645 us
- Appears to run well at wire rate (10 Mbps)

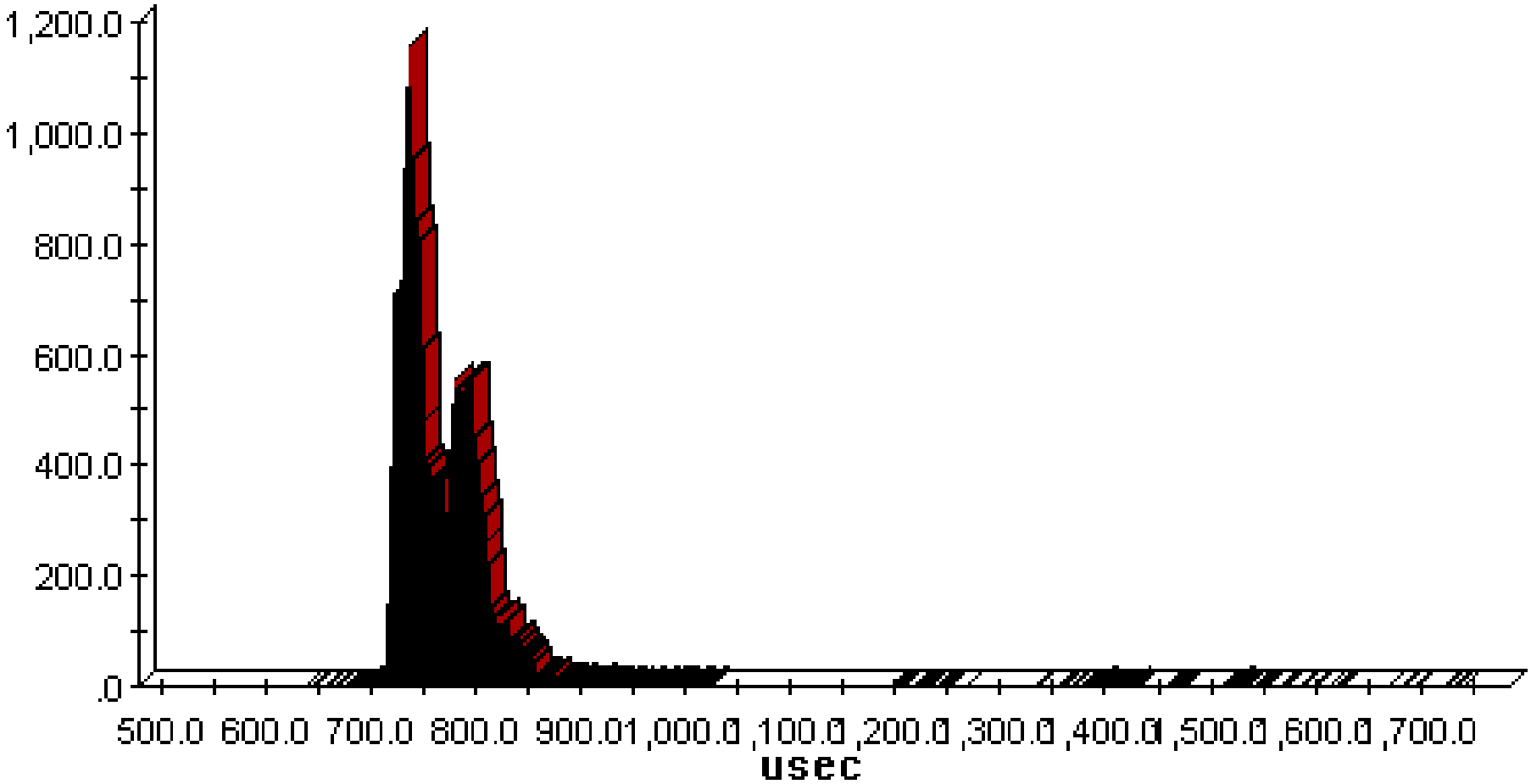
IPv6 Testing

- Basic throughput IPv6 only, single stack.

IPv6 Traffic Generation

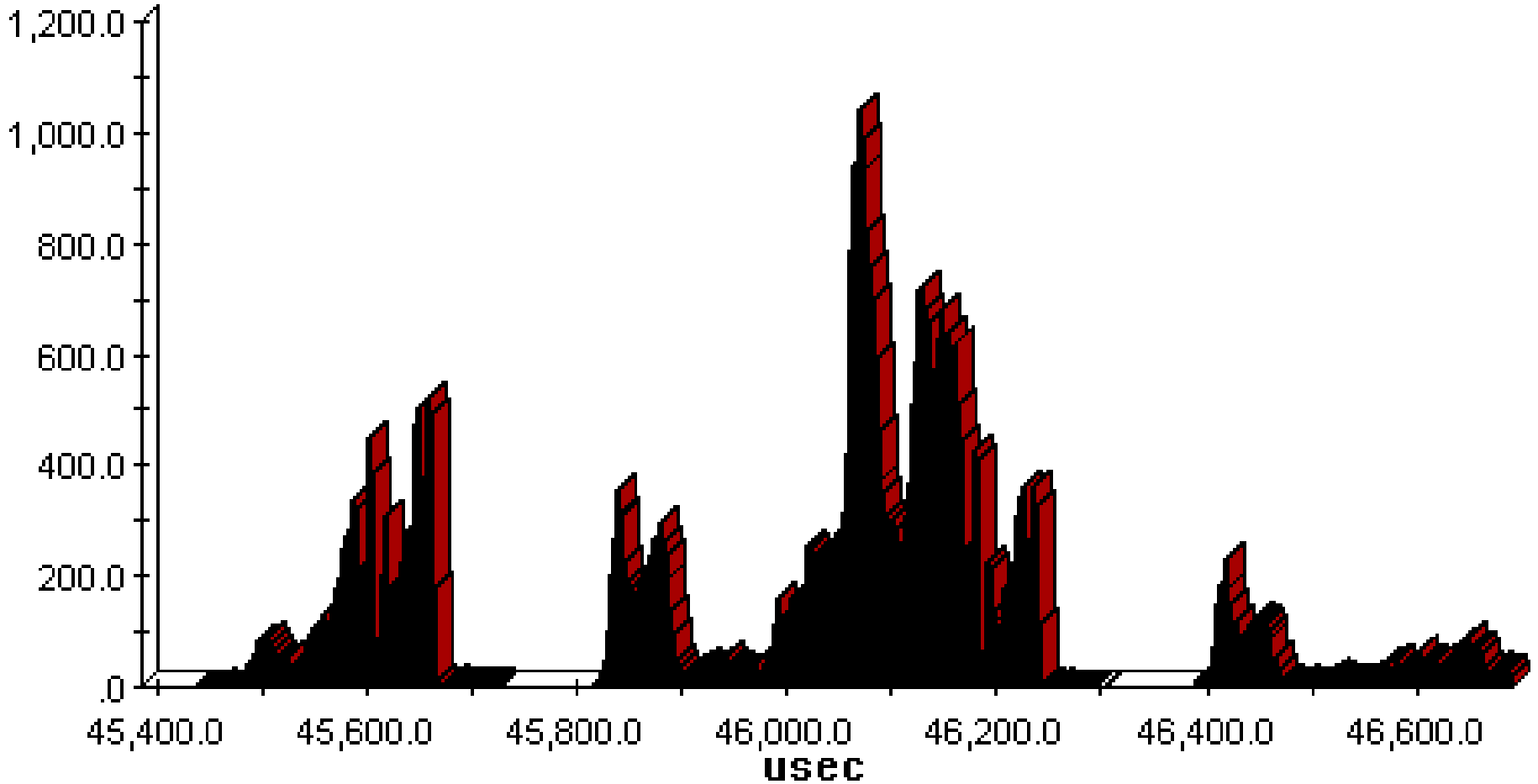
- **Measurements taken from 1% to 100% load (HDX)**
- **Quick test to gauge IPv6 performance**

IPv6 Latency



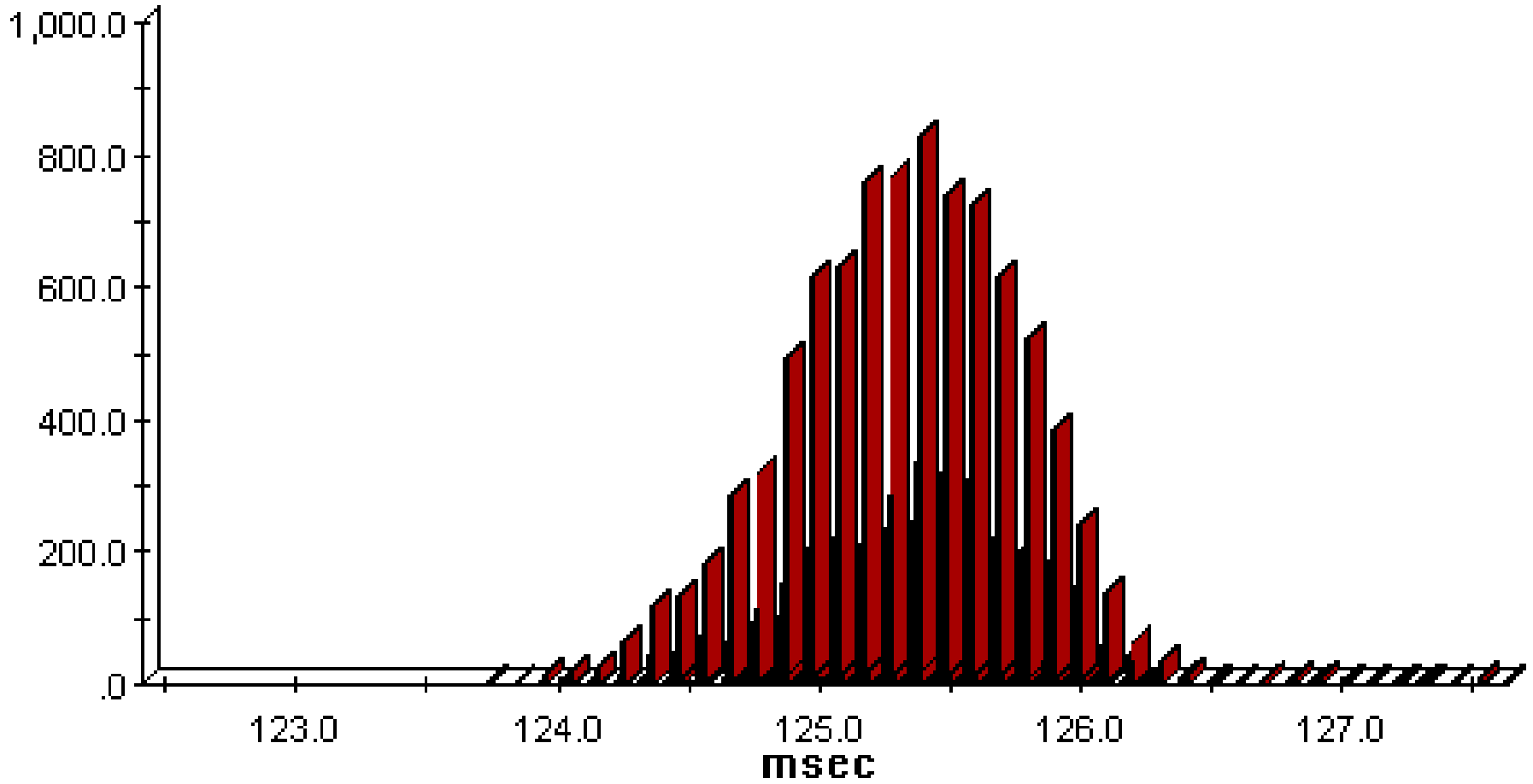
- IPv6 1% Load (10Mbps) with 90 byte packets
- Notice IPv6 latency three times that of IPv4

IPv6 Latency



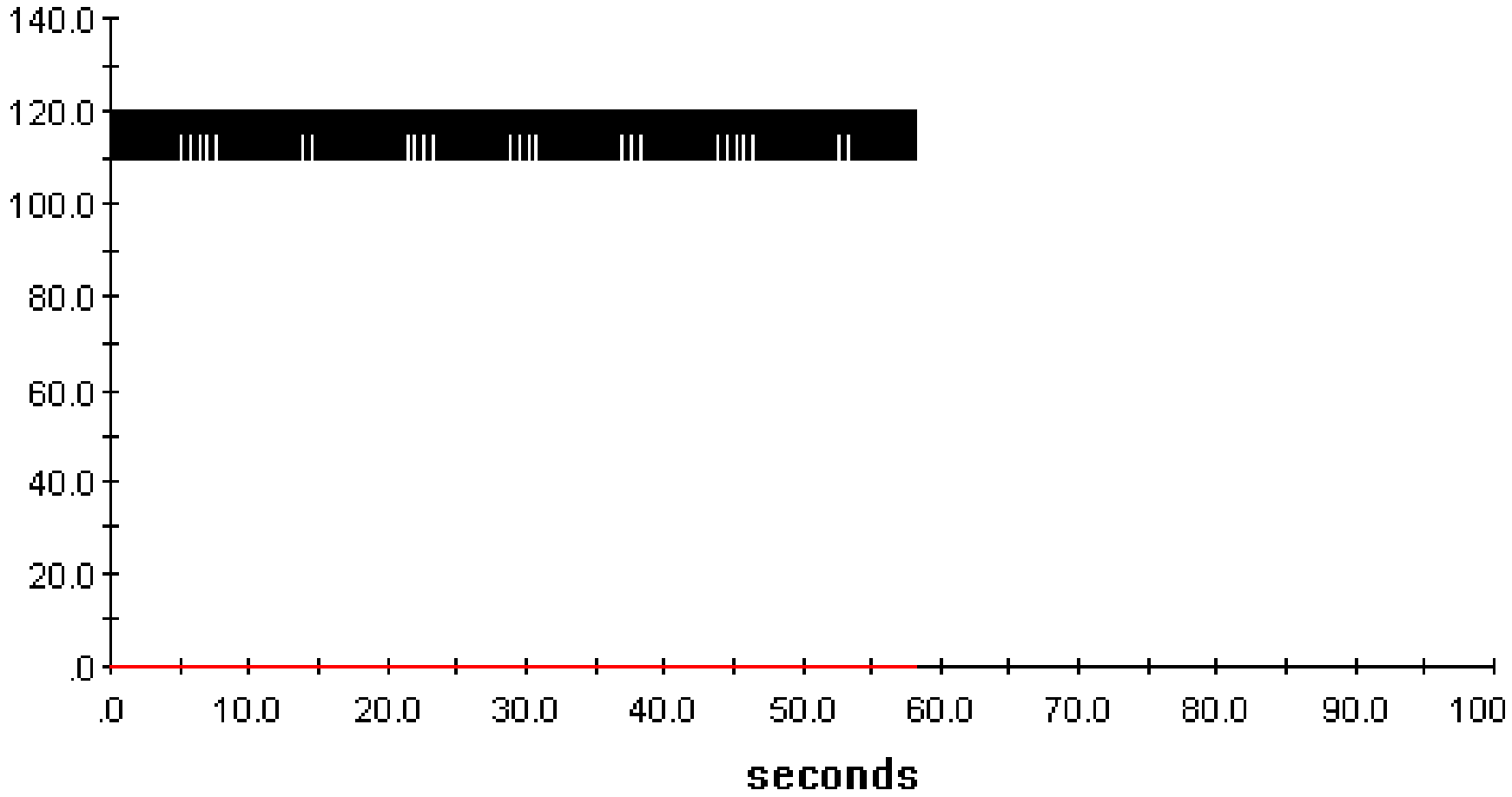
- **IPv6 15% Load (10Mbps) with 90 byte packets**
- **Latency through the router has greatly increased**

IPv6 Latency



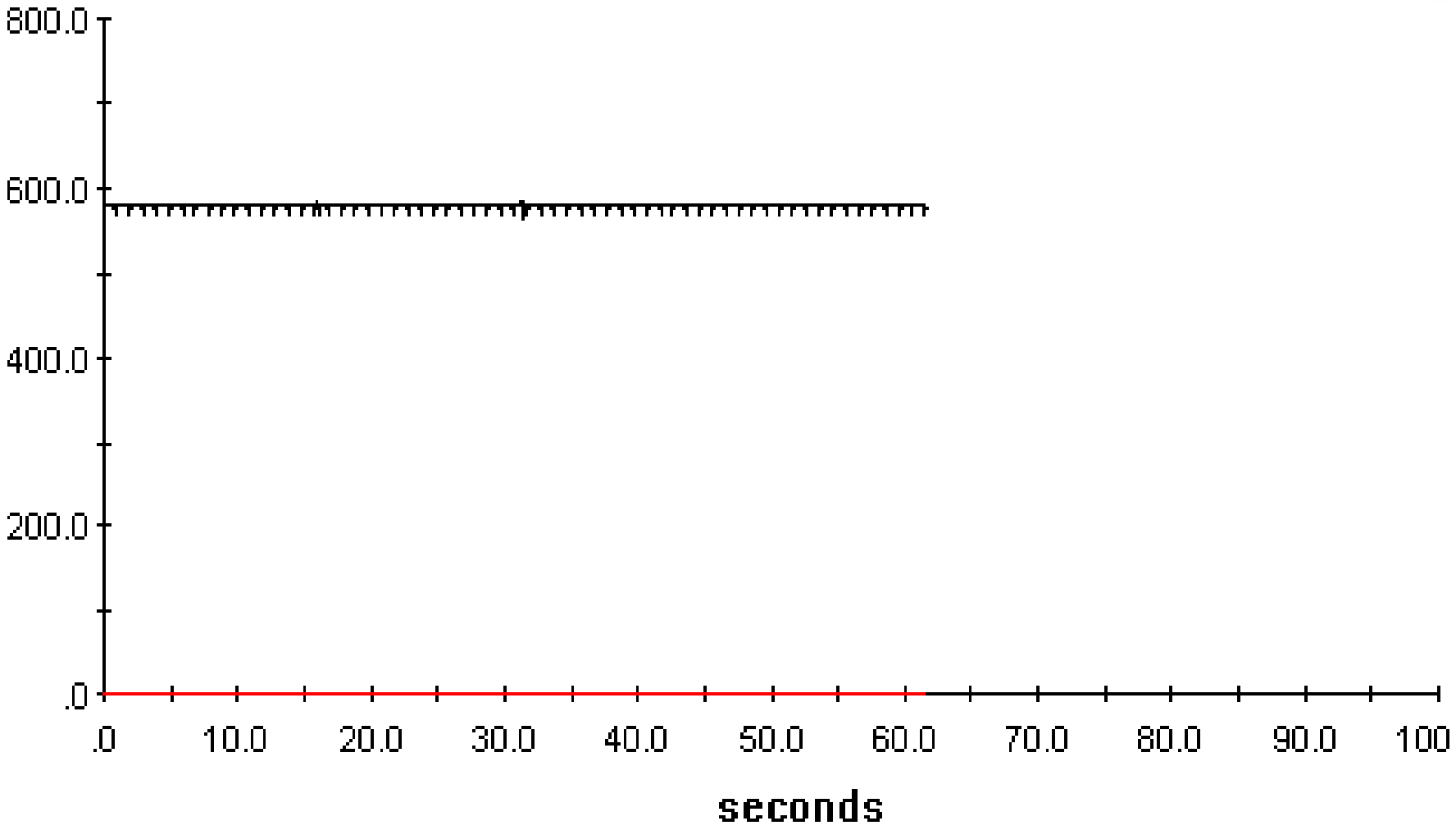
- IPv6 100% Load (10Mbps) with 90 byte packets
- The router is now experiencing over 125 ms of latency

IPv6 Packet Loss



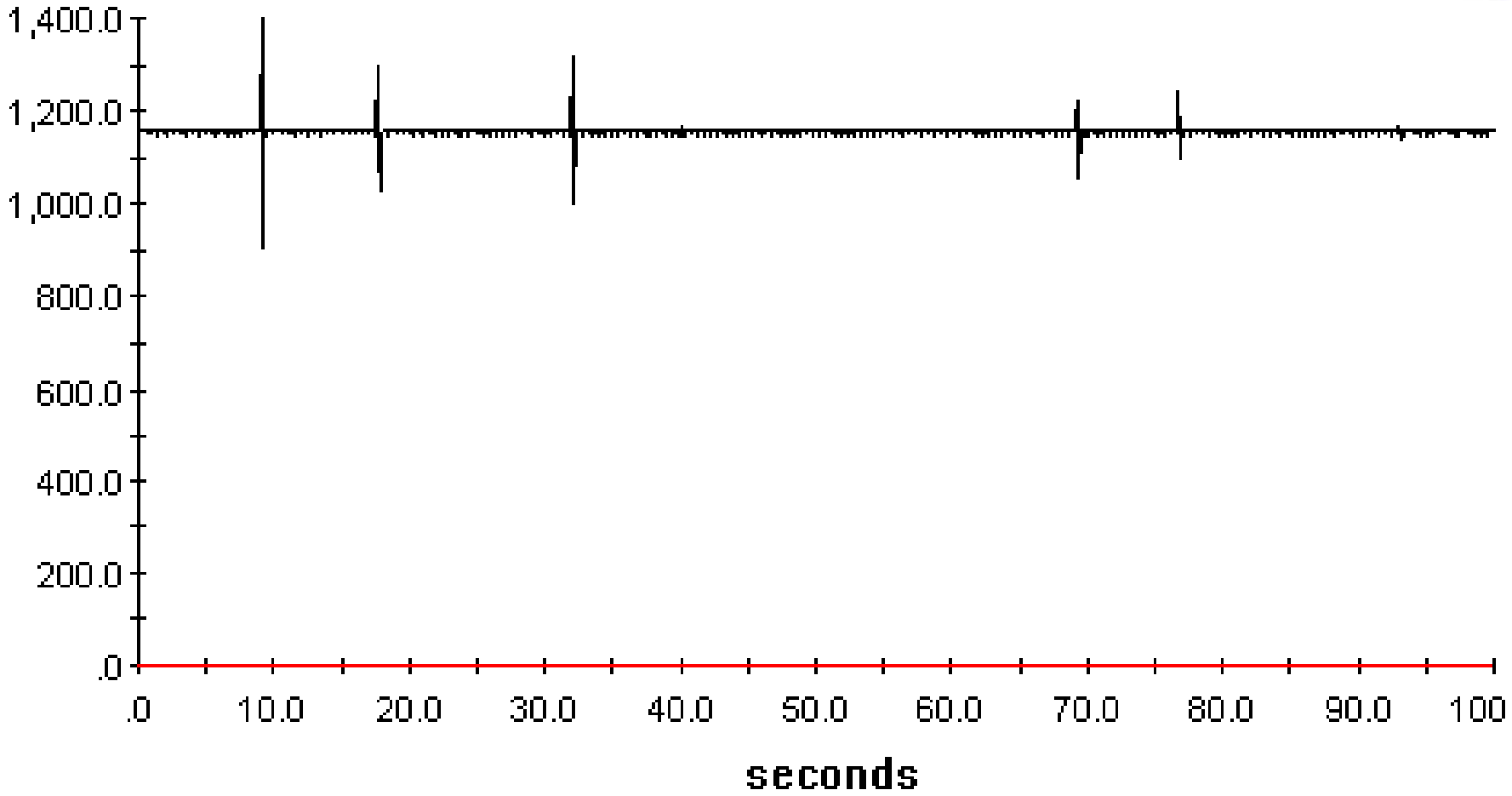
No packet loss shown at 1% load

IPv6 Packet Loss



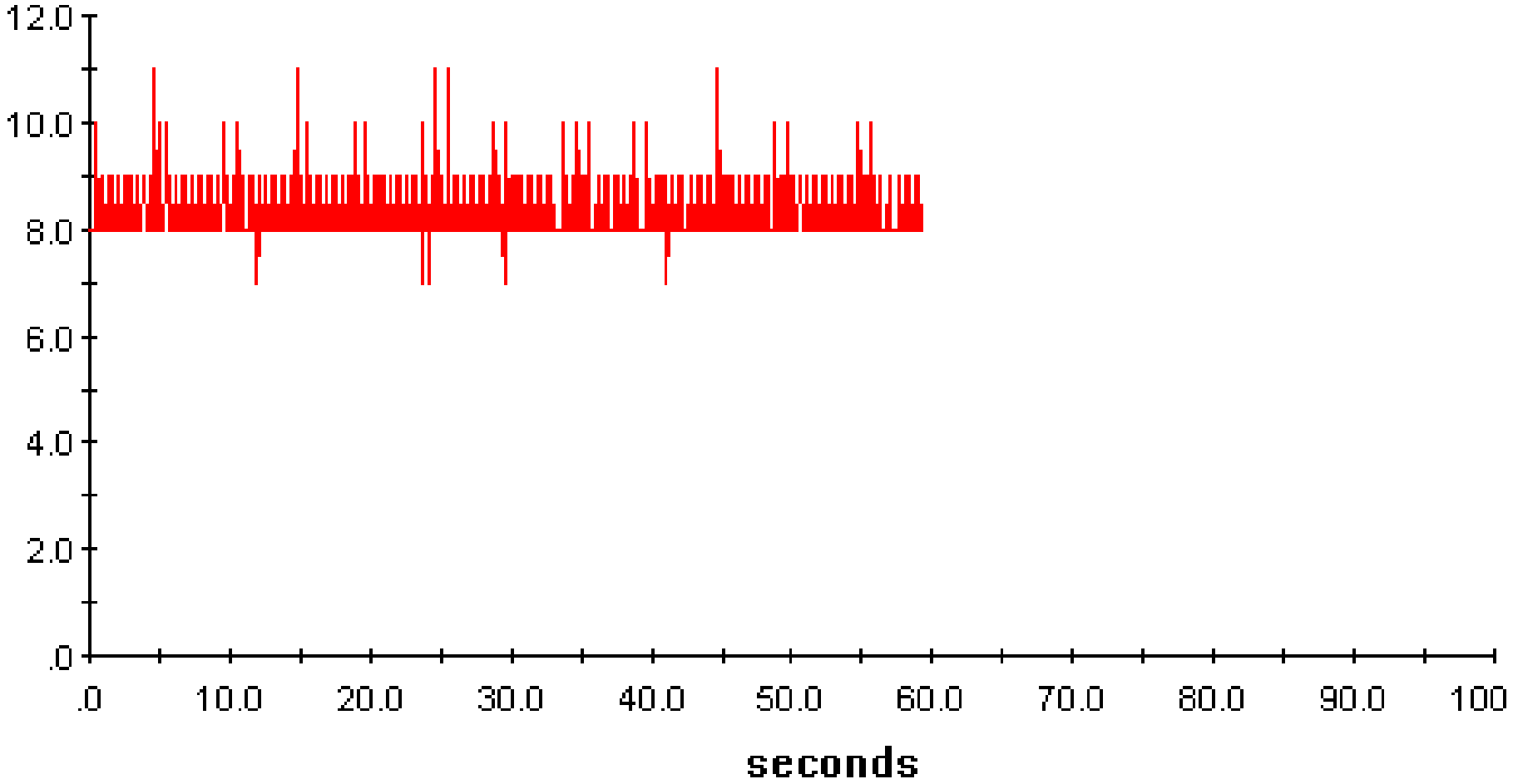
No packet loss shown at 5% load

IPv6 Packet Loss



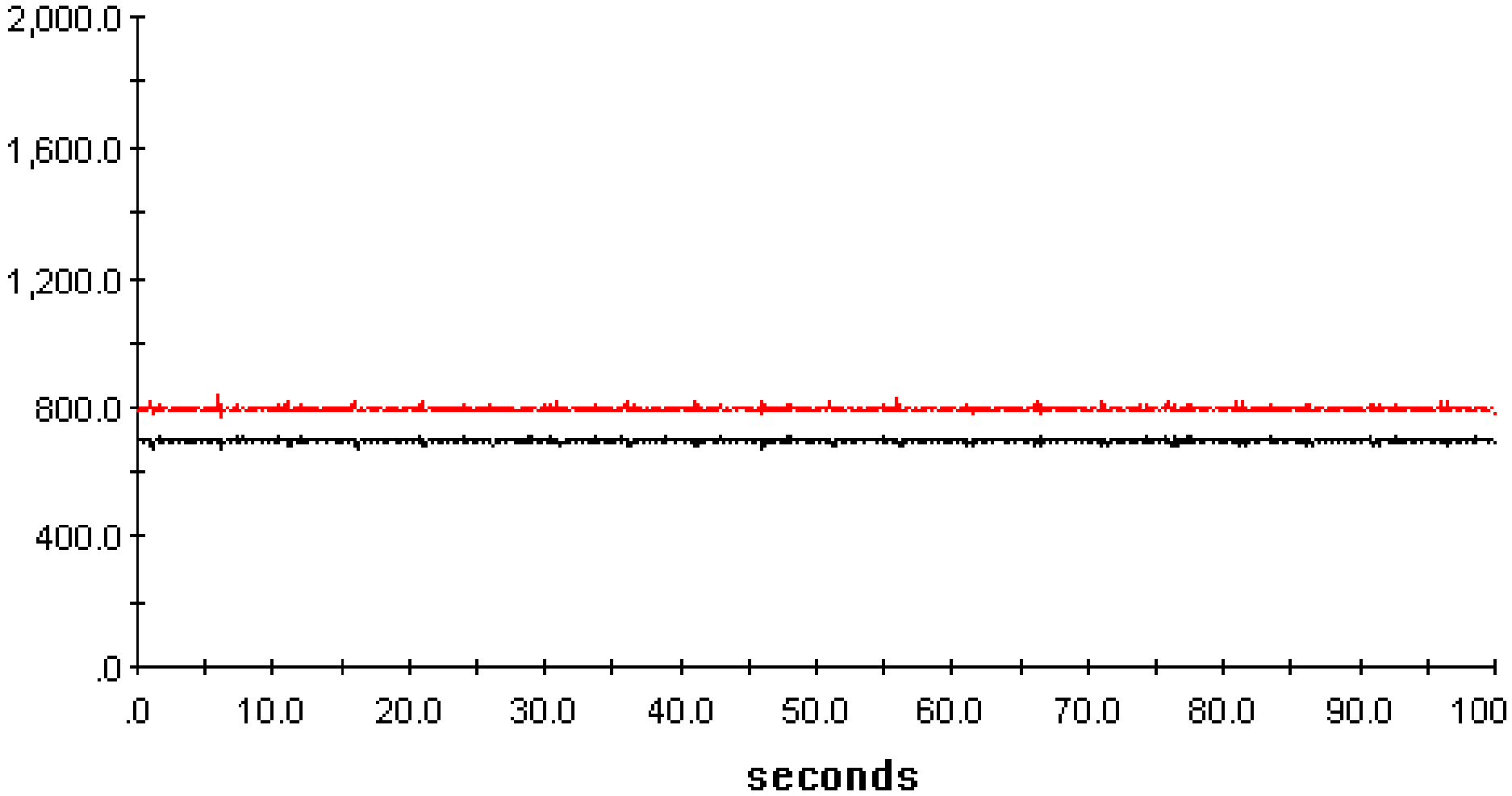
- At 10% packet load the router begins to have problems
- There is now a small packet loss (not shown due to scale)

IPv6 Packet Loss



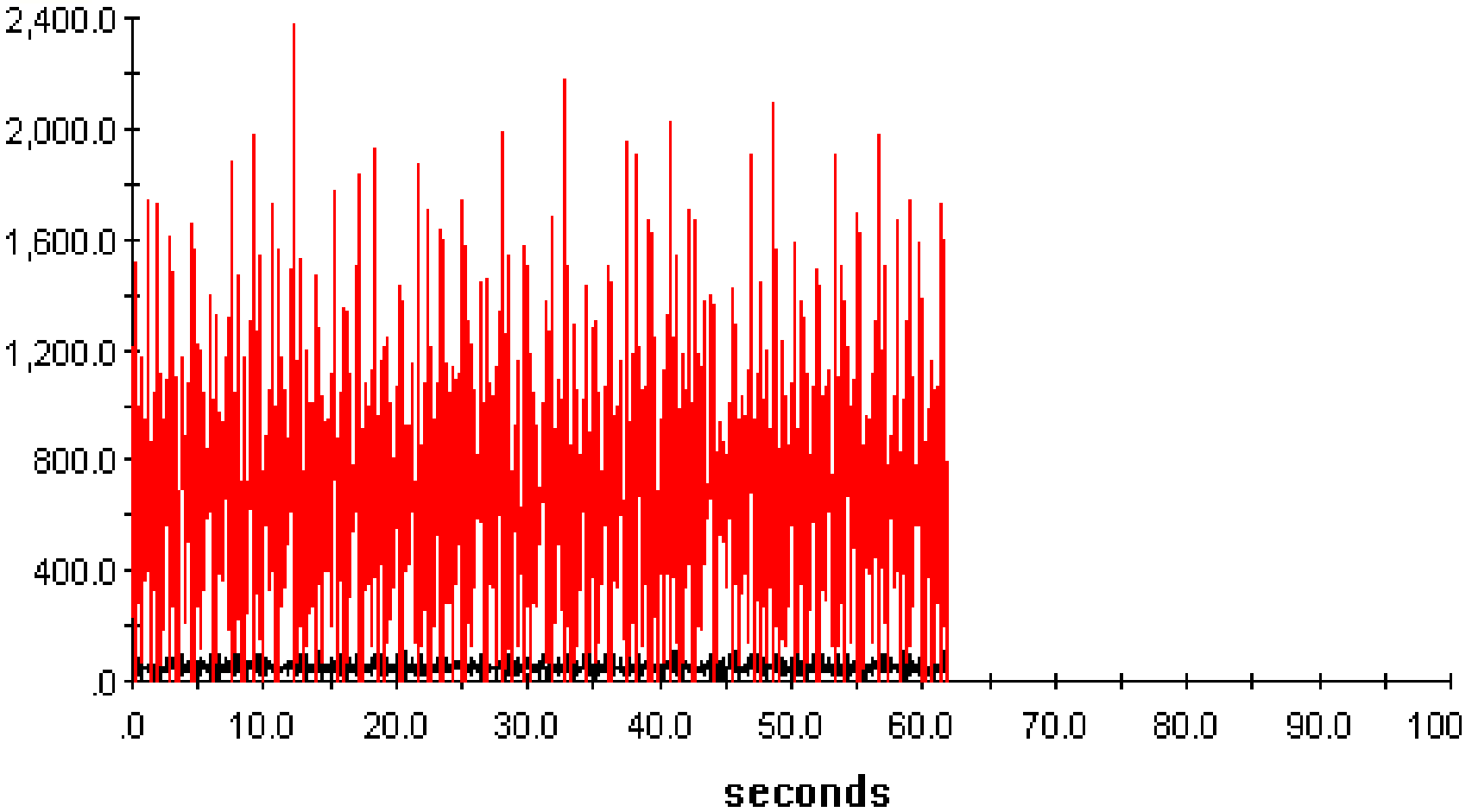
- **At 15% packet load the router begins to have problems**
- **Packet loss increases**

IPv6 Packet Loss



- **At 75% packet load the router has problems**
- **More packets are lost then received**

IPv6 Packet Loss



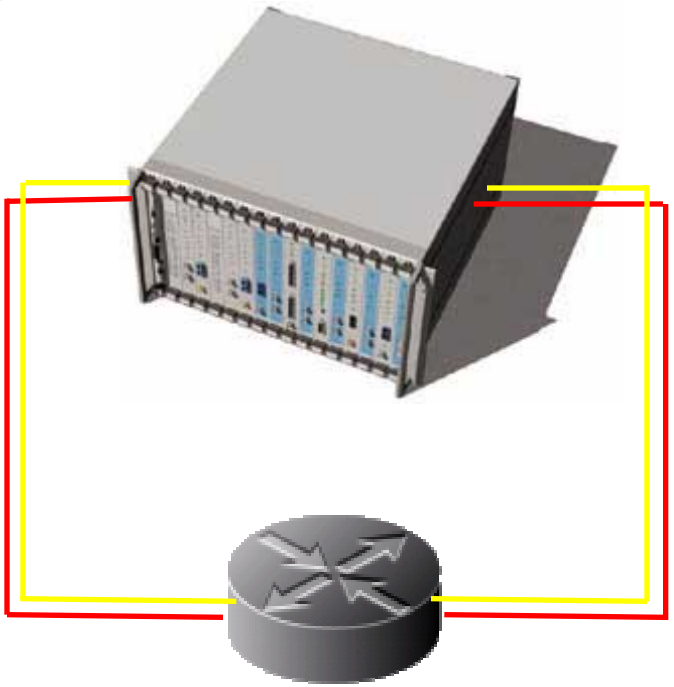
- At 65% packet load FDX the router has major problems
- Very few packets are successfully received

IPv6 Testing Conclusions

- IPv6 Performance very poor
- At very light loads latency is more then 3X IPv4
- At just 15% load latency will be unacceptable
- Packet loss is very high at 65% load FDX
- At 3% packet loss TCP can slow down by 50%
- Although this router supports IPv6 it would only be suitable for experimental testing. Not recommended for a production environment.

IPv6 Testing

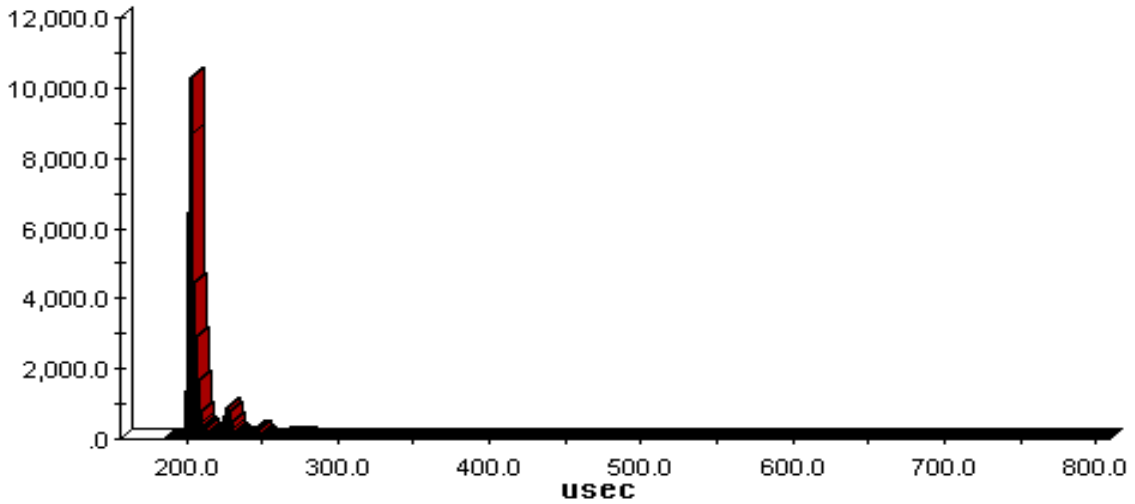
- Basic throughput IPv6 & IPv4, Dual stack.



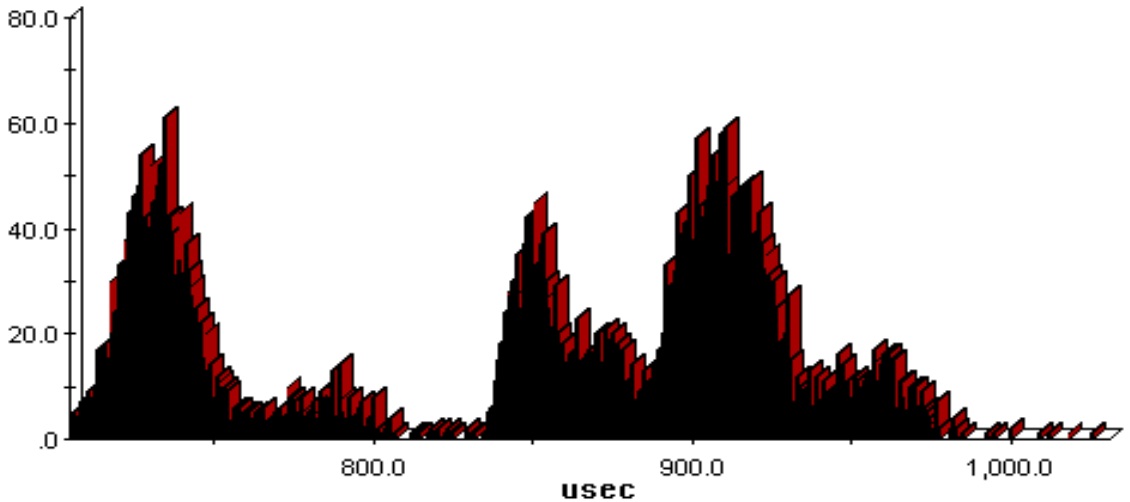
IPv6 & IPv4 Traffic Generation

IPv6 Dual Stack Latency Testing

IPv4



IPv6

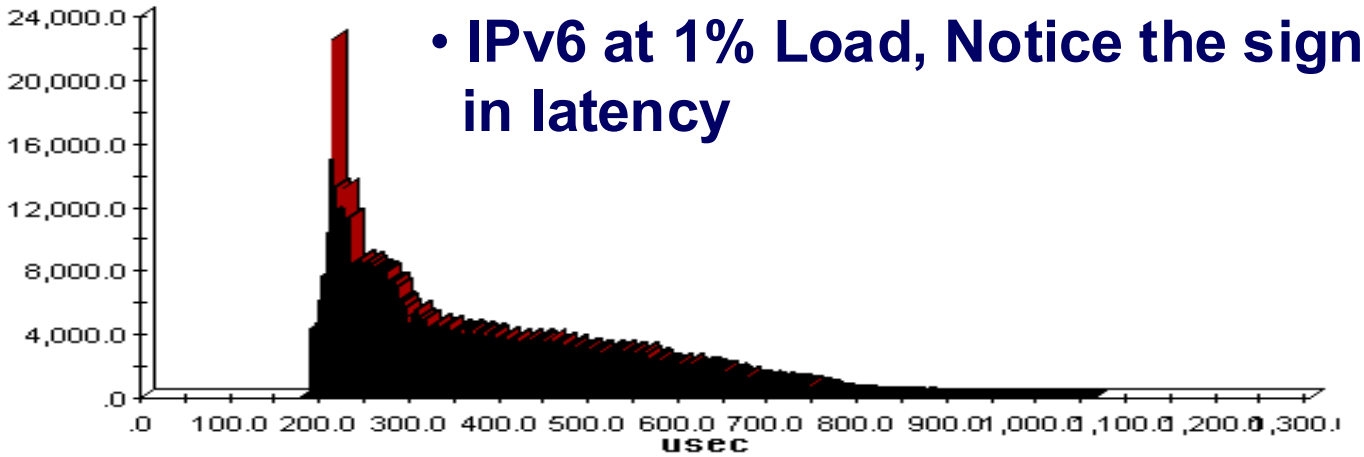


- **IPv4 at 10% Load**
- **IPv6 at 1% Load, Notice the difference in latency**

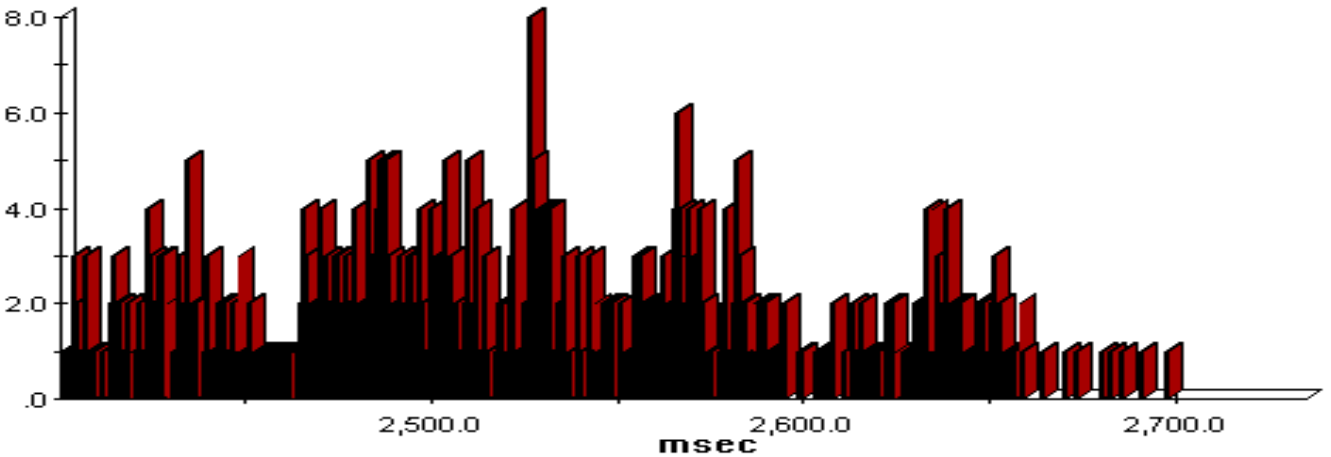
IPv6 Dual Stack Latency Testing

- IPv4 at 75% Load
- IPv6 at 1% Load, Notice the significant increase in latency

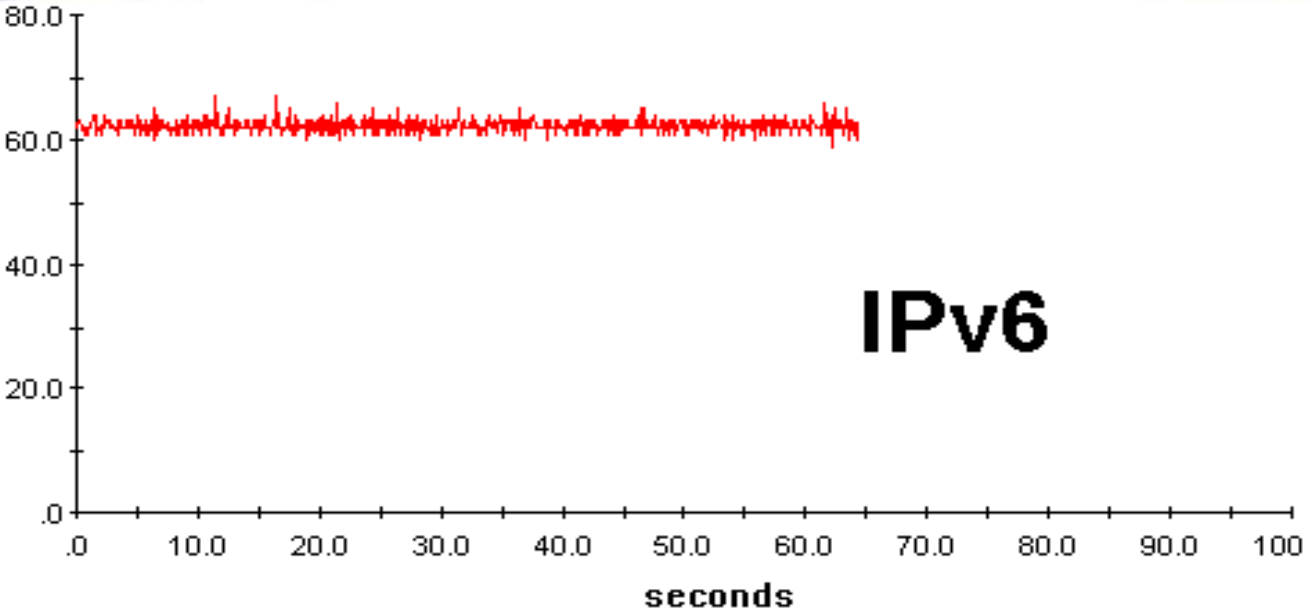
IPv4

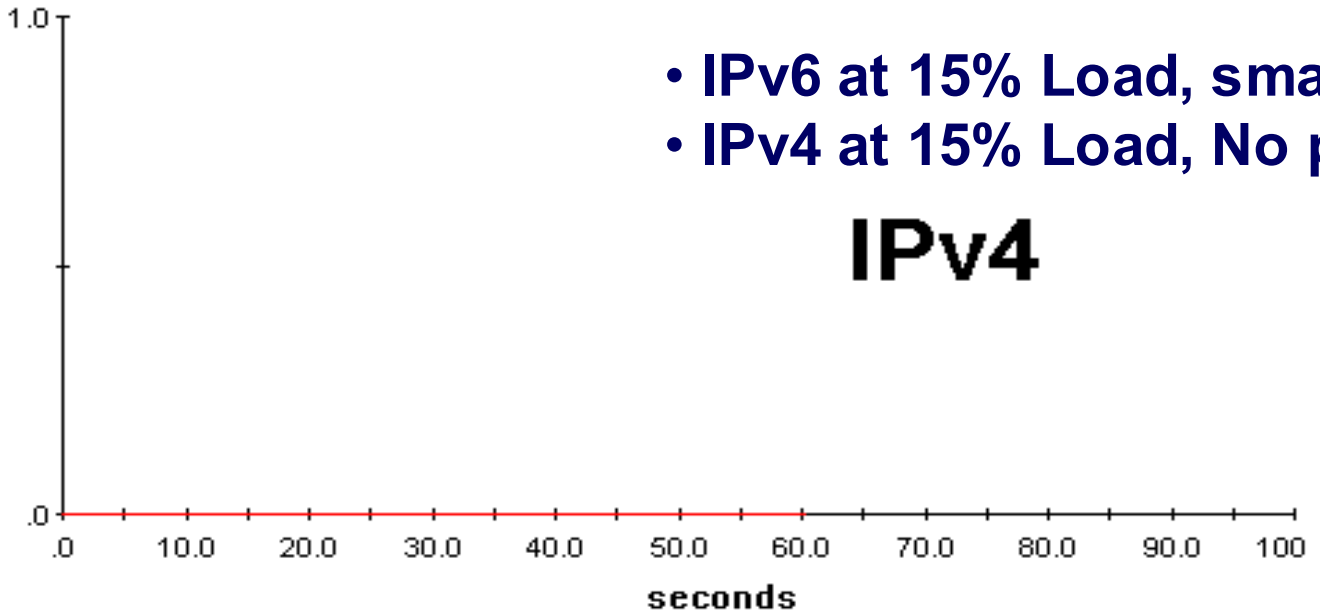


IPv6



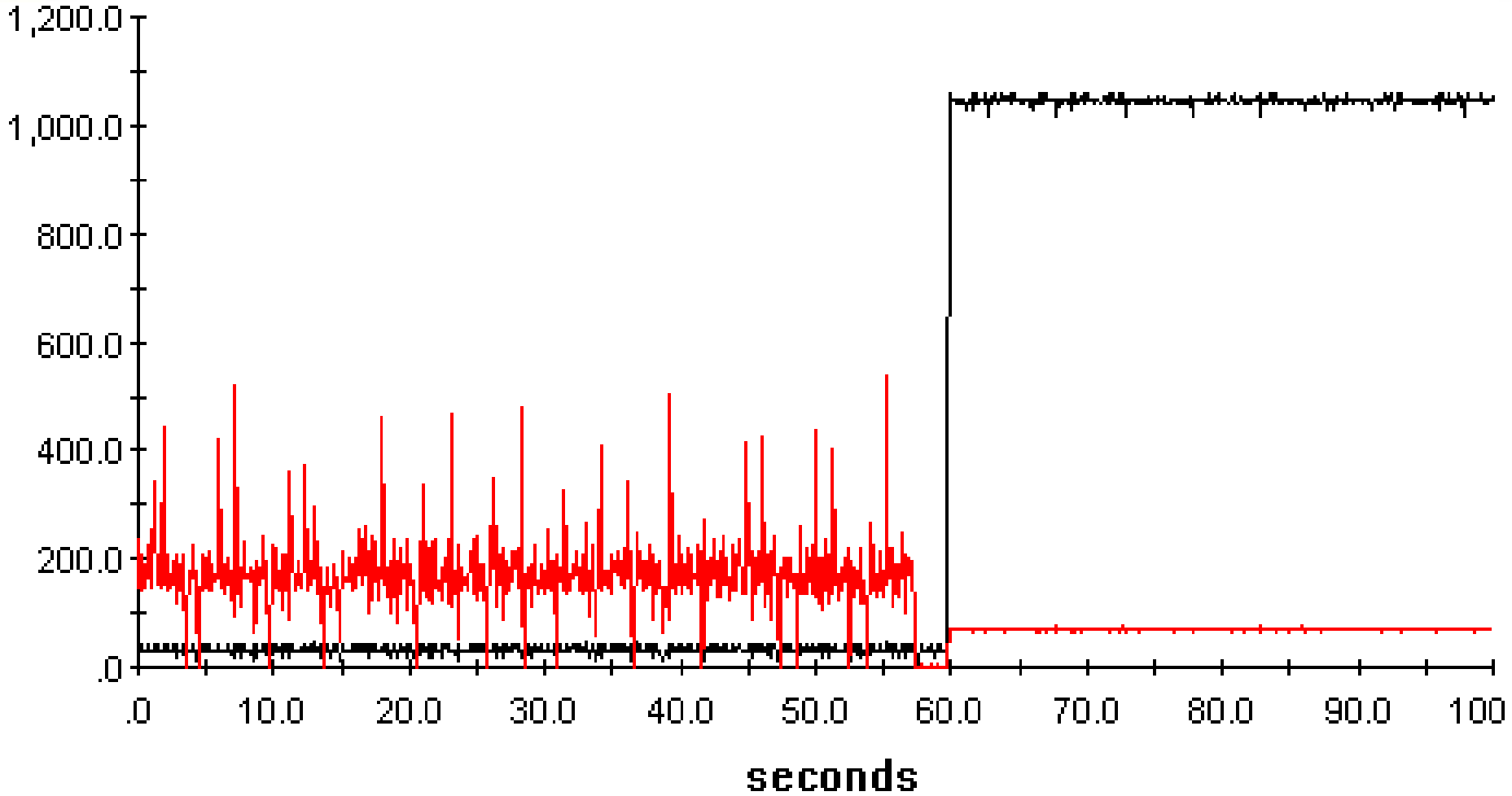
IPv6 Frame Loss





- IPv6 at 15% Load, small packet loss
- IPv4 at 15% Load, No packet loss

IPv6 Dual Stack Frame Loss



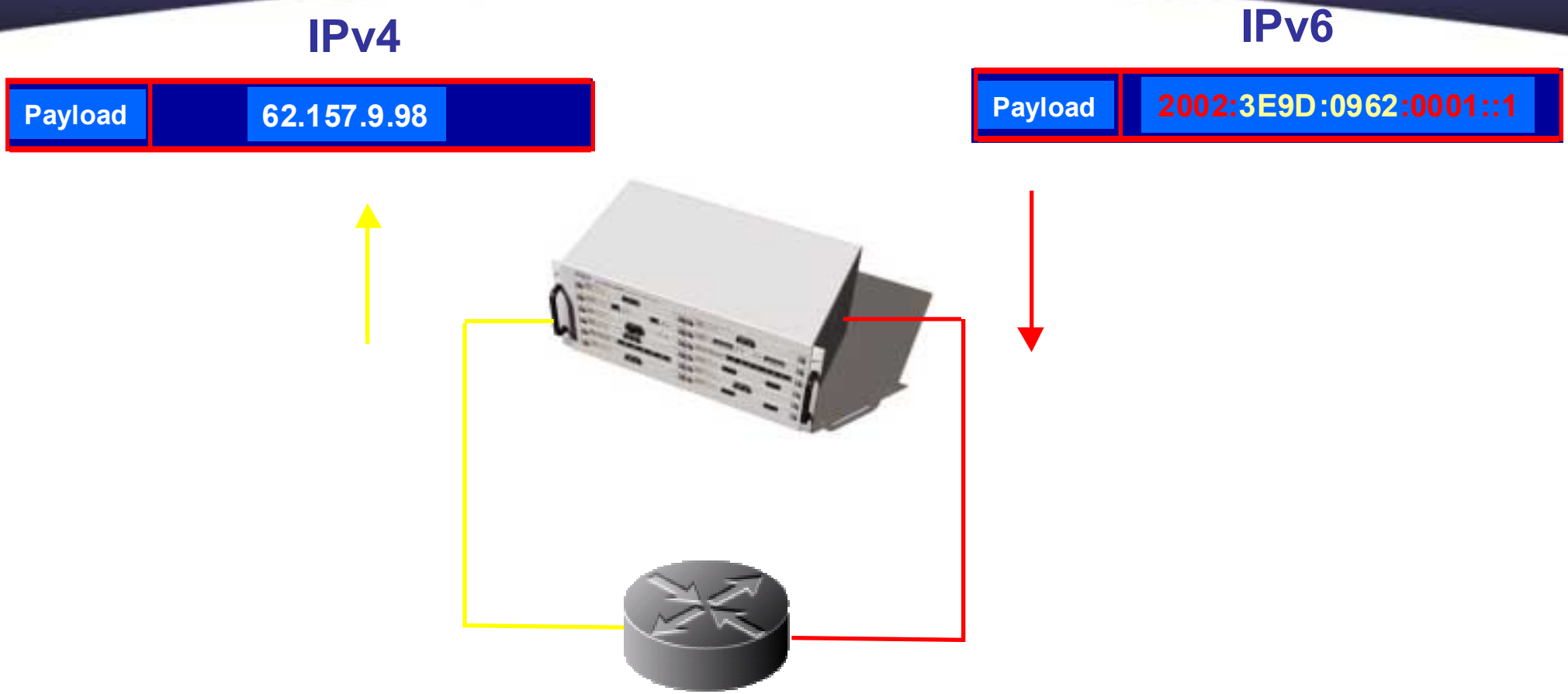
- **Increase IPv4 15% to 75% Load**
- **IPv6 at 15% Load, significant packet loss**

IPv6 Dual Stack Latency Testing

Conclusions

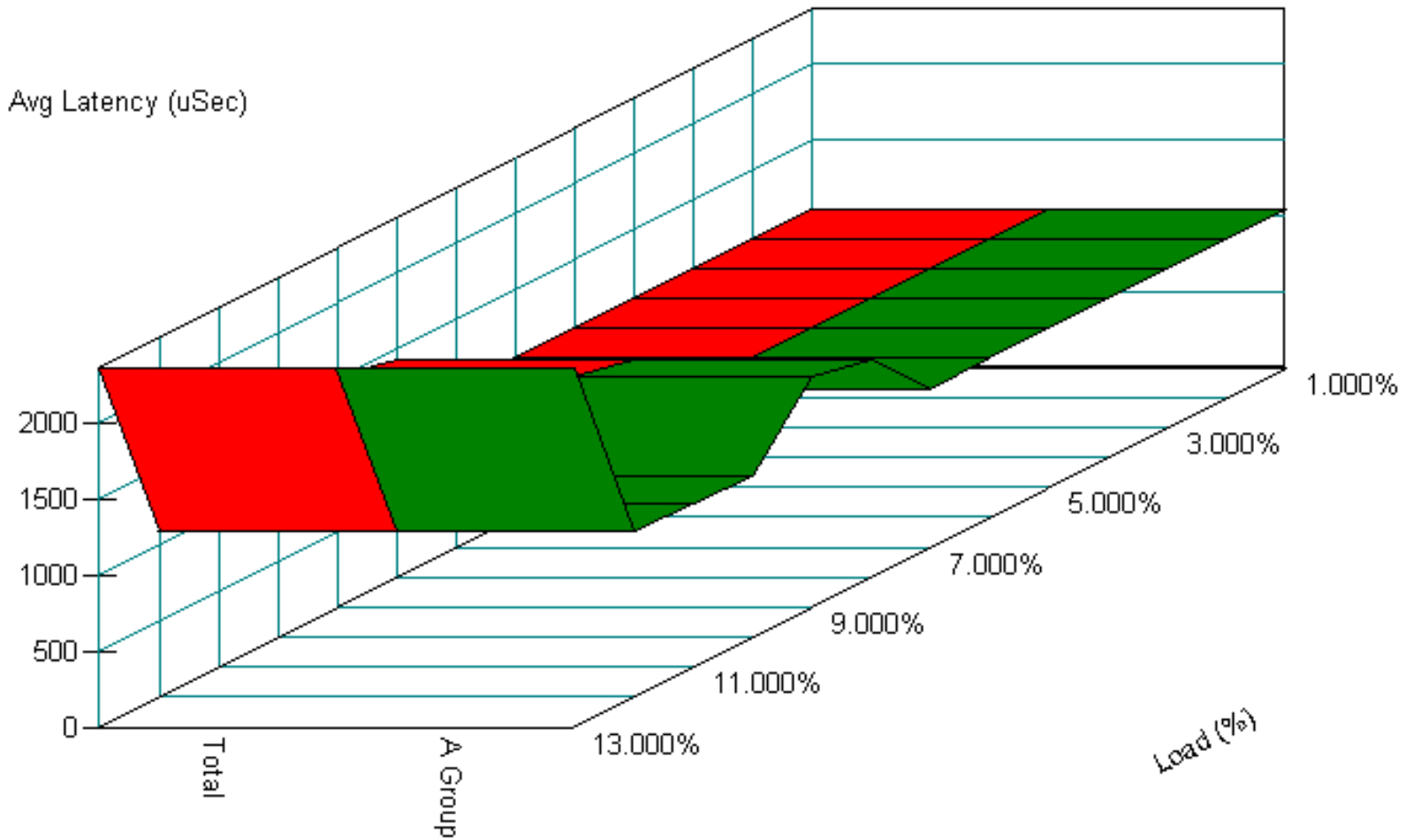
- This router has (as expected) very poor performance in dual stack for IPv6.
- Heavy IPv4 traffic has a significant impact on even light IPv6 loads.
- This router would not be suitable in an IPv6/IPv4 production environment.

Transition Method “6to4”Testing



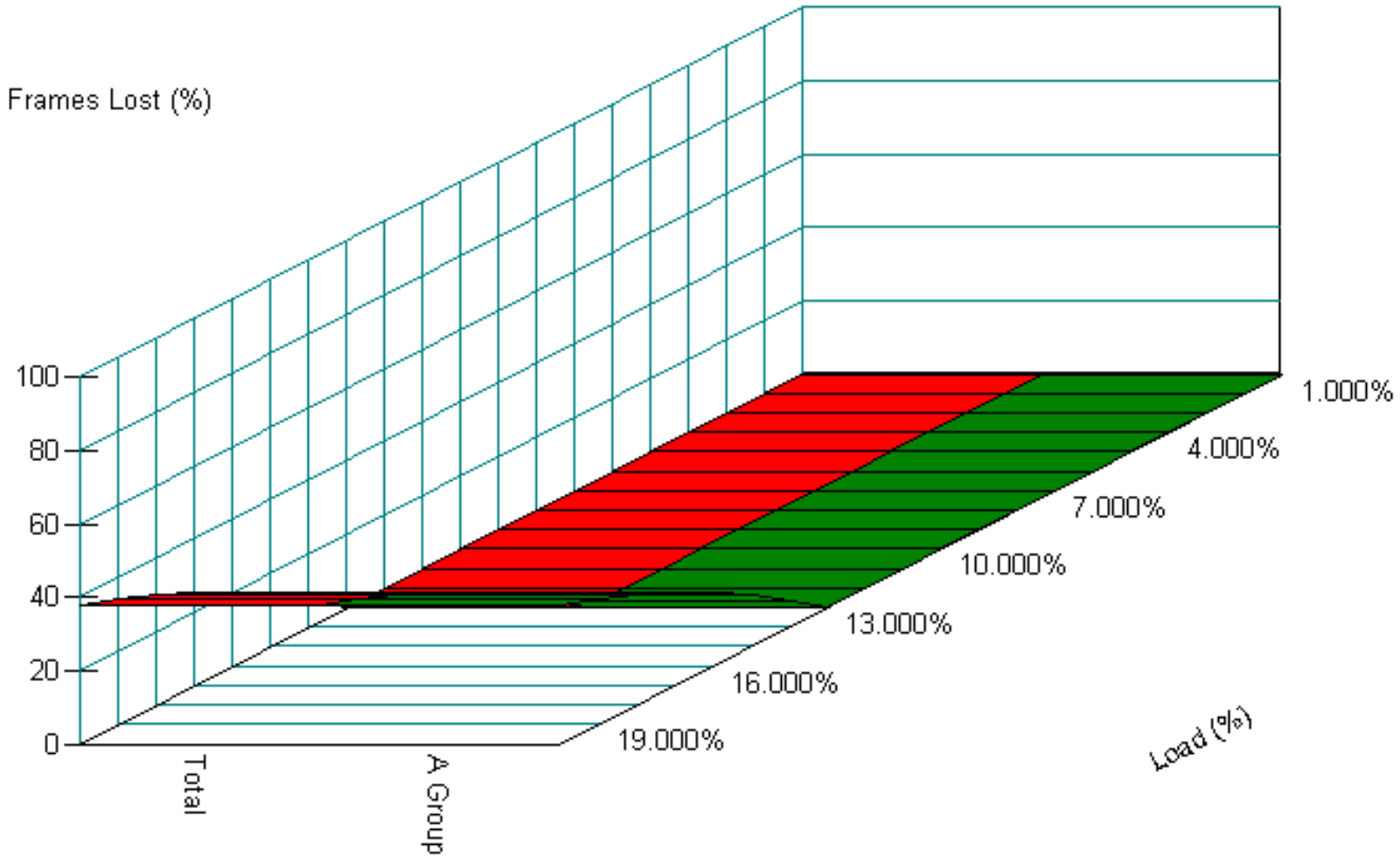
RFC- 3056 Testing “6to4”

Transition Method “6to4”Testing



6to4 Latency Test

Transition Method “6to4”Testing



6to4 Frame Loss Test

IPv6 Performance

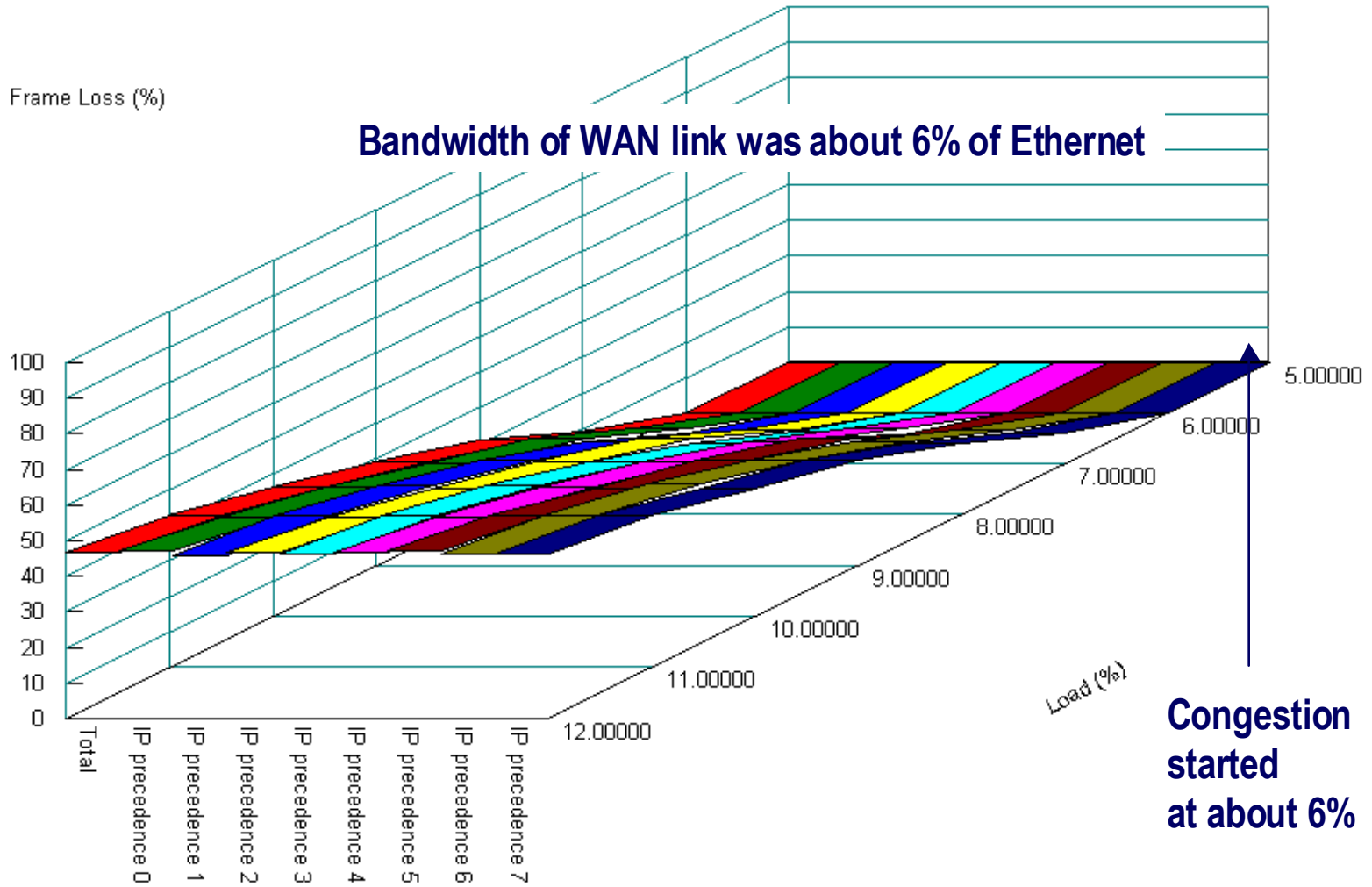
Additional Future Testing

IPv6 Testing with QoS

- Basic throughput IPv6 only, single stack.

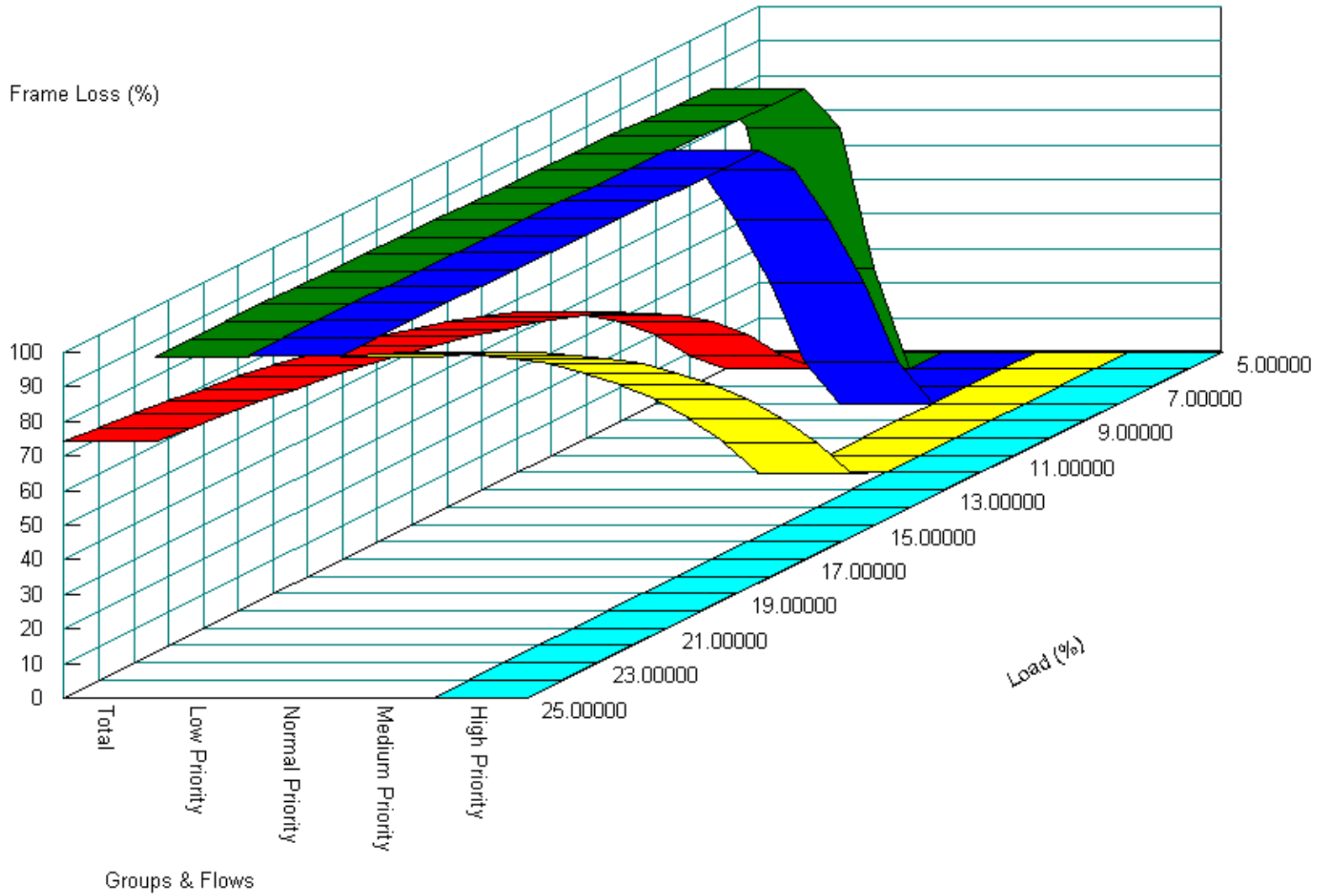
IPv6 Traffic Generation

FIFO - No QoS

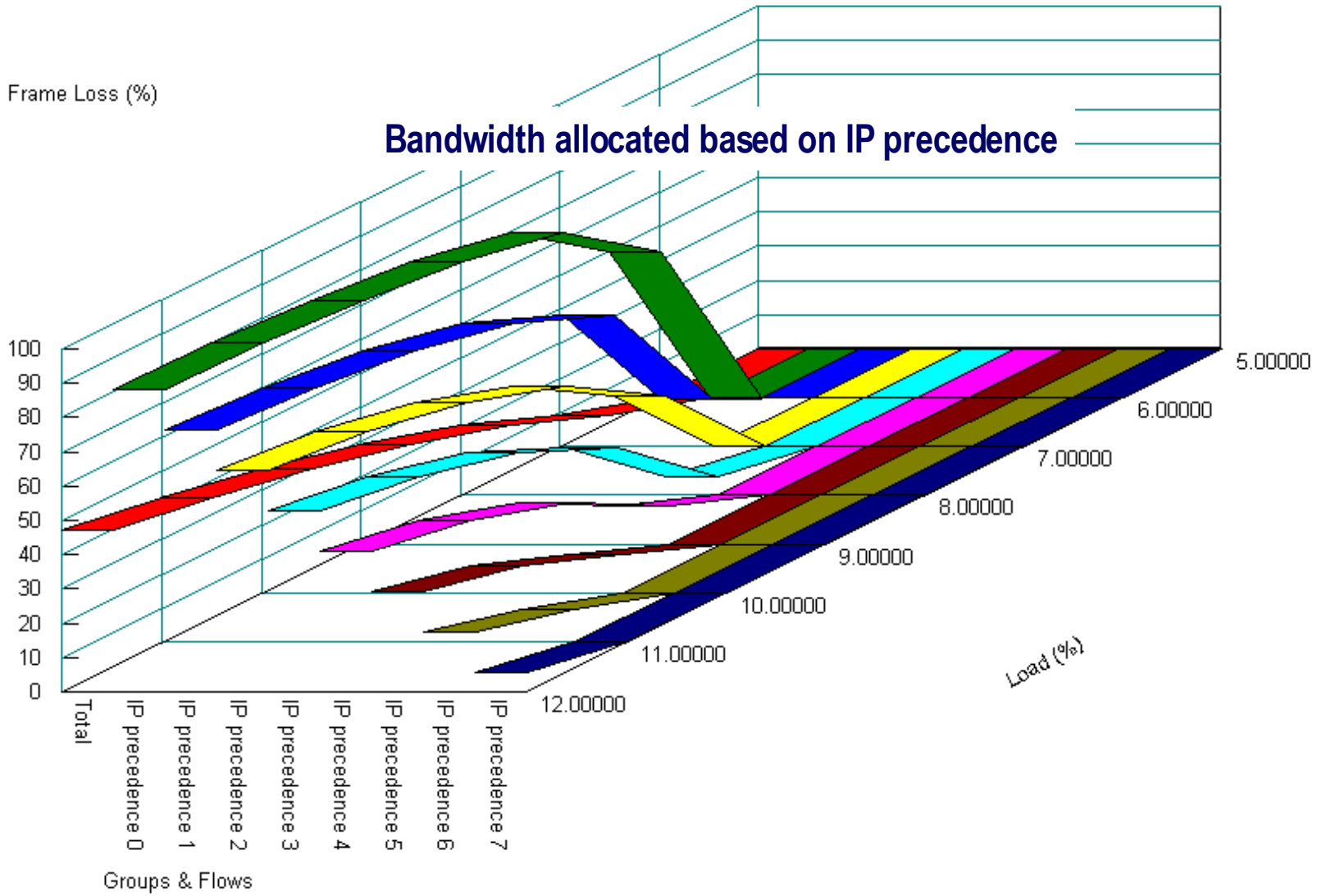


Note: In all the tests, input load was specified as % of Ethernet

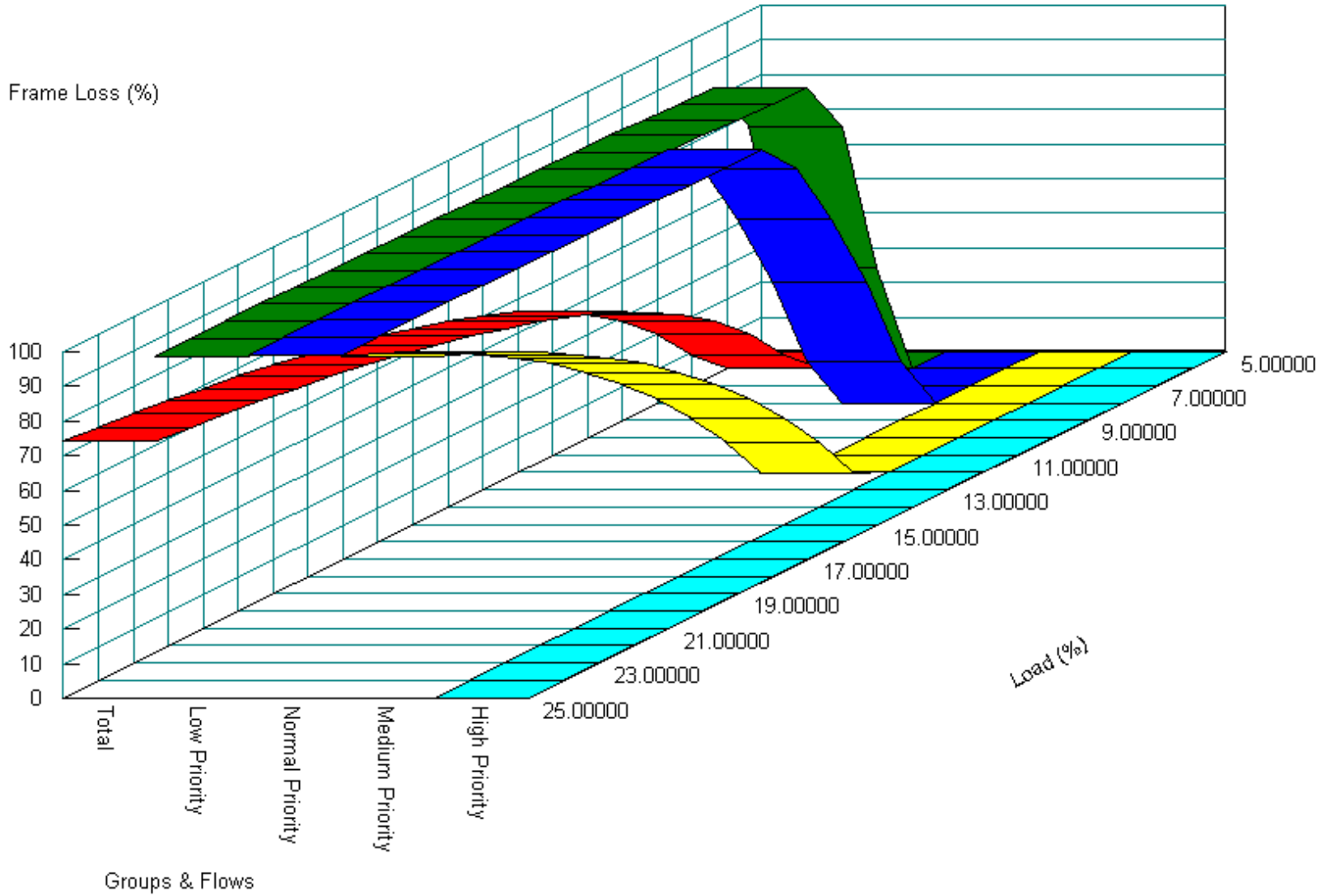
Classical Priority Queuing



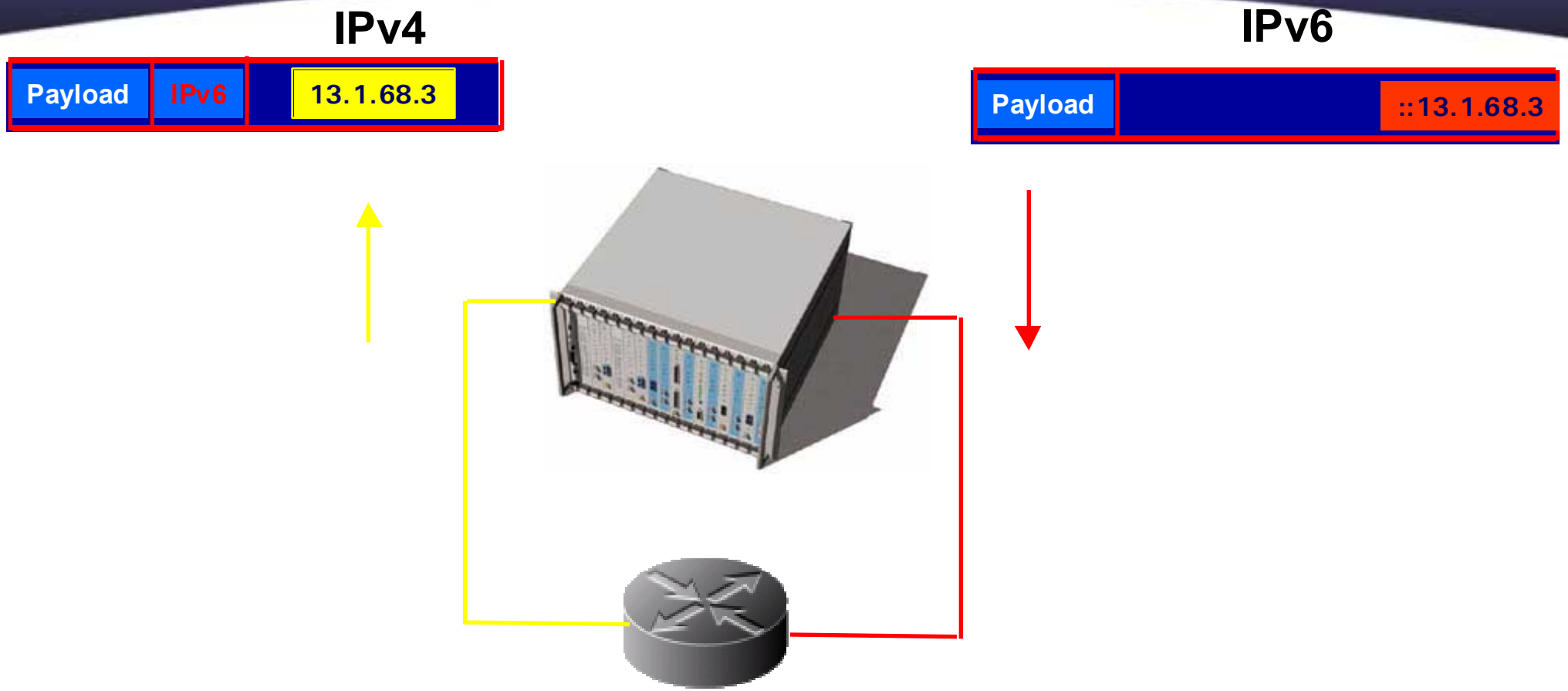
WFQ and Frame Loss



Classical Priority Queuing

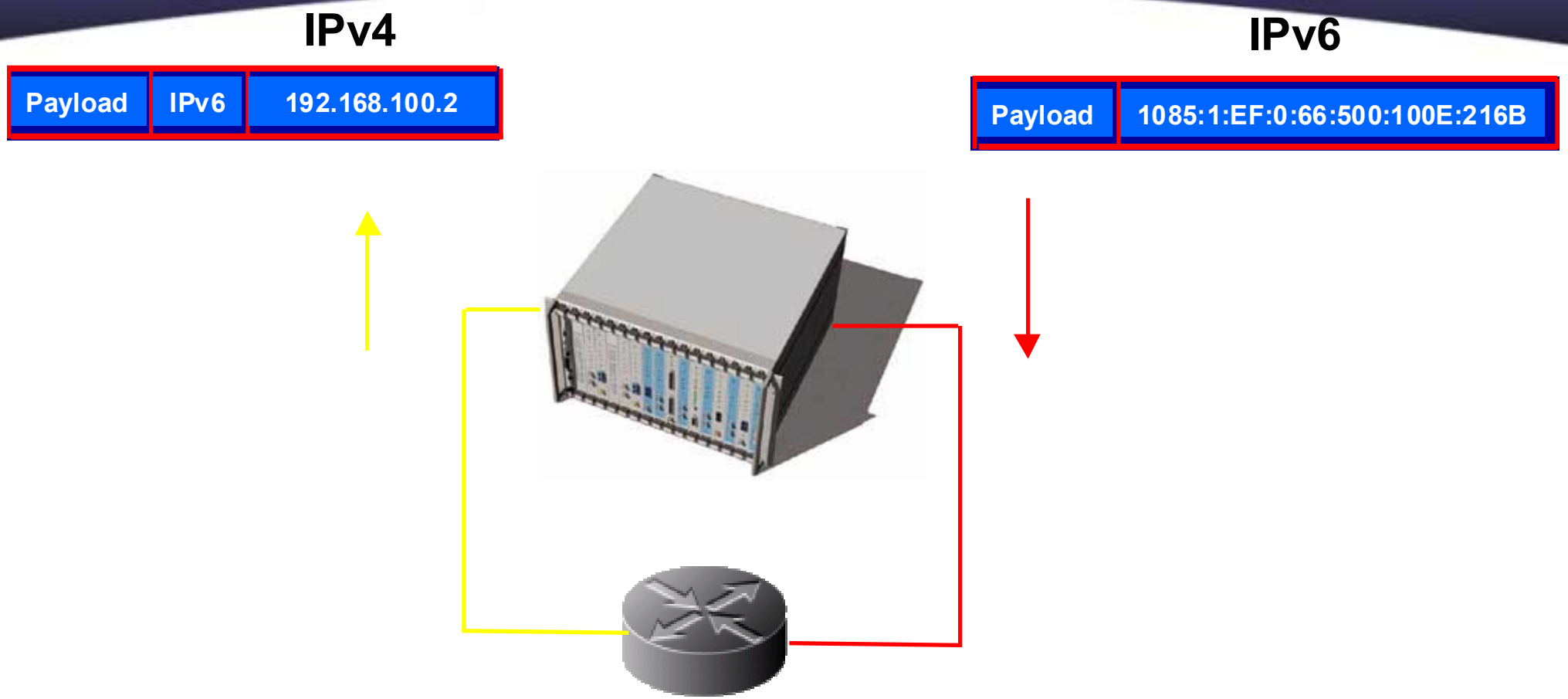


Tunneling Method 1: Automatic



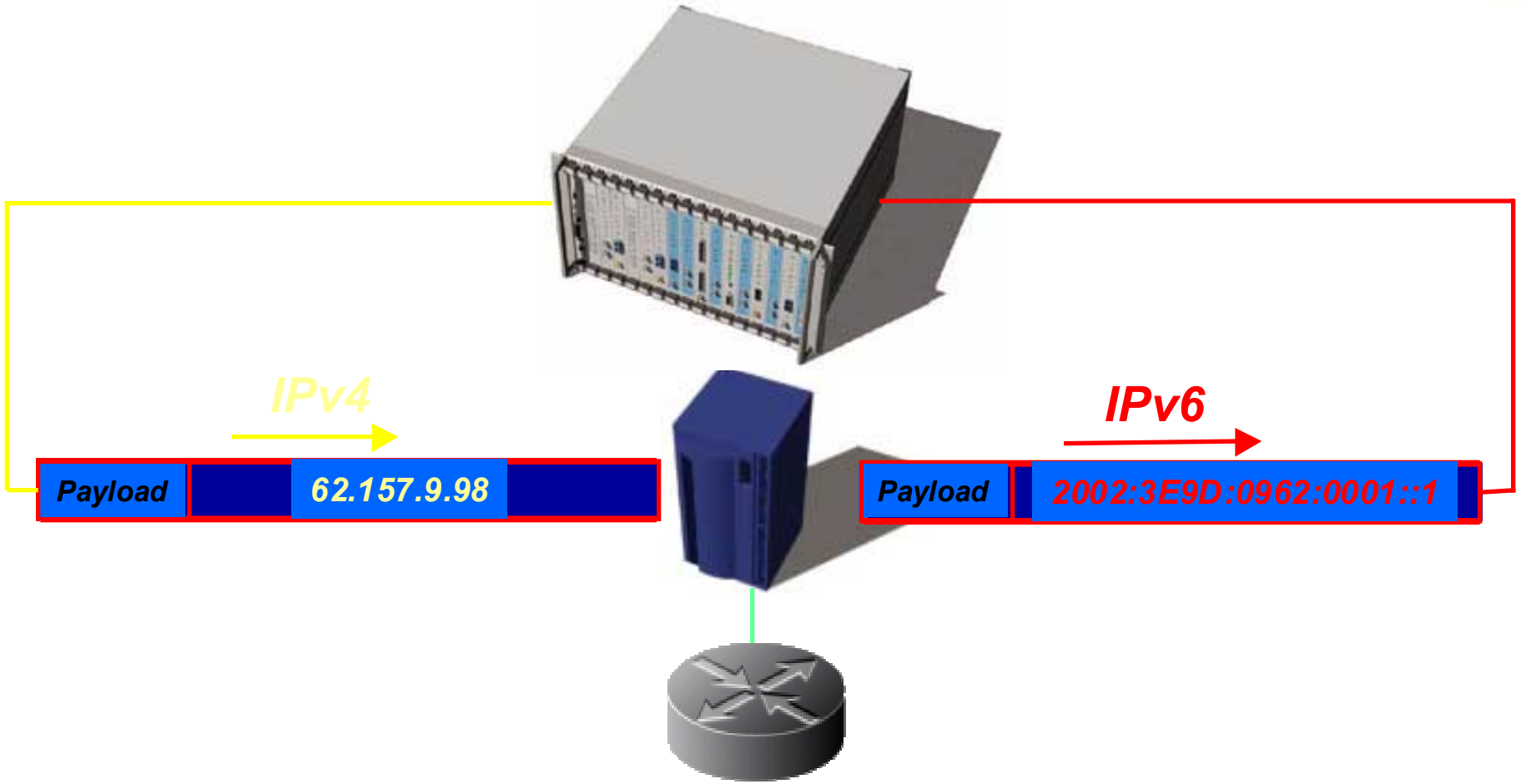
RFC 2893 Testing "6over4"

Tunneling Method 2: Configured



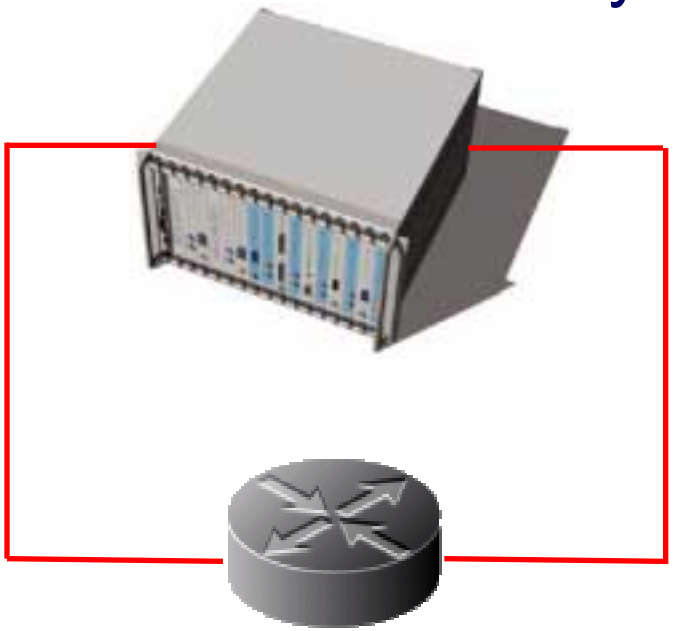
RFC 2893 Testing "6over4"

NAT-PT Testing



Router Alert Performance Testing

- By transmitting a Router Alert storm, you can test the routers ability to maintain its ability to maintain throughput



Thank You