

IPv6 addressing

[Main Page](#) > [FAQ](#) > [Networking](#) > **IPv6 addressing**



Contents

- [1 Introduction](#)
- [2 IPv4 vs IPv6 header](#)
- [3 Abbreviating IPv6 addresses](#)
- [4 Types of IPv6 addresses](#)
- [5 External links](#)

Introduction

Due to exhaustion of IPv4 addresses, it's becoming a common practice to configure routers, servers, standalone computers, IoT devices and many other appliances using IPv6 address pool. IP version 6 [[RFC8200](#)] defines the exact same features and functions as IPv4, but the way IP version 6 is implemented differs from IP version 4. While both IPv4 and IPv6 address pools are finite, IPv6 has an extremely large amount of available total addresses compared to IPv4. In theory, the total IPv4 address pool contains $2^{32}=4,294,967,296$. Some of these addresses are already reserved for internal or special use-cases.

In comparison, IPv6 has a pool of $2^{128}=340,282,366,920,938,463,463,374,607,431,768,211,456$ (more than 340 undecillion) available addresses. This amount of available addresses means that any device can have at least a single public IP address. This is especially relevant to the growing IoT industry as some of the devices require their own individual IP addresses to become reachable directly via the internet. However, while this IP address pool massive in theory, much like the IPv4, the **IPv6 addressing** has some notable rules and nuances that must be accounted for before implementing the newer IP standard.

IPv4 vs IPv6 header

Before diving into the details of how each IPv6 address space is classified and what it's used for, the inspection of IPv6 header from theoretical standpoint would assist in comparing and understanding the difference between IPv4 and IPv6.

Note: due to differences in headers, the IPv4 and IPv6 protocols cannot directly interoperate without utilizing supplementary methods such as dual-stack capable equipment, tunneling, 6to4, NAT64, Teredo, 6rd, ISATAP or other supported mechanisms.



Comparison between IPv4 and IPv6 headers

The biggest differences between the IPv4 and IPv6 headers are the byte size and simplified designed

of the header overall. IPv4 header varies in byte size - it begins from **20 bytes** ($5 \times 32 = 160$ bits) and can go up all the way to **60 bytes** ($15 \times 32 = 480$ bits). Due to variation in size, networking devices may require additional processing power for the when receiving and sending packets.

On the other hand, the size of main IPv6 header is constant - it will always remain at **40 bytes**. There is an extension header for IPv6, which is similar to the "Options" field in the IPv4 header. However, in IPv6 case, the extension header is implemented past the fixed header. Most extension headers will be processed only by the destination node which results in optimization and efficiency of overall packet processing due to devices in the middle of the packet path not having to process extra information. The only type of extension header that must be processed by every single network device in the path is the "Hop-by-Hop" options header, which must always be the first one after the main IPv6 header.

There are many other differences between the IPv4 and IPv6 headers which won't be discussed or mentioned here. To read more about the design of IPv6 header and how it works, please refer to the IPv6 protocol RFC [[RFC8200](#)].

Abbreviating IPv6 addresses

There are two basic rules that must be followed when abbreviating IPv6 addresses

1. Find the longest string of consecutive hexets (every 4 digits in the IPv6 address, separated by colon ":" sign) with a value of "0" and replace it with double colon "::". This is the biggest possible abbreviation in any given IPv6 address. **This abbreviation can only be used once per any single IPv6 address!**
2. In every other hexet (every 4 digits in the IPv6 address) it is allowed to remove leading 0s by simply leaving a **x:0:y** between the entire IP address This can be used as many times as needed in any IPv6 address.

For example, the following is a random public IPv6 address with a /64 mask in full form (expanded):

2001:0db8:3c4d:0015:0000:0000:0009:0144

In this example, first four hexets of this IP address will remain consistent at all times due to /64 mask - that means the beginning of IP address **2001:0db8:3c4d:0015** won't change. Reading a full IPv6 address, however, will take a very long time. To **shorten** it, we can use the second rule to eliminate unnecessary 0s. The following would be a **shortened** version of the given IPv6 address:

2001:0db8:3c4d:0015:0:0:0009:0144

In order to reduce the size of this IPv6 address further, we can apply first rule eliminate the longest part between any consecutive hexets with a value of "0":

2001:0db8:3c4d:0015::9:0144

At this point the biggest work has already been done and all that's left to be done is cleaning up all the leading in specific hexets. In this example, reading from left to right, hexet #2, #4 and #8 can be shortened further by removing the leading 0s.

2001:db8:3c4d:15::9:144

At this point the abbreviation part is done and we have a fully compressed IPv6 address. Please note

that hexet #1, "2001" cannot be abbreviated due to not having a leading 0 at the beginning of hexet.

Important note: removing 0s in the middle or at the end of an IPv6 hexet is not permitted according to standard and it will result in either different or completely invalid IPv6 address!

To expand the IPv6 back to full format use the exact same logic but reversed. First, fill in the partially full hexets without leading 0s. Then, find the double colon "::" and count the hexets that are currently visible. Once done, fill in the 0s until all eight hexets in total are back in the IPv6 address.

Needless to say, while it may be easy to do, it's still manual work. Understanding IPv6 addressing, abbreviating and formatting is useful, there are tools to both validate and expand or abbreviate any given IPv6 address automatically. The following online resources are recommended to use when working with IPv6 addressing:

- <https://www.ipaddressguide.com/ipv6> - a tool to expand or compress and verify an IPv6 address
- http://sqa.fyicenter.com/1000334_IPv6_Address_Validator.html - a tool to verify IPv6 address

Types of IPv6 addresses

This section will briefly cover types of preassigned IPv6 addresses. Similarly to IPv4, some of IPv6 address pool is reserved for specific services and use-cases. The table below will be useful when working with IPv6 addresses and assigning or simply trying to understand how each IPv6 address has been allocated or used in comparison to IPv4. [The original source of this table can be found and downloaded here.](#)

Note: There are no broadcast addresses in IPv6. Multicast addresses have superseded this function.

IPv6 Address Types

Prefix	Address type	IPv4 equivalent	Designation and explanation
2000::/3	Global Unicast	No equivalent single block	Other than the exceptions documented in this table, the operators of networks using these addresses can be found using the Whois servers of the RIRs listed in the registry at: https://www.iana.org/assignments/ipv6-unicast-address-assignments/ipv6-unicast-address-assignments.xhtml
fe80::/10	Link-Local Addresses	169.254.0.0/16 (RFC3927)	These addresses are used on a single link or a non-routed common access network, such as an Ethernet LAN. They do not need to be unique outside of that link. Link-local addresses may appear as the source or destination of an IPv6 packet. Routers must not forward IPv6 packets if the source or destination contains a linklocal address. Link-local addresses may appear as the source or destination of an IPv6 packet. Routers must not forward IPv6 packets if the source or destination contains a linklocal address.
ff00::/8	Multicast	224.0.0.0/4	These addresses are used to identify multicast groups. They should only be used as destination addresses, never as source addresses.
fc00::/7	Unique Local Addresses (ULAs)	Private, or RFC1918 address space: 10.0.0.0/8 172.16.0.0/12 192.168.0.0/16	These addresses are reserved for local use in home and enterprise environments and are not public address space. These addresses might not be unique, and there is no formal address registration. Packets with these addresses in the source or destination fields are not intended to be routed on the public Internet but are intended to be routed within the enterprise or organisation. See RFC4193 for more details.
::0	Routing	0.0.0.0/0	Default route (no specific route)
::128	Unspecified	0.0.0.0	This address may only be used as a source address by an initializing host before it has learned its own address. It must never be assigned to any node. It indicates the absence of an address.
::1/128	Loopback	127.0.0.1	This address is used when a host talks to itself over IPv6. This often happens when one program sends data to another.
::ffff/96	IPv4-Mapped	No equivalent.	These addresses are used to embed IPv4 addresses in an IPv6 address. One use for this is in a dual stack transition scenario where IPv4 addresses can be mapped into an IPv6 address. See RFC4038 for more details. Additionally, the mapped IPv4 address can be looked up in the relevant RIR's Whois database.

2001:0000::/32	Teredo	No equivalent	This is a mapped address allowing IPv6 tunneling through IPv4 NATs. The address is formed using the Teredo prefix, the server's unique IPv4 address, flags describing the type of NAT, the obfuscated client port and the client IPv4 address, which is probably a private address. It is possible to reverse the process and identify the IPv4 address of the relay server, which can then be looked up in the relevant RIR's Whois database. You can do this on the following webpage: https://www.potaroo.net/cgi-bin/ipv6addr
2001:0002::/48	Benchmarking	198.18.0.0/15	These addresses are reserved for use in documentation. They should not be used as source or destination addresses.
2001:0010::/28	Orchid	No equivalent	These addresses are used for a fixed-term experiment. They should only be visible on an end-to-end basis and routers should not see packets using them as source or destination addresses.
2002::/16	6to4	There is no equivalent but 192.88.99.0/24 has been reserved as the 6to4 relay anycast address prefix by the IETF.	A 6to4 gateway adds its IPv4 address to this 2002::/16, creating a unique /48 prefix. As the IPv4 address of the gateway router is used to compose the IPv6 prefix, it is possible to reverse the process and identify the IPv4 address, which can then be looked up in the relevant RIR's Whois database. You can do this on the following webpage: https://www.potaroo.net/cgi-bin/ipv6addr
2001:db8::/32	Documentation	192.0.2.0/24 198.51.100.0/24 203.0.113.0/24	These addresses are used in examples and documentation. They should never be source or destination addresses.

External links

- <https://datatracker.ietf.org/doc/html/rfc8200> - official IPv6 RFC
- <https://www.ipaddressguide.com/ipv6> - compress, expand and verify IPv6 addresses
- http://sqa.fyicenter.com/1000334_IPv6_Address_Validator.html - IPv6 address verifier
- https://www.ripe.net/participate/member-support/lir-basics/ipv6_reference_card.pdf - RIPE.net IPv6 addressing types sheet
- <https://datatracker.ietf.org/doc/html/rfc4038> - Application aspects of IPv6 transition
- <https://datatracker.ietf.org/doc/html/rfc1918> - Address allocation for private networks
- <https://datatracker.ietf.org/doc/html/rfc4193> - Unique Local IPv6 Unicast Addresses (ULA)
- <https://datatracker.ietf.org/doc/html/rfc3927> - Dynamic Configuration of IPv4 Link-Local Addresses
- <https://www.potaroo.net/cgi-bin/ipv6addr> - Generate prefix report the provided address
- <https://www.iana.org/assignments/ipv6-unicast-address-assignments/ipv6-unicast-address-assignments.xhtml> - IPv6 Global Unicast Address Assignments