

IPv6 and Features

Department of Information Technology



Outline

- The IPv6 Header
- IPv6 Addresses
- Network Prefixes
- IPv6 Multicast
- IPv6 Extension Header
- IPv6 Host Address Assignment
- Epilog

Issue with IPv4, why IPv6?

- More than 32 bits are needed for addresses.
- The primary motive in developing the new version of IP known as IPv6 was the specter of running out of IPv4 addresses.
- Another important issue is that IPv4 requires (or used to require) a modest amount of effort at configuration; IPv6 was supposed to improve this.

History of IPv6

- By 1990 the IPv4 address-space issue was well understood, and the IETF was actively interested in proposals to replace IPv4. A working group for the so-called IP Next Generation (IPng), was created in 1993 to select the new version; [RFC 1550](#) was this group's formal solicitation of proposals. In July 1994 the IPng directors voted to accept a modified version of the Simple Internet Protocol Plus (SIPP) ([RFC 1710](#)), as the basis for IPv6. The first IPv6 specifications, released in 1995, were [RFC 1883](#) (now [RFC 2460](#), with updates) for the basic protocol, and [RFC 1884](#) (now [RFC 4291](#), again with updates) for the addressing architecture.

History of IPv6

- Simple Internet Protocol Plus (SIPP) addresses were originally 64 bits in length, but in the month leading up to adoption as the basis for IPv6 this was increased to 128. 64 bits would probably have been enough, but the problem is less the actual number than the simplicity with which addresses can be allocated; the more bits, the easier this becomes, as sites can be given relatively large address blocks without fear of waste. A secondary consideration in the 64-to-128 leap was the potential to accommodate now-obsolete ConnectionLess Network Protocol (CLNP) addresses ([1.15 IETF and OSI](#)), which were up to 160 bits in length, but compressible.

History of IPv6

- IPv6 has to some extent returned to the idea of a fixed division between network and host portions; for most IPv6 addresses, the first 64 bits is the network prefix (including any subnet portion) and the remaining 64 bits represents the host portion. The rule as spelled out in [RFC 2460](#), in 1998, was that the 64/64 split would apply to all addresses except those beginning with the bits 000; those addresses were then held in reserve in the unlikely event that the 64/64 split ran into problems in the future. This was a change from 1995, when [RFC 1884](#) envisioned 48-bit host portions and 80-bit prefixes.

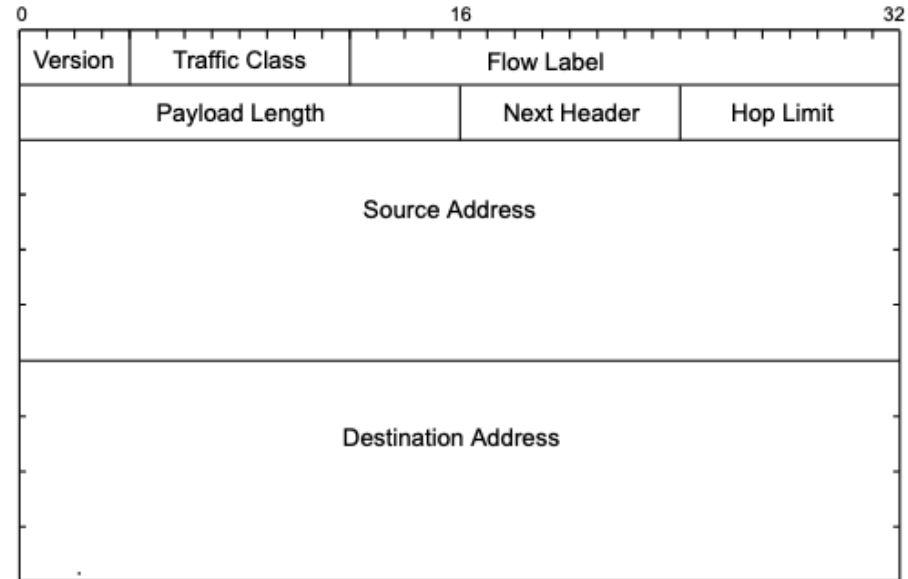
The IPv6 Header

- High-level routing, however, can, as in IPv4, be done on prefixes of any length (usually that means lengths shorter than /64). Routing can also be done on different prefix lengths at different points of the network.
- IPv6 is now twenty years old, and yet usage as of 2015 remains quite modest. However, the shortage in IPv4 addresses has begun to loom ominously; IPv6 adoption rates may rise quickly if IPv4 addresses begin to climb in price.

The IPv6 Header

The IPv6 **fixed header** is pictured below; at 40 bytes, it is twice the size of the IPv4 header.

The fixed header is intended to support only what *every* packet needs: there is no support for fragmentation, no header checksum, and no option fields.



The IPv6 Header

- However, the concept of **extension headers** has been introduced to support some of these as options; some IPv6 extension headers are described in [11.5 IPv6 Extension Headers](#). Whatever header comes next is identified by the Next Header field, much like the IPv4 Protocol field. Some other fixed-header fields have also been renamed from their IPv4 analogues: the IPv4 TTL is now the IPv6 Hop_Limit (still decremented by each router with the packet discarded when it reaches 0), and the IPv4 DS field has become the IPv6 Traffic Class.

The IPv6 Header

- The Flow Label is new. [RFC 2460](#) states that it may be used by a source to label sequences of packets for which it requests special handling by the IPv6 routers, such as non-default quality of service or “real-time” service.
- Senders not actually taking advantage of any quality-of-service options are supposed to set the Flow Label to zero.
- When used, the Flow Label represents a sender-computed hash of the source and destination addresses, and perhaps the traffic class. Routers can use this field as a way to look up quickly any priority or reservation state for the packet. All packets belonging to the same flow should have the same Routing Extension header, [11.5.3 Routing Header](#). The Flow Label will in general *not* include any information about the source and destination *port* numbers, except that only some of the connections between a pair of hosts may make use of this field.
- A **flow**, as the term is used here, is *one-way*; the return traffic belongs to a different flow. Historically, the term “flow” has also been used at various other scales: a single bidirectional TCP connection, multiple *related* TCP connections, or even all traffic from a particular subnet (eg the “computer-lab flow”).

IPv6 Addresses

- IPv6 addresses are written in eight groups of four hex digits (with lowercase a-f preferred over A-F) ([RFC 5952](#)). The groups are separated by colons, and have leading 0's removed, *e.g.*
- fedc:13:1654:310:fedc:bc37:61:3210
- If an address contains a long run of 0's – for example, if the IPv6 address had an embedded IPv4 address – then when writing the address, the string “::” should be used to represent however many blocks of 0000 as are needed to create an address of the correct length; to avoid ambiguity this can be used only once. Also, embedded IPv4 addresses may continue to use the “.” separator:
- ::ffff:147.126.65.141

- The prior example represents one standard IPv6 format for representing IPv4 addresses (see [12.4 Using IPv6 and IPv4 Together](#)). 48 bits are explicitly displayed; the :: means these are prefixed by 80 0-bits.
- The IPv6 loopback address is ::1 (that is, 127 0-bits followed by a 1-bit).
- Network address **prefixes** may be written with the “/” notation, as in IPv4:
- 12ab:0:0:cd30::/60

- [RFC 3513](#) suggested that initial IPv6 unicast-address allocation be initially limited to addresses beginning with the bits 001, that is, the 2000::/3 block (20 in binary is 0010 0000).
- Generally speaking, IPv6 addresses consist of a 64-bit network prefix (perhaps including subnet bits) followed by a 64-bit “interface identifier”. See [11.3 Network Prefixes](#) and [11.2.1 Interface identifiers](#).
- IPv6 addresses all have an associated **scope**, defined in [RFC 4007](#). The scope of a unicast address is either **global**, meaning it is intended to be globally routable, or **link-local**, meaning that it will only work with directly connected neighbors ([11.2.2 Link-local addresses](#)). The loopback address is considered to have link-local scope. A few more scope levels are available for multicast addresses, *eg* “site-local” ([RFC 4291](#)). The scope of an IPv6 address is implicitly coded within the first 64 bits; addresses in the 2000::/3 block above, for example, have global scope.
- Packets with local-scope addresses (*eg* link-local addresses) for either the destination or the source cannot be routed (the latter because a reply would be impossible).



Interface identifiers

- most IPv6 addresses can be divided into a 64-bit network prefix and a 64-bit “host” portion, the latter corresponding to the “host” bits of an IPv4 address. These host-portion bits are known officially as the **interface identifier**; the change in terminology reflects the understanding that all IP addresses attach to interfaces rather than to hosts.

Network Prefixes

- We have been assuming that an IPv6 address, at least as seen by a host, is composed of a 64-bit network prefix and a 64-bit interface identifier. As of 2015 this remains a requirement; [RFC 4291](#) (IPv6 Addressing Architecture) states:
- For all unicast addresses, except those that start with the binary value 000, Interface IDs are required to be 64 bits long....

Network Prefixes

- This /64 requirement is occasionally revisited by the IETF, but is unlikely to change for mainstream IPv6 traffic. This firm 64/64 split is a departure from IPv4, where the host/subnet division point has depended, since the development of subnets, on local configuration.
- Note that while the net/interface (net/host) division point is fixed, routers may still use CIDR ([14.1 Classless Internet Domain Routing: CIDR](#)) and may still base forwarding decisions on prefixes shorter than /64.
- As of 2015, all allocations for globally routable IPv6 prefixes are part of the 2000::/3 block.
- IPv6 also defines a variety of specialized network prefixes, including the link-local prefix and prefixes for anycast and multicast addresses. For example, as we saw earlier, the prefix ::ffff:0:0/96 identifies IPv6 addresses with embedded IPv4 addresses.

Network Prefixes

- The most important class of 64-bit network prefixes, however, are those supplied by a provider or other address-numbering entity, and which represent the first half of globally routable IPv6 addresses. These are the prefixes that will be visible to the outside world.
- IPv6 customers will typically be assigned a relatively large block of addresses, *eg* /48 or /56. The former allows $64 - 48 = 16$ bits for local “subnet” specification within a 64-bit network prefix; the latter allows 8 subnet bits. These subnet bits are – as in IPv4 – supplied through router configuration; see [12.3 IPv6 Subnets](#). The closest IPv6 analogue to the IPv4 subnet mask is that all network prefixes are supplied to hosts with an associated length, although that length will almost always be 64 bits.
- Many sites will have only a single externally visible address block. However, some sites may be multihomed and thus have multiple independent address blocks.

Network Prefixes

- The first 8 bits of a unique-local prefix are 1111 1101 (fd00::/8). The related prefix 1111 1100 (fc00::/8) is reserved for future use; the two together may be consolidated as fc00::/7. The last 16 bits of a 64-bit unique-local prefix represent the subnet ID, and are assigned either administratively or via autoconfiguration. The 40 bits in between, from bit 8 up to bit 48, represent the **Global ID**. A site is to set the Global ID to a pseudorandom value.

Network Prefixes

- The resultant unique-local prefix is “almost certainly” globally unique (and is considered to have **global scope** in the sense of [11.2 IPv6 Addresses](#)), although it is not supposed to be routed off a site. Furthermore, a site would generally not admit any packets from the outside world addressed to a destination with the Global ID as prefix. One rationale for choosing unique Global IDs for each site is to accommodate potential later mergers of organizations without the need for renumbering; this has been a chronic problem for sites using private IPv4 address blocks. Another justification is to accommodate VPN connections from other sites. For example, if I use IPv4 block 10.0.0.0/8 at home, and connect using VPN to a site also using 10.0.0.0/8, it is possible that my printer will have the same IPv4 address as their application server.

IPv6 Multicast

- IPv6 has moved away from LAN-layer *broadcast*, instead providing a wide range of LAN-layer *multicast* groups. (Note that LAN-layer multicast is often straightforward; it is general IP-layer multicast that is problematic. This switch to multicast is intended to limit broadcast traffic in general, though many switches still propagate LAN multicast traffic everywhere, like broadcast.

IPv6 Multicast

- An IPv6 multicast address is one beginning with the eight bits 1111 1111 (ff00::/8); numerous specific such addresses, and even classes of addresses, have been defined. For actual delivery, IPv6 multicast addresses correspond to LAN-layer (eg Ethernet) multicast addresses through a well-defined static correspondence; specifically, if x, y, z and w are the last four bytes of the IPv6 multicast address, in hex, then the corresponding Ethernet multicast address is 33:33:x:y:z:w ([RFC 2464](#)). A typical IPv6 host will need to join (that is, subscribe to) several Ethernet multicast groups.

IPv6 Multicast

- The IPv6 multicast address with the broadest scope is **all-nodes**, with address ff02::1; the corresponding Ethernet multicast address is 33:33:00:00:00:01. This essentially corresponds to IPv4's LAN broadcast, though the use of LAN multicast here means that non-IPv6 hosts should not see packets sent to this address. Another important IPv6 multicast address is ff02::2, the **all-routers** address. This is meant to be used to reach all routers, and routers only; ordinary hosts do not subscribe.

IPv6 Multicast

- IPv6 nodes on Ethernets send LAN-layer **Multicast Listener Discovery** (MLD) messages to multicast groups they wish to start using;
- these messages allow multicast-aware Ethernet switches to optimize forwarding so that only those hosts that have subscribed to the multicast group in question will receive the messages.
- Otherwise, switches are supposed to treat multicast like broadcast; worse, some switches may simply fail to forward multicast packets to destinations that have not explicitly opted to join the group.

IPv6 Extension Headers

- In IPv4, the IP header contained a Protocol field to identify the next header; usually UDP or TCP. All IPv4 options were contained in the IP header itself. IPv6 has replaced this with a scheme for allowing an arbitrary chain of supplemental IPv6 headers. The IPv6 Next Header field *can* indicate that the following header is UDP or TCP, but can also indicate one of several IPv6 options. These optional, or extension, headers include:
 - Hop-by-Hop options header
 - Destination options header
 - Routing header
 - Fragment header
 - Authentication header
 - Mobility header
 - Encapsulated Security Payload header
- These extension headers must be processed in order; the recommended order for inclusion is as above. Most of them are intended for processing only at the destination host; the hop-by-hop and routing headers are exceptions.



IPv6 Host Address Assignment

- IPv6 provides two competing ways for hosts to obtain their full IP addresses. One is **DHCPv6**, based on IPv4's DHCP ([10.3 Dynamic Host Configuration Protocol \(DHCP\)](#)), in which the entire address is handed out by a DHCPv6 server. The other is **StateLess Address AutoConfiguration**, or SLAAC, in which the interface-identifier part of the address is generated locally, and the network prefix is obtained via prefix discovery. The original idea behind SLAAC was to support complete plug-and-play network setup: hosts on an isolated LAN could talk to one another out of the box, and if a router was introduced connecting the LAN to the Internet, then hosts would be able to determine unique, routable addresses from information available from the router.



IPv6 Host Address Assignment

- In the early days of IPv6 development, in fact, DHCPv6 may have been intended only for address assignments to routers and servers, with SLAAC meant for “ordinary” hosts. In that era, it was still common for IPv4 addresses to be assigned “statically”, via per-host configuration files. [RFC 4862](#) states that SLAAC is to be used when “a site is not particularly concerned with the exact addresses hosts use, so long as they are unique and properly routable.”

IPv6 Host Address Assignment

- **Duplicate Address Detection**
- Whenever an IPv6 host obtains a unicast address – a link-local address, an address created via SLAAC, an address received via DHCPv6 or a manually configured address – it goes through a **duplicate-address detection** (DAD) process. The host sends one or more Neighbor Solicitation messages (that is, like an ARP query), as in [11.6 Neighbor Discovery](#), asking if any other host has this address. If anyone answers, then the address is a duplicate. As with IPv4 ACD ([10.2.1 ARP Finer Points](#)), but *not* as with the original IPv4 self-ARP, the source-IP-address field of this NS message is set to a special “unspecified” value; this allows other hosts to recognize it as a DAD query.

Epilog

- IPv4 has run out of large address blocks, as of 2011. IPv6 has reached a mature level of development. Most common operating systems provide excellent IPv6 support.
- Yet conversion has been slow. Many ISPs still provide limited (to nonexistent) support, and inexpensive IPv6 firewalls to replace the ubiquitous consumer-grade NAT routers are just beginning to appear. Time will tell how all this evolves. However, while IPv6 has now been around for twenty years, top-level IPv4 address blocks disappeared much more recently. It is quite possible that this will prove to be just the catalyst IPv6 needs.