IPv6: The New Version of the Internet Protocol

Steve Deering
Cisco Systems, Inc.
deering@cisco.com

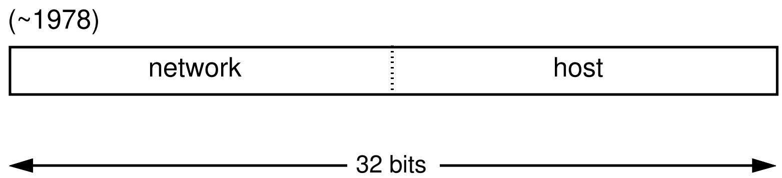
1

How "Real" Is IPv6?

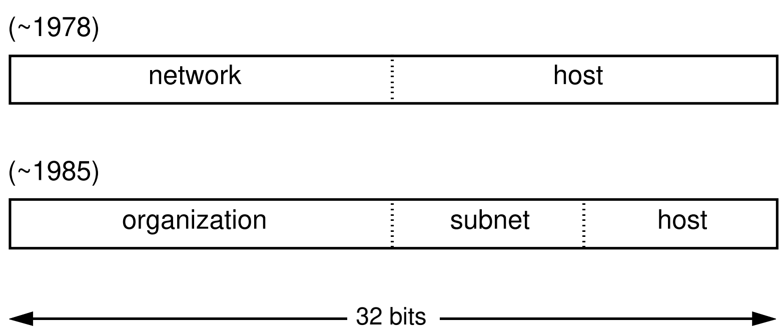
- > 20 vendors and organizations are doing implementations
  - several beta releases available
  - one router vendor (Telebit, Denmark) is already shipping
- multi-vendor demos (Interop) and interoperability tests (UNH) have been done; more are planned
- the fledgling "6Bone" is up and running
- expect wide availability within two years

2

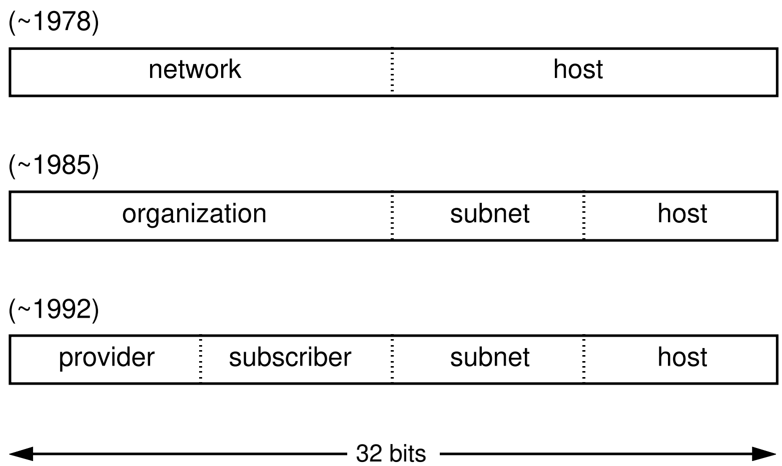
The IP Address Hierarchy



The IP Address Hierarchy



The IP Address Hierarchy



5

The IP Address Squeeze

- must keep adding levels of hierarchy to keep the routing working, which shrinks the existing fields
- number of items each field must identify keeps going up (hosts, subnets, providers, subscribers)
- each additional layer reduces total addressing capacity due to fragmentation of address space

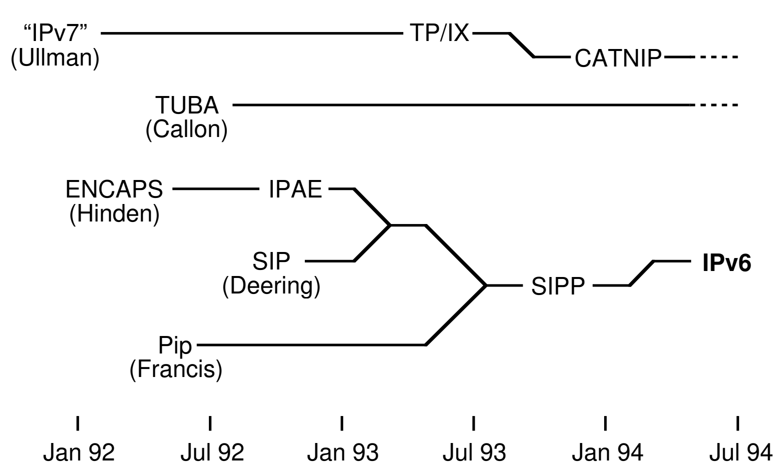
6

When Will the Addresses Run Out?

- hard to predict, because of lack of data — predictions vary between 20 years and yesterday
- not really the right question — there won't be a particular day when last address is consumed — they'll just get harder and harder to obtain
- anticipating this problem, IETF undertook 2.5 year effort to select a "next-generation IP" (IPng) with bigger, more hierarchical addresses…

7

IPng Candidates



8

The Address Size Debate

for fixed-length, 64-bit addresses:

- easily good for 10^{12} sites, 10^{15} nodes, at .0001 allocation efficiency (3 orders of mag. more than IPng requirement)
- minimizes growth of per-packet header overhead
- efficient for software processing

for variable-length addresses, up to 160 bits:

- compatible with OSI NSAP addressing plans
- big enough for autoconfiguration using IEEE 802 addresses
- could start with addresses shorter than 64 bits & grow later

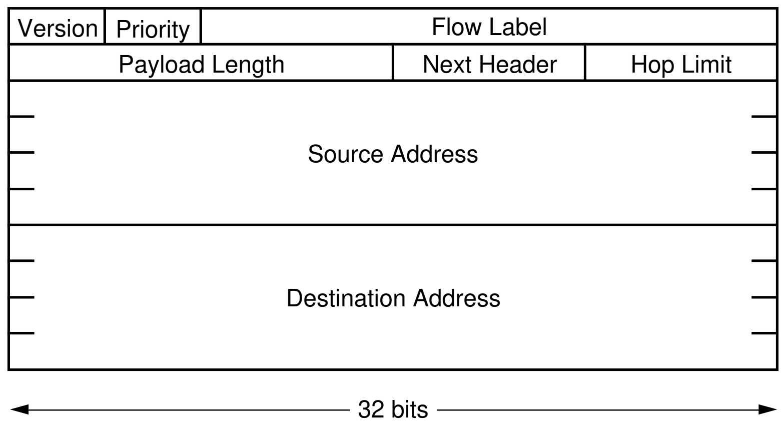
settled on fixed-length, 128-bit addresses

(340,282,366,920,938,463,463,374,607,431,768,211,456 in all!)

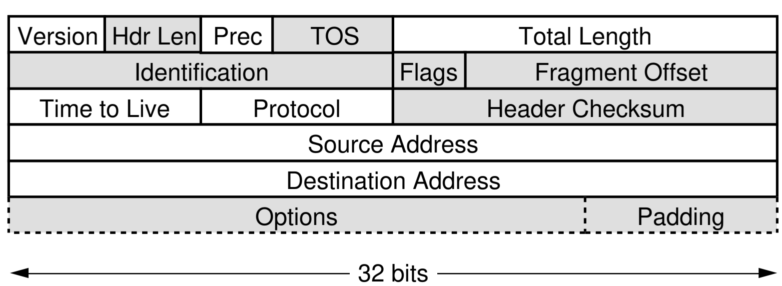
IPv6 Features

- expanded addressing capabilities
- streamlined header format
- improved support for options / extensions
- flow labeling capabilities
- support for mobility, autoconfiguration, and secure communication in all implementations
- incrementally deployable

The IPv6 Header



The IPv4 Header



shaded fields are absent from IPv6 header

Text Representation of Addresses

“preferred” form:	1080:0:FF:0:8:800:200C:417A
compressed form:	FF01:0:0:0:0:0:0:43 becomes FF01::43
IPv4-compatible:	0:0:0:0:0:0:13.1.68.3 or ::13.1.68.3

Address Types

- unicast (one-to-one)
  - global
  - link-local
  - site-local
  - compatible (IPv4, IPX, NSAP)
- multicast (one-to-many)
- anycast (one-to-nearest)
- reserved

Provider-Based Global Unicast Addresses

010	registry	provider	subscriber	subnet	interface

← 128 bits →

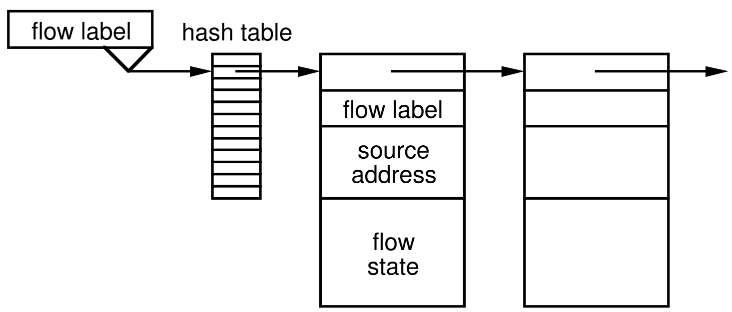
- all subfields variable-length, non-self-encoding (ref. CIDR)
- registry, provider, subscriber, and subnet fields may be further subdivided for more hierarchy
- alternatives for sites with multiple providers:
  - multiple addresses for all nodes
  - different parts of site use different prefixes
  - use one provider’s prefix, leaked into other providers’ tables
  - treat site as a provider

Flows

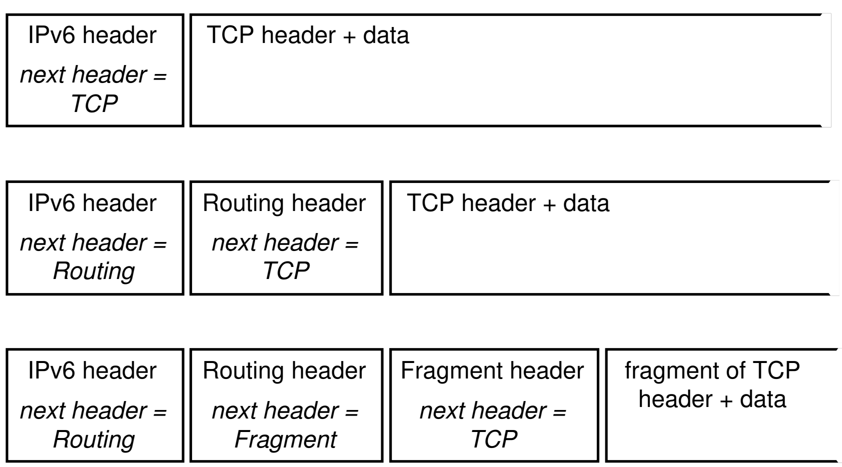
a sequence of packets for which the source desires non-default or “real-time” service

- service requested by explicit set-up, by in-packet options, or by other means
- packets in a flow identified by source address + flow label
- flow labels are random numbers, so any bits can serve as pre-computed hash…

Flow-State Lookup



Extension Headers



Extension Headers (cont.)

- processed only by node identified in IPv6 Destination Address field => much lower overhead than IPv4 options
  - –exception: Hop-by-Hop Options header
- eliminated IPv4's 40-octet limit on options
  - –in IPv6, limit is total packet size, or Path MTU in some cases
- currently defined extension headers:
  - –Hop-by-Hop Options, Routing, Fragment, Authentication, Encryption, Destination Options

Minimum MTU

- definitions:
  - **link MTU** — a link's maximum transmission unit, i.e., the max IP packet size that can be transmitted over the link
  - **path MTU** — the minimum MTU of all the links in a path between a source and a destination
- minimum link MTU for IPv6 is 576 octets (versus 68 octets for IPv4)
- on links with MTU < 576, link-specific fragmentation and reassembly must be used

Path MTU Discovery

implementations are expected to perform path MTU discovery to send packets bigger than 576 octets:

- for each dest., start by assuming MTU of first-hop link
- if a packet reaches a link in which it cannot fit, will invoke ICMP “packet too big” message to source, reporting the link’s MTU; MTU is cached by source for specific destination
- occasionally discard cached MTU to detect possible increase

minimal implementation can omit path MTU discovery as long as all packets kept ≤ 576 octets

e.g., in a boot ROM implementation

Fragment Header

Next Header	Reserved	Fragment Offset	0 0 M
Original Packet Identifier			

- though discouraged, can use IPv6 Fragment header to support upper layers that do not (yet) do path MTU discovery
- IPv6 frag. & reasm. is an **end-to-end** function; routers do **not** fragment packets en-route if too big—they send ICMP “packet too big” instead

Maximum Packet Size

- base IPv6 header supports payloads of up to 65,535 octets (not including 40 octet IPv6 header)
- bigger payloads can be carried by setting IPv6 Payload Length field to zero, and adding the “jumbogram” hop-by-hop option:

	Option Type=194	Opt Data Len=4
Payload Length		

- cannot use Fragment header with jumbograms

Routing

- uses same “longest-prefix match” routing as IPv4 CIDR
- straightforward changes to existing IPv4 routing protocols to handle bigger addresses
  - unicast: OSPF, RIP-II, IDRP, …
  - multicast: MOSPF, DVMRP, PIM, …
- can use Routing header with anycast addresses to route packets through particular regions
  - e.g., for provider selection, policy, performance, etc.

IPv6 Security

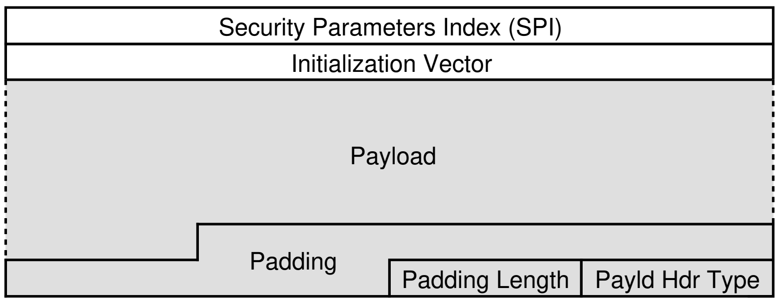
- all implementations expected to support authentication and encryption headers
- authentication separate from encryption for use in situations where encryption is prohibited or prohibitively expensive
- key distribution protocols are under development; independent of IP v4/v6
- support for manual key configuration required

Authentication Header

Next Header	Auth Data Len	Reserved
Security Parameters Index (SPI)		
Authentication Data		

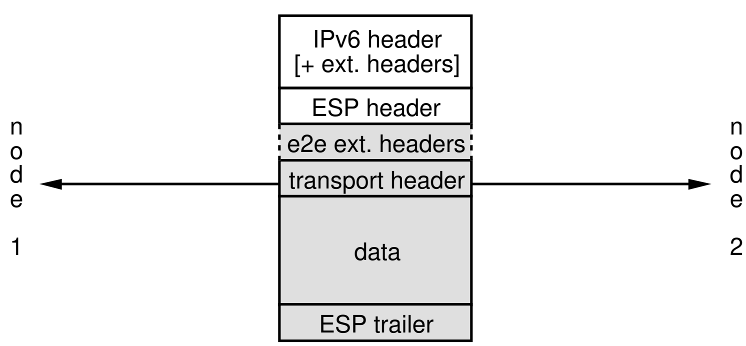
- Destination Address + SPI identifies security association state (key, lifetime, algorithm, etc.)
- provides authentication and data integrity for all fields of IPv6 packet that do not change en-route
- default algorithm is Keyed MD5

Encapsulating Security Payload (ESP), DES-CBC Variant



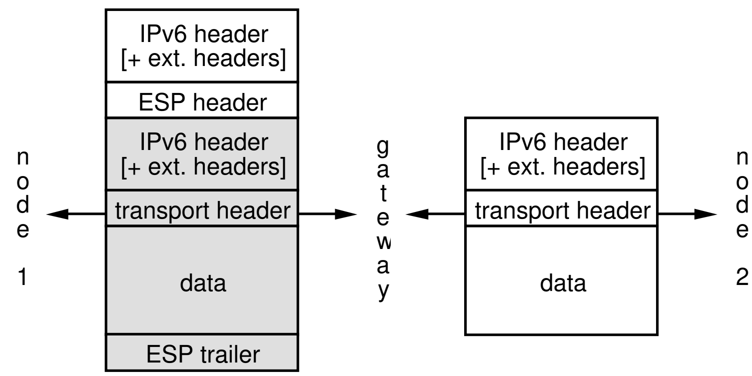
27

Transport Mode ESP (End-to-End)



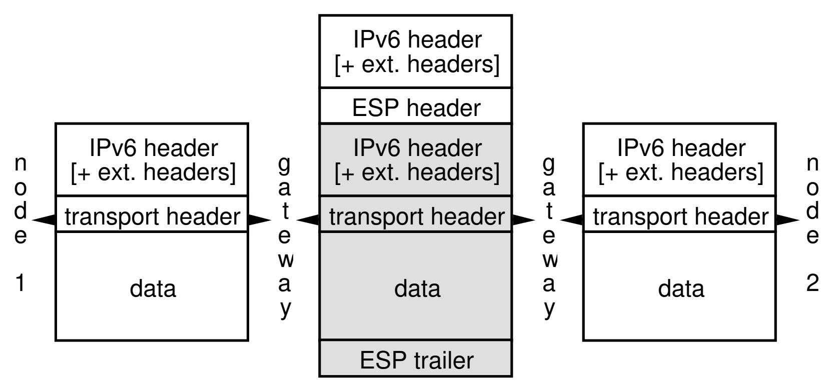
28

Tunnel Mode ESP
(End to Security Gateway)



29

Tunnel Mode ESP
(Security Gateway to Security Gateway)



30

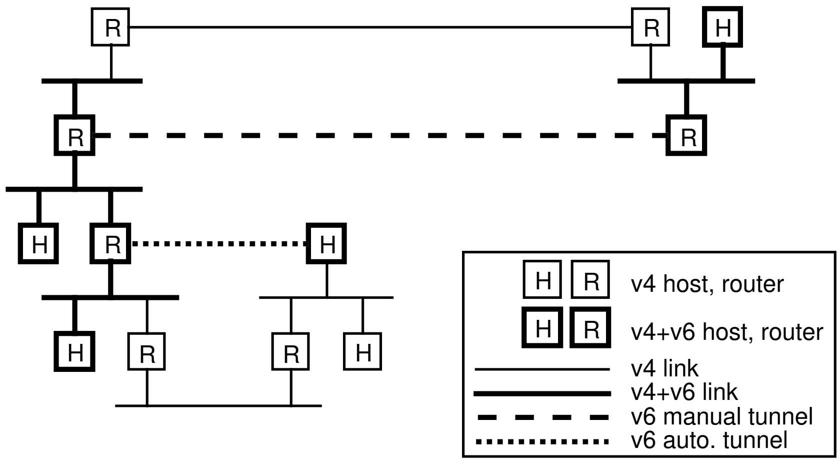
So Much for the Modularity of Layering!

- changes TCP/UDP checksum “pseudo-header”
- affects anything that reads/writes/stores/passes IP addresses (just about every higher protocol)
- packet lifetime no longer limited by IP layer (it never was, anyway!)
- bigger IP header must be taken into account when computing max payload sizes
- new DNS record type: AAAA
- …

Transition Approach

- gradually upgrade IPv4 hosts & routers to *also* support IPv6
- build overlay network by linking “islands” of IPv6 capability with “tunnels” (like the MBone)
  - tunnel by encapsulating IPv6 in IPv4
  - router to router tunnels are manually configured
  - isolated-host to router tunnels established automatically, using IPv4-compatible IPv6 addresses for router –> host, and IPv4 anycast for host –> router
- as routers are upgraded, tunnels are dismantled

Transition Components



33

Currently Published RFCs (Proposed Standard Status)

- RFC 1752 — IPng Area Directors' Recommendation
- RFC 1883 — IPv6 Protocol Specification
- RFC 1884 — IPv6 Addressing Architecture
- RFC 1885 — ICMP for IPv6
- RFC 1886 — DNS Extensions for IPv6
- RFC 1887 — IPv6 Unicast Address Allocation Arch.
- RFC 1897 — IPv6 Testing Address Allocation
- RFC 1933 — IPv6 Transition Mechanisms
- RFC 1825–1829 — IP Security Specifications

34

Currently Published RFCs (cont.) (Proposed Standard Status)

- RFC 1970 — IPv6 Neighbor Discovery
- RFC 1971 — IPv6 Stateless Autoconfiguration
- RFC 1972 — IPv6 over Ethernet
- RFC 1981 — Path MTU Discovery for IPv6
- RFC 1888 — OSI NSAPs and IPv6

Approved as Proposed Standards, But Not Yet Published as RFCs

- Provider-Based IPv6 Unicast Address Format
- IPv6 over FDDI
- IPv6 over PPP
- RIP-II for IPv6

Internet Drafts (to be submitted as RFCs)

- BSD API Extensions for IPv6
- IPv6 Mobility Support
- DHCP for IPv6
- DNS Update Protocol
- IPv6 Header Compression
- routing protocol extensions: OSPF, IDRP, PIM, …
- link-layer encapsulations: Token Ring, Frame Relay, X.25, ATM, AppleTalk, IPv6 (tunneling), …

Implementations Underway

router implementations:

- Bay
- Cisco
- Digital
- Ipsilon
- Penril
- Telebit (Denmark)

host implementations:

- Bull (BSD)
- Digital (UNIX & VMS)
- FTP Software (DOS & Windows)
- IBM (AIX)
- INRIA (BSD)
- Linux
- Mentat & Apple (MacOS)
- Microsoft (Windows '95)
- Novell
- NRL (BSD)
- Pacific Softworks (embedded systems)
- SICS & HP (HP-UX)
- Siemens Nixdorf
- Sun (Solaris)
- Pacific Softworks (embedded systems)
- WIDE: Keio Univ., NAIST, Hitachi, Sony, NTT (BSD)

Other Sources of Information

- web pages:
  - http://playground.sun.com/ipng/
- mailing lists:
  - ipng-request@sunroof.eng.sun.com — IPv6 working group
  - ngtrans-request@sunroof.eng.sun.com — transition wk. group
  - 6bone-request@isi.edu — 6Bone management
- books:
  - IPng, Internet Protocol Next Generation
    by Scott O. Bradner & Allison Mankin (Addison-Wesley)
  - IPv6, The New Internet Protocol
    by Christian Huitema (Prentice Hall)
  - IPng and the TCP/IP Protocols
    by Stephen Thomas (Wiley)