Network Working Group   G. Bauer
Internet-Draft   23 April 2026
Intended status: Informational
Expires: 25 October 2026

Dotted Decimal notation for IPv6 addresses
draft-bauer-ipv6-dotted-decimal-address-format-00

Abstract

This document defines a new canonical format for IPv6 addresses, that uses familiar dotted decimal notation.

Status of This Memo

Copyright Notice

Table of Contents

1. Introduction

The global use of the Internet Protocol v6 (IPv6) is constantly growing. At the time of writing of this document, Google reports that 45% of it's customer traffic uses IPv6 [Google-IPv6]. Other actors like Cloudflare report a mean IPv6 traffic share of over 40% in the last 12 moths [Cloudflare-IPv6]. With the slow but steady grows in IPv6 adoption, network administrator are expected to be confronted more often with IPv6 addresses, represented in their canonical form [RFC5952]. While ensuring efficient text representation, the canonical IPv6 address form requires a deep knowledge of hexadecimal notation to be understood. This is often considered a barrier to the understanding ov the IPv6 protocol by network administrators who are more familiar with IPv4. A secondary IPv6 address notation could help close the gap between network administrators and the IPv6 protocol by providing a more familiar way of representing addresses. This proposal DOES NOT replace the standard canonical notation, but rather propose an alternate format.

2. Conventions and Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

3. Address representation

The way of textually representing IPv6 addresses is addressed in this section.

3.1. Current form

When being displayed or printed, an IPv6 address is represented in conformance with [RFC4291] and [RFC5952]. An example of an address with its prefix length is shown below.

```
2001:db8::bad/64
```

This representation uses hexadecimal notation in conjunction with the leading zeroes omission and zero compression rules. the prefix length is separated from the address by the '/' character.

3.2. Issues with current form

The canonical notation that is currently used for IPv6 address representation poses a number of issues :

* The use of hexadecimal notation introduces new digits that expand the traditional ten digits (0-9). However, as the standard english alphabet doesn't have more glyphs to represent digits, we are forced to use letters to represent values from 10 to 15. This introduces a lot of confusion when trying to read a hexadecimal number and forces network administrators to learn a new numbering format.

* The zero compression rule introduces a lot of confusion, because the length of an IPv6 represented in standard notation can vary greatly. Some examples of IPv6 addresses are shown below.

  ```
  2001:db8::1
  2001:db8:1111:1111:1111:1111
  ::
  ```

* The colon (':') character is normally used to separate an IP address and a port number. Its usage in the representation of an IPv6 address introduces event more complexity and the necessity to use more characters to separate the address from the port number. An example is shown below.

  ```
  [2001:db8::1]:443
  ```

These issues have greatly slowed down the global adoption of IPv6. Some proposals have been made to mitigate the issues, like [IPv8], but redefining the architecture of the entire Internet seems excessive just to resolve an address representation issue.

3.3. New representation

The new IPv6 address representation aims to fix the issues listed in Section 3.2 and improve the readability of IPv6 addresses for seasoned network administrators. With this new representation, an IPv6 address is divided into 16 groups of 8 bits. Each byte is represented in decimal notation. The individual bytes are separated by dots ('.'). Leading zeroes in a byte MUST be omitted, but there is no zero compression rule. An example of an IPv6 address is shown below.

Classic IPv6 address representation	New IPv6 address representation
2001:db8::bad	32.1.13.184.0.0.0.0.0.0.0.0.0.0.11.173

Table 1: Dotted decimal format

While being longer, the new representation is more similar to an IPv4 address. The new representation also reintroduces network masks using the same semantics. An IPv6 prefix length with a value of _N_ can be converted to a network mask by setting the _N_ leftmost bits of a 128 bits integer to one and the reste of the bits to zero. The mask can them be displayed using the same new semantics as for an IPv6 address.

IPv6 prefix length	Resulting IPv6 netmask in dotted decimal notation
0	0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0
32	255.255.255.255.0.0.0.0.0.0.0.0.0.0.0.0
64	255.255.255.255.255.255.255.255.0.0.0.0.0.0.0.0.

Table 2: IPv6 network mask notation

3.4. Examples of fixed representation

The IPv6 specification contains a number of special use addresses whose representation are very confusion. A few examples and their dotted decimal representation are listed below.

Address	Standard IPv6 representation	Dotted decimal representation
Unspecified	::	0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0
Loopback	::1	0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.1
IPv4-Mapped	::ffff:0:0	0.0.0.0.0.0.0.0.255.255.255.255.0.0.0.0

Table 3: Special use addresses

Note that in the case of the IPv4-Mapped address, the original IPv4 address is always readable in its native format.

4. Implementation considerations

Most programming languages include in their standard library a way to parse a string containing an IPv6 address in canonical representation into an actual binary address and vice-versa. As there has only been one standard format to represent an IPv4 address and an IPv6 address, these APIs do not include functionality to choose a format. Section 4.1 shows an example for adapting the C standard library to the new dotted decimal format.

4.1. C standard library

The C standard library provides a few functions to convert an IP address from its textual representation to a binary value and vice-versa. One of them is inet_pton, that can be used with IPv4 or IPv6 addresses. Its counterpart inet_ntop does the opposite.

```
int inet_pton(int af,
                const char *restrict src,
                void *restrict dst);

const char *inet_ntop(socklen_t size,
                        int af,
                        const void *restrict src,
                        char dst[restrict size],
                        socklen_t size);
```

The af argument represents the address family and can be set to AF_INETfor an IPv4 address and AF_INET6 for an IPv6 address.

One way of implementing the new dotted decimal notation would be to reserve a new value for the af argument, like AF_INET6_DD. This constant would have a decimal value of _11_, which follows the value of the constant AF_INET6 (_10_) defined in linux/socket.h.

5. Security Considerations

IPv6 address representation does not have a direct impact on Internet infrastructure security.

6. IANA Considerations

This document has no IANA actions.

7. References

7.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/rfc/rfc2119>.

[RFC4291] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 4291, DOI 10.17487/RFC4291, February 2006, <https://www.rfc-editor.org/rfc/rfc4291>.

[RFC5952] Kawamura, S. and M. Kawashima, "A Recommendation for IPv6 Address Text Representation", RFC 5952, DOI 10.17487/RFC5952, August 2010, <https://www.rfc-editor.org/rfc/rfc5952>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/rfc/rfc8174>.

7.2. Informative References

[Cloudflare-IPv6] Cloudflare, "Cloudflare Radar", n.d., <https://radar.cloudflare.com/adoption-and-usage?dateRange=52w>.

[Google-IPv6]
Google, "Google IPv6 Statistics", n.d., <https://www.google.com/intl/en/ipv6/statistics.html#tab=ipv6-adoption>.

[IPv8] Thain, J., "IPv8", n.d., <https://www.ietf.org/archive/id/draft-thain-ipv8-00.html>.

Author’s Address

Guillaume Bauer
Email: guillaume.bauer.418@gmail.com