IP Version 6 Addressing Architecture
draft-ietf-6man-rfc4291bis-07

Abstract

   This specification defines the addressing architecture of the IP
   Version 6 (IPv6) protocol. The document includes the IPv6 addressing
   model, text representations of IPv6 addresses, definition of IPv6
   unicast addresses, anycast addresses, and multicast addresses, and an
   IPv6 node's required addresses.

   This document obsoletes RFC 4291, "IP Version 6 Addressing
   Architecture".

Status of This Memo

   This Internet-Draft is submitted in full conformance with the
   provisions of BCP 78 and BCP 79.

   Internet-Drafts are working documents of the Internet Engineering
   Task Force (IETF). Note that other groups may also distribute
   working documents as Internet-Drafts. The list of current Internet-
   Drafts is at http://datatracker.ietf.org/drafts/current/.

   Internet-Drafts are draft documents valid for a maximum of six months
   and may be updated, replaced, or obsoleted by other documents at any
   time. It is inappropriate to use Internet-Drafts as reference
   material or to cite them other than as "work in progress."

   This Internet-Draft will expire on August 4, 2017.

Table of Contents

1. Introduction

   This specification defines the addressing architecture of the IP
   Version 6 protocol. It includes the basic formats for the various
   types of IPv6 addresses (unicast, anycast, and multicast).

2. IPv6 Addressing

   IPv6 addresses are 128-bit identifiers for interfaces and sets of
   interfaces (where "interface" is as defined in Section 2 of
   [I-D.ietf-6man-rfc2460bis]). There are three types of addresses:

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

      Anycast: An identifier for a set of interfaces (typically
                   belonging to different nodes). A packet sent to an
                   anycast address is delivered to one of the interfaces
                   identified by that address (the "nearest" one,
                   according to the routing protocols' measure of
                   distance).

      Multicast: An identifier for a set of interfaces (typically
                   belonging to different nodes). A packet sent to a
                   multicast address is delivered to all interfaces
                   identified by that address.

   There are no broadcast addresses in IPv6, their function being
   superseded by multicast addresses.

   In this document, fields in addresses are given a specific name, for
   example, "subnet". When this name is used with the term "ID" for
   identifier after the name (e.g., "subnet ID"), it refers to the
   contents of the named field. When it is used with the term "prefix"
   (e.g., "subnet prefix"), it refers to all of the address from the
   left up to and including this field.

   In IPv6, all zeros and all ones are legal values for any field,
   unless specifically excluded. Specifically, prefixes may contain, or
   end with, zero-valued fields.

2.1. **Addressing Model**

   IPv6 addresses of all types are assigned to interfaces, not nodes.
   An IPv6 unicast address refers to a single interface. Since each
   interface belongs to a single node, any of that node's interfaces'
   unicast addresses may be used as an identifier for the node.

   All interfaces are required to have at least one Link-Local unicast
   address (see Section 2.7 for additional required addresses). A
   single interface may also have multiple IPv6 addresses of any type
   (unicast, anycast, and multicast) or scope. Unicast addresses with a
   scope greater than link-scope are not needed for interfaces that are
   not used as the origin or destination of any IPv6 packets to or from
   non-neighbors. This is sometimes convenient for point-to-point
   interfaces. There is one exception to this addressing model:

      A unicast address or a set of unicast addresses may be assigned to
      multiple physical interfaces if the implementation treats the
      multiple physical interfaces as one interface when presenting it
      to the internet layer. This is useful for load-sharing over
      multiple physical interfaces.

   Currently, IPv6 continues the IPv4 model in that a subnet prefix is
   associated with one link. Multiple subnet prefixes may be assigned
   to the same link.

2.2. **Text Representation of IPv6 Addresses**

2.2.1. **Text Representation of Addresses**

   There are three conventional forms for representing IPv6 addresses as
   text strings:

   1. The preferred form is x:x:x:x:x:x:x:x, where the 'x's are one to
      four hexadecimal digits of the eight 16-bit pieces of the address.
      Examples:

         abcd:ef01:2345:6789:abcd:ef01:2345:6789
         2001:db8:0:0:8:800:200c:417a

      Note that it is not necessary to write the leading zeros in an
      individual field, but there must be at least one numeral in every
      field (except for the case described in 2.).

   2. Due to some methods of allocating certain styles of IPv6
      addresses, it will be common for addresses to contain long strings
      of zero bits. In order to make writing addresses containing zero

bits easier, a special syntax is available to compress the zeros.
The use of "::" indicates one or more groups of 16 bits of zeros.
The "::" can only appear once in an address. The "::" can also be
used to compress leading or trailing zeros in an address.

For example, the following addresses

```
2001:db8:0:0:8:800:200c:417a  a unicast address
ff01:0:0:0:0:0:0:101          a multicast address
0:0:0:0:0:0:0:1               the loopback address
0:0:0:0:0:0:0:0               the unspecified address
```

may be represented as

```
2001:db8::8:800:200c:417a     a unicast address
ff01::101                     a multicast address
::1                           the loopback address
::                            the unspecified address
```

3. An alternative form that is sometimes more convenient when dealing
   with a mixed environment of IPv4 and IPv6 nodes is
   x:x:x:x:x:x:d.d.d.d, where the 'x's are the hexadecimal values of
   the six high-order 16-bit pieces of the address, and the 'd's are
   the decimal values of the four low-order 8-bit pieces of the
   address (standard IPv4 representation). Examples:

```
0:0:0:0:0:0:13.1.68.3
0:0:0:0:0:ffff:129.144.52.38
```

or in compressed form:

```
::13.1.68.3
::ffff:129.144.52.38
```

2.2.2. Text Representation of Address Prefixes

The text representation of IPv6 address prefixes is similar to the
way IPv4 address prefixes are written in Classless Inter-Domain

Routing (CIDR) notation [RFC4632]. An IPv6 address prefix is
represented by the notation:

    ipv6-address/prefix-length

where

ipv6-address is an IPv6 address in any of the notations listed in
              Section 2.2.

prefix-length is a decimal value specifying how many of the leftmost
              contiguous bits of the address comprise the prefix.

For example, the following are legal representations of the 60-bit
prefix 20010db80000cd3 (hexadecimal):

    2001:0db8:0000:cd30:0000:0000:0000:0000/60

    2001:0db8::cd30:0:0:0:0/60

    2001:0db8:0:cd30::/60

The following are NOT legal representations of the above prefix:

    2001:0db8:0:cd3/60   may drop leading zeros, but not trailing
                          zeros, within any 16-bit chunk of the address

    2001:0db8::cd30/60   address to left of "/" expands to
                          2001:0db8:0000:0000:0000:0000:0000:cd30

    2001:0db8::cd3/60   address to left of "/" expands to
                          2001:0db8:0000:0000:0000:0000:0000:0cd3

When writing both a node address and a prefix of that node address
(e.g., the node's subnet prefix), the two can be combined as follows:

    the node address   2001:0db8:0:cd30:123:4567:89ab:cdef
    and its subnet number   2001:0db8:0:cd30::/60

    can be abbreviated as   2001:0db8:0:cd30:123:4567:89ab:cdef/60

2.2.3. **Recommendation for outputting IPv6 addresses**

   This section provides a recommendation for systems generating and
   outputting IPv6 addresses as text. Note, all implementations must
   accept and process all addresses in the formats defined in the
   previous two sections of this document. Background on this
   recommendation can be found in [RFC5952].

   The recommendations are as follows:

   1. The hexadecimal digits "a", "b", "c", "d", "e", and "f" in an IPv6
      address must be represented in lowercase.

   2. Leading zeros in a 16-Bit Field must be suppressed. For example,

         2001:0db8::0001

      is not correct and must be represented as

         2001:db8::1

   3. A single 16-bit 0000 field must be represented as 0.

      The use of the symbol "::" must be used to its maximum capability.
      For example:

         2001:db8:0:0:0:0:2:1

      must be shortened to

         2001:db8::2:1

      Likewise,

         2001:db8::0:1

      is not correct, because the symbol "::" could have been used to
      produce a shorter representation

     2001:db8::1.

  4. When there is an alternative choice in the placement of a "::",
     the longest run of consecutive 16-bit 0 fields must be shortened,
     that is, in

        2001:0:0:1:0:0:0:1

     the sequence with three consecutive zero fields is shortened to

        2001:0:0:1::1

  5. When the length of the consecutive 16-bit 0 fields are equal, for
     example

        2001:db8:0:0:1:0:0:1

     the first sequence of zero bits must be shortened. For example

        2001:db8::1:0:0:1

     is the correct representation.

  6. The symbol "::" must not be used to shorten just one 16-bit 0
     field. For example, the representation

        2001:db8:0:1:1:1:1:1

     is correct, but

        2001:db8::1:1:1:1:1

     is not correct.

7. The text representation method describe in this section should
    also be use for text Representation of IPv6 Address Prefixes. For
    example

        2001:0db8:0000:cd30:0000:0000:0000:0000/60

    should be shown as

        2001:0db8:0:cd30::/60

8. The text representation method describe in this section should be
    applied for IPv6 addresses with embedded IPv4 address. For
    example

        0:0:0:0:0:ffff:192.0.2.1

    should be shown as

        ::ffff:192.0.2.1

[2.3](). Address Type Identification

The type of an IPv6 address is identified by the high-order bits of
the address, as follows:

```
Address type            Binary prefix        IPv6 notation    Section
------------            -------------        -------------    -------
Unspecified             00...0  (128 bits)   ::/128           2.4.2
Loopback                00...1  (128 bits)   ::1/128          2.4.3
Multicast               11111111             ff00::/8         2.6
Link-Local unicast      1111111010           fe80::/10        2.4.6
Global Unicast          (everything else)
```

Anycast addresses are taken from the unicast address spaces (of any
scope) and are not syntactically distinguishable from unicast
addresses.

The general format of Global Unicast addresses is described in
[Section 2.4.4](). Some special-purpose subtypes of Global Unicast

addresses that contain embedded IPv4 addresses (for the purposes of
IPv4-IPv6 interoperation) are described in Section 2.4.5.

Future specifications may redefine one or more sub-ranges of the
Global Unicast space for other purposes, but unless and until that
happens, implementations must treat all addresses that do not start
with any of the above-listed prefixes as Global Unicast addresses.

The current assigned IPv6 prefixes and references to their usage can
be found in the IANA Internet Protocol Version 6 Address Space
registry [IANA-AD] and the IANA IPv6 Special-Purpose Address Registry
[IANA-SP].

2.4. Unicast Addresses

IPv6 unicast addresses are aggregatable with prefixes of arbitrary
bit-length, similar to IPv4 addresses under Classless Inter-Domain
Routing.

IPv6 unicast routing is based on prefixes of any valid length up to
128 [BCP198]. For example, [RFC6164] standardises 127 bit prefixes
on inter-router point-to-point links. However, the Interface ID of
all unicast addresses, except those that start with the binary value
000, is required to be 64 bits long. The rationale for the 64 bit
boundary in IPv6 addresses can be found in [RFC7421]

There are several types of unicast addresses in IPv6, in particular,
Global Unicast, Local unicast, and Link-Local unicast. There are
also some special-purpose subtypes of Global Unicast, such as IPv6
addresses with embedded IPv4 addresses. Additional address types or
subtypes can be defined in the future.

IPv6 nodes may have considerable or little knowledge of the internal
structure of the IPv6 address, depending on the role the node plays
(for instance, host versus router). At a minimum, a node may
consider that unicast addresses (including its own) have no internal
structure:

```
|                          128 bits                              |
+---------------------------------------------------------------+
|                        node address                           |
+---------------------------------------------------------------+
```

A slightly sophisticated host (but still rather simple) may
additionally be aware of subnet prefix(es) for the link(s) it is
attached to, where different addresses may have different values for
n:

```
|           n bits              |            128-n bits           |
+-------------------------------+---------------------------------+
|          subnet prefix        |            interface ID         |
+-------------------------------+---------------------------------+
```
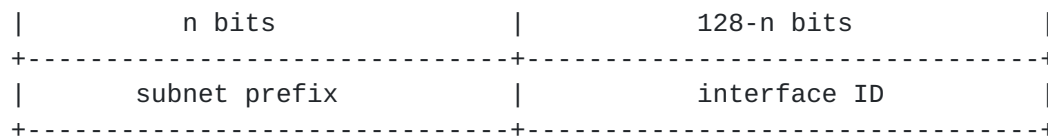
Though a very simple router may have no knowledge of the internal
structure of IPv6 unicast addresses, routers will more generally have
knowledge of one or more of the hierarchical boundaries for the
operation of routing protocols. The known boundaries will differ
from router to router, depending on what positions the router holds
in the routing hierarchy.

Except for the knowledge of the subnet boundary discussed in the
previous paragraphs, nodes should not make any assumptions about the
structure of an IPv6 address.

2.4.1. Interface Identifiers

Interface identifiers in IPv6 unicast addresses are used to identify
interfaces on a link. They are required to be unique within a subnet
prefix. It is recommended that the same interface identifier not be
assigned to different nodes on a link. They may also be unique over
a broader scope. The same interface identifier may be used on
multiple interfaces on a single node, as long as they are attached to
different subnets.

Interface IDs must be viewed outside of the node that created
Interface ID as an opaque bit string without any internal structure.

Note that the uniqueness of interface identifiers is independent of
the uniqueness of IPv6 addresses. For example, a Global Unicast
address may be created with an interface identifier that is only
unique on a single subnet, and a Link-Local address may be created
with interface identifier that is unique over multiple subnets.

As noted in Section 2.4, all unicast addresses, except those that
start with the binary value 000, Interface IDs are required to be 64
bits long.

The details of forming interface identifiers are defined in other
specifications, such as "Privacy Extensions for Stateless Address
Autoconfiguration in IPv6" [RFC4941] or "A Method for Generating
Semantically Opaque Interface Identifiers with IPv6 Stateless Address
Autoconfiguration (SLAAC)"[RFC7217]. Specific cases are described in
appropriate "IPv6 over <link>" specifications, such as "IPv6 over
Ethernet" [RFC2464] and "Transmission of IPv6 Packets over ITU-T
G.9959 Networks" [RFC7428]. The security and privacy considerations
for IPv6 address generation is described in [RFC7721].

Earlier versions of this document described a method of forming
interface identifiers derived from IEEE MAC-layer addresses call
Modified EUI-64 format. These are described in Appendix A and are no
longer recommended.

2.4.2. The Unspecified Address

The address 0:0:0:0:0:0:0:0 is called the unspecified address. It
must never be assigned to any node. It indicates the absence of an
address. One example of its use is in the Source Address field of
any IPv6 packets sent by an initializing host before it has learned
its own address.

The unspecified address must not be used as the destination address
of IPv6 packets or in IPv6 Routing headers. An IPv6 packet with a
source address of unspecified must never be forwarded by an IPv6
router.

2.4.3. The Loopback Address

The unicast address 0:0:0:0:0:0:0:1 is called the loopback address.
It may be used by a node to send an IPv6 packet to itself. It must
not be assigned to any physical interface. It is treated as having
Link-Local scope, and may be thought of as the Link-Local unicast
address of a virtual interface (typically called the "loopback
interface") to an imaginary link that goes nowhere.

The loopback address must not be used as the source address in IPv6
packets that are sent outside of a single node. An IPv6 packet with
a destination address of loopback must never be sent outside of a
single node and must never be forwarded by an IPv6 router. A packet
received on an interface with a destination address of loopback must
be dropped.

2.4.4. Global Unicast Addresses

The general format for IPv6 Global Unicast addresses is as follows:

```
|         n bits         |   m bits  |       128-n-m bits         |
+------------------------+-----------+----------------------------+
| global routing prefix  | subnet ID |       interface ID         |
+------------------------+-----------+----------------------------+
```

where the global routing prefix is a (typically hierarchically-
structured) value assigned to a site (a cluster of subnets/links),
the subnet ID is an identifier of a link within the site, and the
interface ID is as defined in Section 2.4.1.

As noted in [Section 2.4](#), all Global Unicast addresses other than
those that start with binary 000 have a 64-bit interface ID field
(i.e., n + m = 64), formatted as described in [Section 2.4.1](#). Global
Unicast addresses that start with binary 000 have no such constraint
on the size or structure of the interface ID field.

Examples of Global Unicast addresses that start with binary 000 are
the IPv6 address with embedded IPv4 addresses described in
[Section 2.4.5](#). An example of global addresses starting with a binary
value other than 000 (and therefore having a 64-bit interface ID
field) can be found in [[RFC3587](#)].

[2.4.5](#). IPv6 Addresses with Embedded IPv4 Addresses

Two types of IPv6 addresses are defined that carry an IPv4 address in
the low-order 32 bits of the address. These are the "IPv4-Compatible
IPv6 address" and the "IPv4-mapped IPv6 address".

[2.4.5.1](#). IPv4-Compatible IPv6 Address

The "IPv4-Compatible IPv6 address" was defined to assist in the IPv6
transition. The format of the "IPv4-Compatible IPv6 address" is as
follows:

```
|                80 bits               | 16 |      32 bits        |
+--------------------------------------+--------------------------+
|0000..............................0000|0000|    IPv4 address     |
+--------------------------------------+----+---------------------+
```

Note: The IPv4 address used in the "IPv4-Compatible IPv6 address"
must be a globally-unique IPv4 unicast address.

The "IPv4-Compatible IPv6 address" is now deprecated because the
current IPv6 transition mechanisms no longer use these addresses.
New or updated implementations are not required to support this
address type.

[2.4.5.2](#). IPv4-Mapped IPv6 Address

A second type of IPv6 address that holds an embedded IPv4 address is
defined. This address type is used to represent the addresses of
IPv4 nodes as IPv6 addresses. The format of the "IPv4-mapped IPv6
address" is as follows:

```
|                80 bits               | 16 |      32 bits        |
+--------------------------------------+--------------------------+
|0000..............................0000|ffff|    IPv4 address     |
+--------------------------------------+----+---------------------+
```

See [RFC4038] for background on the usage of the "IPv4-mapped IPv6 address".

2.4.6. Link-Local IPv6 Unicast Addresses

Link-Local addresses are for use on a single link. Link-Local addresses have the following format:

```
|   10     |
|  bits    |            54 bits            |            64 bits          |
+----------+------------------------------+----------------------------+
|1111111010|               0              |          interface ID      |
+----------+------------------------------+----------------------------+
```

Link-Local addresses are designed to be used for addressing on a single link for purposes such as automatic 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.

2.4.7. Other Local Unicast IPv6 Addresses

Unique Local Addresses (ULA) [RFC4193], the current form of Local IPv6 Addresses, are intended to be used for local communications, have global unicast scope, and are not expected to be routable on the global Internet.

Site-Local addresses, deprecated by [RFC3879], the previous form of Local IPv6 Addresses, were originally designed to be used for addressing inside of a site without the need for a global prefix.

The special behavior of Site-Local defined in [RFC3513] must no longer be supported in new implementations (i.e., new implementations must treat this prefix as Global Unicast). Existing implementations and deployments may continue to use this prefix.

2.5. Anycast Addresses

An IPv6 anycast address is an address that is assigned to more than one interface (typically belonging to different nodes), with the property that a packet sent to an anycast address is routed to the "nearest" interface having that address, according to the routing protocols' measure of distance.

Anycast addresses are allocated from the unicast address space, using any of the defined unicast address formats. Thus, anycast addresses are syntactically indistinguishable from unicast addresses. When a

unicast address is assigned to more than one interface, thus turning
it into an anycast address, the nodes to which the address is
assigned must be explicitly configured to know that it is an anycast
address.

For any assigned anycast address, there is a longest prefix P of that
address that identifies the topological region in which all
interfaces belonging to that anycast address reside. Within the
region identified by P, the anycast address must be maintained as a
separate entry in the routing system (commonly referred to as a "host
route"); outside the region identified by P, the anycast address may
be aggregated into the routing entry for prefix P.

Note that in the worst case, the prefix P of an anycast set may be
the null prefix, i.e., the members of the set may have no topological
locality. In that case, the anycast address must be maintained as a
separate routing entry throughout the entire Internet, which presents
a severe scaling limit on how many such "global" anycast sets may be
supported. Therefore, it is expected that support for global anycast
sets may be unavailable or very restricted.

One expected use of anycast addresses is to identify the set of
routers belonging to an organization providing Internet service.
Such addresses could be used as intermediate addresses in an IPv6
Routing header, to cause a packet to be delivered via a particular
service provider or sequence of service providers.

Some other possible uses are to identify the set of routers attached
to a particular subnet, or the set of routers providing entry into a
particular routing domain.

2.5.1. Required Anycast Address

The Subnet-Router anycast address is predefined. Its format is as
follows:

```
|                      n bits                     |   128-n bits  |
+-------------------------------------------------+---------------+
|                   subnet prefix                 | 00000000000000|
+-------------------------------------------------+---------------+
```

The "subnet prefix" in an anycast address is the prefix that
identifies a specific link. This anycast address is syntactically
the same as a unicast address for an interface on the link with the
interface identifier set to zero.

Packets sent to the Subnet-Router anycast address will be delivered
to one router on the subnet. All routers are required to support the

Subnet-Router anycast addresses for the subnets to which they have interfaces.

The Subnet-Router anycast address is intended to be used for applications where a node needs to communicate with any one of the set of routers.

2.6. Multicast Addresses

An IPv6 multicast address is an identifier for a group of interfaces (typically on different nodes). An interface may belong to any number of multicast groups. Multicast addresses have the following format:

```
|   8    |  4 |  4 |                  112 bits                   |
+------ -+----+----+---------------------------------------------+
|11111111|flgs|scop|                  group ID                   |
+--------+----+----+---------------------------------------------+
```

   binary 11111111 at the start of the address identifies the address
   as being a multicast address.

```
                              +-+-+-+-+
   flgs is a set of 4 flags:   |0|R|P|T|
                              +-+-+-+-+
```

      The high-order flag is reserved, and must be initialized to 0.

      T = 0 indicates a permanently-assigned ("well-known") multicast
      address, assigned by the Internet Assigned Numbers Authority
      (IANA).

      T = 1 indicates a non-permanently-assigned ("transient" or
      "dynamically" assigned) multicast address.

      The P flag's definition and usage can be found in [RFC3306].

      The R flag's definition and usage can be found in [RFC3956].

   scop is a 4-bit multicast scope value used to limit the scope of
   the multicast group. The values are as follows:

      0 reserved
      1 Interface-Local scope
      2 Link-Local scope

```
3 Realm-Local scope
4 Admin-Local scope
5 Site-Local scope
6 (unassigned)
7 (unassigned)
8 Organization-Local scope
9 (unassigned)
A (unassigned)
B (unassigned)
C (unassigned)
D (unassigned)
E Global scope
F reserved
```

Interface-Local scope spans only a single interface on a node
and is useful only for loopback transmission of multicast.
Packets with interface-local scope received from another node
must be discarded.

Link-Local multicast scope spans the same topological region as
the corresponding unicast scope.

Interface-Local, Link-Local, and Realm-Local scope boundaries
are automatically derived from physical connectivity or other
non-multicast-related configurations. Global scope has no
boundary. The boundaries of all other non-reserved scopes of
Admin-Local or larger are administratively configured. For
reserved scopes, the way of configuring their boundaries will
be defined when the semantics of the scope are defined.

According to [RFC4007], the zone of a Realm-Local scope must
fall within zones of larger scope. Because the zone of a
Realm-Local scope is configured automatically while the zones
of larger scopes are configured manually, care must be taken in
the definition of those larger scopes to ensure that the
inclusion constraint is met.

Realm-Local scopes created by different network technologies
are considered to be independent and will have different zone
indices (see Section 6 of [RFC4007]). A router with interfaces
on links using different network technologies does not forward
traffic between the Realm-Local multicast scopes defined by
those technologies.

Site-Local scope is intended to span a single site.

Organization-Local scope is intended to span multiple sites
belonging to a single organization.

scopes labeled "(unassigned)" are available for administrators
to define additional multicast regions.

group ID identifies the multicast group, either permanent or
transient, within the given scope. Additional definitions of the
multicast group ID field structure are provided in [RFC3306].

The "meaning" of a permanently-assigned multicast address is
independent of the scope value. For example, if the "NTP servers
group" is assigned a permanent multicast address with a group ID of
101 (hex), then

ff01:0:0:0:0:0:0:101 means all NTP servers on the same interface
(i.e., the same node) as the sender.

ff02:0:0:0:0:0:0:101 means all NTP servers on the same link as the
sender.

ff05:0:0:0:0:0:0:101 means all NTP servers in the same site as the
sender.

ff0e:0:0:0:0:0:0:101 means all NTP servers in the Internet.

Non-permanently-assigned multicast addresses are meaningful only
within a given scope. For example, a group identified by the non-
permanent, site-local multicast address ff15:0:0:0:0:0:0:101 at one
site bears no relationship to a group using the same address at a
different site, nor to a non-permanent group using the same group ID
with a different scope, nor to a permanent group with the same group
ID.

Multicast addresses must not be used as source addresses in IPv6
packets or appear in any Routing header.

Routers must not forward any multicast packets beyond the scope
indicated by the scop field in the destination multicast address.

Nodes must not originate a packet to a multicast address whose scop
field contains the reserved value 0; if such a packet is received, it
must be silently dropped. Nodes should not originate a packet to a
multicast address whose scop field contains the reserved value F; if
such a packet is sent or received, it must be treated the same as
packets destined to a global (scop E) multicast address.

2.6.1. **Pre-Defined Multicast Addresses**

   The following well-known multicast addresses are pre-defined. The
   group IDs defined in this section are defined for explicit scope
   values.

   Use of these group IDs for any other scope values, with the T flag
   equal to 0, is not allowed.

      Reserved Multicast Addresses: ff00:0:0:0:0:0:0:0
                                    ff01:0:0:0:0:0:0:0
                                    ff02:0:0:0:0:0:0:0
                                    ff03:0:0:0:0:0:0:0
                                    ff04:0:0:0:0:0:0:0
                                    ff05:0:0:0:0:0:0:0
                                    ff06:0:0:0:0:0:0:0
                                    ff07:0:0:0:0:0:0:0
                                    ff08:0:0:0:0:0:0:0
                                    ff09:0:0:0:0:0:0:0
                                    ff0a:0:0:0:0:0:0:0
                                    ff0b:0:0:0:0:0:0:0
                                    ff0c:0:0:0:0:0:0:0
                                    ff0d:0:0:0:0:0:0:0
                                    ff0e:0:0:0:0:0:0:0
                                    ff0f:0:0:0:0:0:0:0

   The above multicast addresses are reserved and shall never be
   assigned to any multicast group.

      All Nodes Addresses: ff01:0:0:0:0:0:0:1
                                   ff02:0:0:0:0:0:0:1

   The above multicast addresses identify the group of all IPv6 nodes,
   within scope 1 (interface-local) or 2 (link-local).

      All Routers Addresses: ff01:0:0:0:0:0:0:2
                                   ff02:0:0:0:0:0:0:2
                                   ff05:0:0:0:0:0:0:2

   The above multicast addresses identify the group of all IPv6 routers,
   within scope 1 (interface-local), 2 (link-local), or 5 (site-local).

      Solicited-Node Address: ff02:0:0:0:0:1:ffxx:xxxx

   Solicited-Node multicast address are computed as a function of a
   node's unicast and anycast addresses. A Solicited-Node multicast
   address is formed by taking the low-order 24 bits of an address
   (unicast or anycast) and appending those bits to the prefix
   FF02:0:0:0:0:1:FF00::/104 resulting in a multicast address in the
   range

      ff02:0:0:0:0:1:ff00:0000

   to

      ff02:0:0:0:0:1:ffff:ffff

   For example, the Solicited-Node multicast address corresponding to
   the IPv6 address 4037::01:800:200e:8c6c is ff02::1:ff0e:8c6c. IPv6
   addresses that differ only in the high-order bits (e.g., due to
   multiple high-order prefixes associated with different aggregations)
   will map to the same Solicited-Node address, thereby reducing the
   number of multicast addresses a node must join.

   A node is required to compute and join (on the appropriate interface)
   the associated Solicited-Node multicast addresses for all unicast and
   anycast addresses that have been configured for the node's interfaces
   (manually or automatically).

   Additional defined multicast address can be found in the IANA IPv6
   Multicast Address Allocation registry [IANA-MC]

2.7. A Node's Required Addresses

   A host is required to recognize the following addresses as
   identifying itself:

      o Its required Link-Local address for each interface.

      o Any additional Unicast and Anycast addresses that have been
         configured for the node's interfaces (manually or
         automatically).

      o The loopback address.

      o The All-Nodes multicast addresses defined in Section 2.6.1.

   o The Solicited-Node multicast address for each of its unicast
      and anycast addresses.

   o Multicast addresses of all other groups to which the node
      belongs.

   A router is required to recognize all addresses that a host is
   required to recognize, plus the following addresses as identifying
   itself:

   o The Subnet-Router Anycast addresses for all interfaces for
      which it is configured to act as a router.

   o All other Anycast addresses with which the router has been
      configured.

   o The All-Routers multicast addresses defined in Section 2.6.1.

3. IANA Considerations

   RFC4291 is referenced in a number of IANA registries. These include:

   o Internet Protocol Version 6 Address Space [IANA-AD]

   o IPv6 Global Unicast Address Assignments [IANA-GU]

   o IPv6 Multicast Address Space Registry [IANA-MC]

   o Application for an IPv6 Multicast Address [IANA-MA]

   o Internet Protocol Version 6 (IPv6) Anycast Addresses [IANA-AC]

   o IANA IPv6 Special-Purpose Address Registry [IANA-SP]

   o Reserved IPv6 Interface Identifiers [IANA-ID]

   o Number Resources [IANA-NR]

   o Protocol Registries [IANA-PR]

   o Technical requirements for authoritative name servers [IANA-NS]

   o IP Flow Information Export (IPFIX) Entities [IANA-FE]

The IANA should update these references to point to this document.
There is a reference to RFC4291 (and RFC3307) that appears to be
incorrect and should be removed in:

   o Modify a Port Number assignment [IANA-PN]

There are also other references in IANA procedures documents that the
IANA should investigate to see if they should be updated.

4. Security Considerations

IPv6 addressing documents do not have any direct impact on Internet
infrastructure security. Authentication of IPv6 packets is defined
in [RFC4302].

One area relavant to IPv6 addressing is privacy. IPv6 addresses can
be created using interface identifiers constructed with unique stable
tokens. The addresses created in this manner can be used to track
the movement of devices across the Internet. Since earlier versions
of this document were published, several approaches have been
developed that mitigate these problems. These are described in
"Security and Privacy Considerations for IPv6 Address Generation
Mechanisms" [RFC7721], "Privacy Extensions for Stateless Address
Autoconfiguration in IPv6" [RFC4941], and "A Method for Generating
Semantically Opaque Interface Identifiers with IPv6 Stateless Address
Autoconfiguration (SLAAC)"[RFC7217].

5. Acknowledgments

The authors would like to acknowledge the contributions of Paul
Francis, Scott Bradner, Jim Bound, Brian Carpenter, Matt Crawford,
Deborah Estrin, Roger Fajman, Bob Fink, Peter Ford, Bob Gilligan,
Dimitry Haskin, Tom Harsch, Christian Huitema, Tony Li, Greg
Minshall, Thomas Narten, Erik Nordmark, Yakov Rekhter, Bill Simpson,
Sue Thomson, Markku Savela, Larry Masinter, Jun-ichiro Itojun Hagino,
Tatuya Jinmei, Suresh Krishnan, and Mahmood Ali.

The authors would also like to acknowledge the authors of the
updating RFCs that were incorporated in this version of the document
to move IPv6 to Internet Standard. This includes Marcelo Bagnulo,
Congxiao Bao, Mohamed Boucadair, Brian Carpenter, Ralph Droms,
Christian Huitema, Sheng Jiang, Seiichi Kawamura, Masanobu Kawashima,
Xing Li, and Stig Venaas.

6. References

6.1. Normative References

[I-D.ietf-6man-rfc2460bis]
          Deering, S. and R. Hinden, "Internet Protocol, Version 6
          (IPv6) Specification", draft-ietf-6man-rfc2460bis-08 (work
          in progress), November 2016.

6.2. Informative References

[BCP198]   Boucadair, M., Petrescu, A., and F. Baker, "IPv6 Prefix
           Length Recommendation for Forwarding", BCP 198, RFC 7608,
           DOI 10.17487/RFC7608, July 2015,
           <http://www.rfc-editor.org/info/rfc7608>.

[EUI64]   "IEEE, "Guidelines for 64-bit Global Identifier (EUI-64)
           Registration Authority"", March 1997,
           <http://standards.ieee.org/regauth/oui/tutorials/
           EUI64.html>.

[IANA-AC] "Internet Protocol Version 6 (IPv6) Anycast Addresses",
           <http://www.iana.org/assignments/ipv6-anycast-addresses/
           ipv6-anycast-addresses.xhtml>.

[IANA-AD] "Internet Protocol Version 6 Address Space",
           <https://www.iana.org/assignments/ipv6-address-space/ipv6-
           address-space.xhtml>.

[IANA-FE] "IP Flow Information Export (IPFIX) Entities",
           <http://www.iana.org/assignments/ipfix/ipfix.xhtml>.

[IANA-GU] "IPv6 Global Unicast Address Assignments",
           <http://www.iana.org/assignments/ipv6-unicast-address-
           assignments/ipv6-unicast-address-assignments.xhtml>.

[IANA-ID] "IANA IPv6 Special-Purpose Address Registry",
           <http://www.iana.org/assignments/ipv6-interface-ids/
           ipv6-interface-ids.xhtml>.

[IANA-MA] "Application for an IPv6 Multicast Address",
           <https://www.iana.org/form/multicast-ipv6>.

[IANA-MC] "IPv6 Multicast Address Space Registry",
           <http://www.iana.org/assignments/ipv6-multicast-addresses/
           ipv6-multicast-addresses.xhtml>.

[IANA-NR] "Number Resources", <http://https://www.iana.org/numbers>.

   [IANA-NS] "Technical requirements for authoritative name servers",
              <https://www.iana.org/help/nameserver-requirements>.

   [IANA-PN] "Modify a Port Number assignment",
              <https://www.iana.org/form/port-modification>.

   [IANA-PR] "Protocol Registries", <https://www.iana.org/protocols>.

   [IANA-SP] "IANA IPv6 Special-Purpose Address Registry",
              <https://www.iana.org/assignments/iana-ipv6-special-
              registry/iana-ipv6-special-registry.xhtml>.

   [RFC2464] Crawford, M., "Transmission of IPv6 Packets over Ethernet
              Networks", RFC 2464, DOI 10.17487/RFC2464, December 1998,
              <http://www.rfc-editor.org/info/rfc2464>.

   [RFC3306] Haberman, B. and D. Thaler, "Unicast-Prefix-based IPv6
              Multicast Addresses", RFC 3306, DOI 10.17487/RFC3306,
              August 2002, <http://www.rfc-editor.org/info/rfc3306>.

   [RFC3513] Hinden, R. and S. Deering, "Internet Protocol Version 6
              (IPv6) Addressing Architecture", RFC 3513, DOI 10.17487/
              RFC3513, April 2003,
              <http://www.rfc-editor.org/info/rfc3513>.

   [RFC3587] Hinden, R., Deering, S., and E. Nordmark, "IPv6 Global
              Unicast Address Format", RFC 3587, DOI 10.17487/RFC3587,
              August 2003, <http://www.rfc-editor.org/info/rfc3587>.

   [RFC3879] Huitema, C. and B. Carpenter, "Deprecating Site Local
              Addresses", RFC 3879, DOI 10.17487/RFC3879, September
              2004, <http://www.rfc-editor.org/info/rfc3879>.

   [RFC3956] Savola, P. and B. Haberman, "Embedding the Rendezvous
              Point (RP) Address in an IPv6 Multicast Address", RFC
              3956, DOI 10.17487/RFC3956, November 2004,
              <http://www.rfc-editor.org/info/rfc3956>.

   [RFC4007] Deering, S., Haberman, B., Jinmei, T., Nordmark, E., and
              B. Zill, "IPv6 Scoped Address Architecture", RFC 4007, DOI
              10.17487/RFC4007, March 2005,
              <http://www.rfc-editor.org/info/rfc4007>.

   [RFC4038] Shin, M-K., Ed., Hong, Y-G., Hagino, J., Savola, P., and
              E. Castro, "Application Aspects of IPv6 Transition", RFC
              4038, DOI 10.17487/RFC4038, March 2005,
              <http://www.rfc-editor.org/info/rfc4038>.

   [RFC4193] Hinden, R. and B. Haberman, "Unique Local IPv6 Unicast
              Addresses", RFC 4193, DOI 10.17487/RFC4193, October 2005,
              <http://www.rfc-editor.org/info/rfc4193>.

   [RFC4302] Kent, S., "IP Authentication Header", RFC 4302, DOI
              10.17487/RFC4302, December 2005,
              <http://www.rfc-editor.org/info/rfc4302>.

   [RFC4632] Fuller, V. and T. Li, "Classless Inter-domain Routing
              (CIDR): The Internet Address Assignment and Aggregation
              Plan", BCP 122, RFC 4632, DOI 10.17487/RFC4632, August
              2006, <http://www.rfc-editor.org/info/rfc4632>.

   [RFC4941] Narten, T., Draves, R., and S. Krishnan, "Privacy
              Extensions for Stateless Address Autoconfiguration in
              IPv6", RFC 4941, DOI 10.17487/RFC4941, September 2007,
              <http://www.rfc-editor.org/info/rfc4941>.

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

   [RFC6164] Kohno, M., Nitzan, B., Bush, R., Matsuzaki, Y., Colitti,
              L., and T. Narten, "Using 127-Bit IPv6 Prefixes on Inter-
              Router Links", RFC 6164, DOI 10.17487/RFC6164, April 2011,
              <http://www.rfc-editor.org/info/rfc6164>.

   [RFC7217] Gont, F., "A Method for Generating Semantically Opaque
              Interface Identifiers with IPv6 Stateless Address
              Autoconfiguration (SLAAC)", RFC 7217, DOI 10.17487/
              RFC7217, April 2014,
              <http://www.rfc-editor.org/info/rfc7217>.

   [RFC7421] Carpenter, B., Ed., Chown, T., Gont, F., Jiang, S.,
              Petrescu, A., and A. Yourtchenko, "Analysis of the 64-bit
              Boundary in IPv6 Addressing", RFC 7421, DOI 10.17487/
              RFC7421, January 2015,
              <http://www.rfc-editor.org/info/rfc7421>.

   [RFC7428] Brandt, A. and J. Buron, "Transmission of IPv6 Packets
              over ITU-T G.9959 Networks", RFC 7428, DOI 10.17487/
              RFC7428, February 2015,
              <http://www.rfc-editor.org/info/rfc7428>.

   [RFC7721] Cooper, A., Gont, F., and D. Thaler, "Security and Privacy
              Considerations for IPv6 Address Generation Mechanisms",
              RFC 7721, DOI 10.17487/RFC7721, March 2016,
              <http://www.rfc-editor.org/info/rfc7721>.

Appendix A. Modified EUI-64 Format Interface Identifiers

   Modified EUI-64 format-based interface identifiers may have universal
   scope when derived from a universal token (e.g., IEEE 802 48-bit MAC
   or IEEE EUI-64 identifiers [EUI64]) or may have local scope where a
   global token is not being used (e.g., serial links, tunnel end-
   points) or where global tokens are undesirable (e.g., temporary
   tokens for privacy [RFC4941]).

   Modified EUI-64 format interface identifiers are formed by inverting
   the "u" bit (universal/local bit in IEEE EUI-64 terminology) when
   forming the interface identifier from IEEE EUI-64 identifiers. In
   the resulting Modified EUI-64 format, the "u" bit is set to one (1)
   to indicate universal scope, and it is set to zero (0) to indicate
   local scope. The first three octets in binary of an IEEE EUI-64
   identifier are as follows:

```
       0        0 0      1 1       2
      |0       7 8      5 6       3|
       +----+----+----+----+----+----+
       |cccc|ccug|cccc|cccc|cccc|cccc|
       +----+----+----+----+----+----+
```

   written in Internet standard bit-order, where "u" is the universal/
   local bit, "g" is the individual/group bit, and "c" is the bits of
   the company_id. Appendix A, "Creating Modified EUI-64 Format
   Interface Identifiers", provides examples on the creation of Modified
   EUI-64 format-based interface identifiers.

   The motivation for inverting the "u" bit when forming an interface
   identifier is to make it easy for system administrators to hand
   configure non-global identifiers when hardware tokens are not
   available. This is expected to be the case for serial links and
   tunnel end-points, for example. The alternative would have been for
   these to be of the form 0200:0:0:1, 0200:0:0:2, etc., instead of the
   much simpler 0:0:0:1, 0:0:0:2, etc.

   IPv6 nodes are not required to validate that interface identifiers
   created with modified EUI-64 tokens with the "u" bit set to universal
   are unique.

A.1. Creating Modified EUI-64 Format Interface Identifiers

   Depending on the characteristics of a specific link or node, there
   are a number of approaches for creating Modified EUI-64 format
   interface identifiers. This appendix describes some of these
   approaches.

   Links or Nodes with IEEE EUI-64 Identifiers

   The only change needed to transform an IEEE EUI-64 identifier to an
   interface identifier is to invert the "u" (universal/local) bit. An
   example is a globally unique IEEE EUI-64 identifier of the form:

```
|0              1|1              3|3              4|4              6|
|0              5|6              1|2              7|8              3|
+----------------+----------------+----------------+----------------+
|cccccc0gcccccccc|ccccccccmmmmmmmm|mmmmmmmmmmmmmmmm|mmmmmmmmmmmmmmmm|
+----------------+----------------+----------------+----------------+
```

   where "c" is the bits of the assigned company_id, "0" is the value of
   the universal/local bit to indicate universal scope, "g" is
   individual/group bit, and "m" is the bits of the manufacturer-
   selected extension identifier. The IPv6 interface identifier would
   be of the form:

```
|0              1|1              3|3              4|4              6|
|0              5|6              1|2              7|8              3|
+----------------+----------------+----------------+----------------+
|cccccc1gcccccccc|ccccccccmmmmmmmm|mmmmmmmmmmmmmmmm|mmmmmmmmmmmmmmmm|
+----------------+----------------+----------------+----------------+
```

   The only change is inverting the value of the universal/local bit.

   Links or Nodes with IEEE 802 48-bit MACs

   [EUI64] defines a method to create an IEEE EUI-64 identifier from an
   IEEE 48-bit MAC identifier. This is to insert two octets, with
   hexadecimal values of 0xFF and 0xFE (see the Note at the end of
   appendix), in the middle of the 48-bit MAC (between the company_id
   and vendor-supplied id). An example is the 48-bit IEEE MAC with
   Global scope:

```
|0              1|1              3|3              4|
|0              5|6              1|2              7|
+----------------+----------------+----------------+
|cccccc0gcccccccc|ccccccccmmmmmmmm|mmmmmmmmmmmmmmmm|
+----------------+----------------+----------------+
```

where "c" is the bits of the assigned company_id, "0" is the value of
the universal/local bit to indicate Global scope, "g" is individual/
group bit, and "m" is the bits of the manufacturer- selected
extension identifier. The interface identifier would be of the form:

```
|0              1|1              3|3              4|4              6|
|0              5|6              1|2              7|8              3|
+---------------+---------------+---------------+---------------+
|cccccc1gcccccccc|cccccccc11111111|11111110mmmmmmmm|mmmmmmmmmmmmmmmm|
+---------------+---------------+---------------+---------------+
```

When IEEE 802 48-bit MAC addresses are available (on an interface or
a node), an implementation may use them to create interface
identifiers due to their availability and uniqueness properties.

Links with Other Kinds of Identifiers

There are a number of types of links that have link-layer interface
identifiers other than IEEE EUI-64 or IEEE 802 48-bit MACs. Examples
include LocalTalk and Arcnet. The method to create a Modified EUI-64
format identifier is to take the link identifier (e.g., the LocalTalk
8-bit node identifier) and zero fill it to the left. For example, a
LocalTalk 8-bit node identifier of hexadecimal value 0x4F results in
the following interface identifier:

```
|0              1|1              3|3              4|4              6|
|0              5|6              1|2              7|8              3|
+---------------+---------------+---------------+---------------+
|0000000000000000|0000000000000000|0000000000000000|0000000001001111|
+---------------+---------------+---------------+---------------+
```

Note that this results in the universal/local bit set to "0" to
indicate local scope.

Links without Identifiers

There are a number of links that do not have any type of built-in
identifier. The most common of these are serial links and configured
tunnels. Interface identifiers that are unique within a subnet
prefix must be chosen.

When no built-in identifier is available on a link, the preferred
approach is to use a universal interface identifier from another
interface or one that is assigned to the node itself. When using
this approach, no other interface connecting the same node to the
same subnet prefix may use the same identifier.

If there is no universal interface identifier available for use on
the link, the implementation needs to create a local-scope interface
identifier. The only requirement is that it be unique within a
subnet prefix. There are many possible approaches to select a
subnet-prefix-unique interface identifier. These include the
following:

    Manual Configuration
    Node Serial Number
    Other Node-Specific Token

The subnet-prefix-unique interface identifier should be generated in
a manner such that it does not change after a reboot of a node or if
interfaces are added or deleted from the node.

The selection of the appropriate algorithm is link and implementation
dependent. The details on forming interface identifiers are defined
in the appropriate "IPv6 over <link>" specification. It is strongly
recommended that a collision detection algorithm be implemented as
part of any automatic algorithm.

Note: [EUI64] actually defines 0xFF and 0xFF as the bits to be
        inserted to create an IEEE EUI-64 identifier from an IEEE MAC-
        48 identifier. The 0xFF and 0xFE values are used when
        starting with an IEEE EUI-48 identifier. The incorrect value
        was used in earlier versions of the specification due to a
        misunderstanding about the differences between IEEE MAC-48 and
        EUI-48 identifiers.

        This document purposely continues the use of 0xFF and 0xFE
        because it meets the requirements for IPv6 interface
        identifiers (i.e., that they must be unique on the link), IEEE
        EUI-48 and MAC-48 identifiers are syntactically equivalent,
        and that it doesn't cause any problems in practice.

Appendix B. CHANGES SINCE RFC 4291

This document has the following changes from RFC4291, "IP Version 6
Addressing Architecture". Numbers identify the Internet-Draft
version that the change was made.:

Working Group Internet Drafts

    07) Added text to Section 2.4 summarizing IPv6 unicast routing
         and referencing BCP198, citing RFC6164 as an example of

longer prefixes, and that IIDs are required to be 64 bits
long as described in RFC7421.

07) Based on review by Brian Haberman added reference to RFC5952
     in Section 2.2.3, corrected case errors in Section 2.6.1, and
     added a reference to the IANA Multicast address registry in
     Section 2.6.1.

07) Corrected errors in Section 2.2.3 where the examples in 7.
     and 8. were reversed.

07) Editorial changes.

06) Editorial changes.

05) Expanded Security Considerations Section to discuss privacy
     issues related to using stable interface identifiers to
     create IPv6 addresses, and reference solutions that mitigate
     these issues such as RFC7721, RFC4941, RFC7271.

05) Added instructions in IANA Considerations to update
     references in the IANA registries that currently point to
     RFC4291 to point to this document.

05) Rename Section 2.4.7 to "Other Local Unicast Addresses" and
     rewrote the text to point to ULAs and say that Site-Local
     addresses were deprecated by RFC3879. The format of Site-
     Local was removed.

05) Added to Section 2.4.1 a reference to RFC7421 regarding the
     background on the 64 bit boundary in Interface Identifiers.

05) Editorial changes.

04) Added text and a pointer to the ULA specification in
     Section 2.4.7

04) Removed old IANA Considerations text, this was left from the
     baseline text from RFC4291 and should have been removed
     earlier.

04) Editorial changes.

03) Changes references in Section 2.4.1 that describes the
     details of forming IIDs to RFC7271 and RFC7721.

02) Remove changes made by RFC7371 because there isn't any known
     implementation experience.

01) Revised Section 2.4.1 on Interface Identifiers to reflect
     current approach, this included saying Modified EUI-64
     identifiers not recommended and moved the text describing the
     format to Appendix A.

01) Editorial changes.

00) Working Group Draft.

00) Editorial changes.

Individual Internet Drafts

06) Incorporate the updates made by RFC7371. The changes were to
     the flag bits and their definitions in Section 2.6.

05) Incorporate the updates made by RFC7346. The change was to
     add Realm-Local scope to the multicast scope table in
     Section 2.6, and add the updating text to the same section.

04) Incorporate the updates made by RFC6052. The change was to
     add a text in Section 2.3 that points to the IANA registries
     that records the prefix defined in RFC6052 and a number of
     other special use prefixes.

03) Incorporate the updates made by RFC7136 to deprecate the U
     and G bits in Modified EUI-64 format Internet IDs.

03) Add note to the reference section acknowledging the authors
     of the updating documents.

03) Editorial changes.

02) Updates to resolve the open Errata on RFC4291. These are:

     Errata ID: 3480: Corrects the definition of Interface-
     Local multicast scope to also state that packets with
     interface-local scope received from another node must be
     discarded.

     Errata ID: 1627: Remove extraneous "of" in Section 2.7.

     Errata ID: 2702: This errata is marked rejected. No
     change is required.

          Errata ID: 2735: This errata is marked rejected. No
          change is required.

          Errata ID: 4406: This errata is marked rejected. No
          change is required.

          Errata ID: 2406: This errata is marked rejected. No
          change is required.

          Errata ID: 863: This errata is marked rejected. No change
          is required.

          Errata ID: 864: This errata is marked rejected. No change
          is required.

          Errata ID: 866: This errata is marked rejected. No change
          is required.

   02) Update references to current versions.

   02) Editorial changes.

   01) Incorporate the updates made by RFC5952 regarding the text
        format when outputting IPv6 addresses. A new section was
        added for this and addresses shown in this document were
        changed to lower case.

   01) Revise this Section to document to show the changes from
        RFC4291.

   01) Editorial changes.

   00) Establish a baseline from RFC4291. The only intended changes
        are formatting (XML is slightly different from .nroff),
        differences between an RFC and Internet Draft, fixing a few
        ID Nits, and updates to the authors information. There
        should not be any content changes to the specification.

Authors' Addresses

   Robert M. Hinden
   Check Point Software
   959 Skyway Road
   San Carlos, CA 94070
   USA

   Email: bob.hinden@gmail.com

   Stephen E. Deering
   Retired
   Vancouver, British Columbia
   Canada