IETF IPv6 Working Group
Internet-Draft
Expires: August 13, 2004

S. Deering
Cisco Systems
B. Haberman
Caspian Networks
T. Jinmei
Toshiba
E. Nordmark
Sun Microsystems
B. Zill
Microsoft

February 13, 2004

IPv6 Scoped Address Architecture
draft-ietf-ipv6-scoping-arch-01.txt

Status of this Memo

Copyright Notice

Abstract

This document specifies the architectural characteristics, expected behavior, textual representation, and usage of IPv6 addresses of different scopes. According to a decision in the IPv6 working group, this document intentionally avoids using the syntax and usage of unicast site-local addresses.

1. Introduction

Internet Protocol version 6 includes support for addresses of different "scope", that is, both global and non-global (e.g., link-local) addresses. While non-global addressing has been introduced operationally in the IPv4 Internet, both in the use of private address space ("net 10", etc.) and with administratively scoped multicast addresses, the design of IPv6 formally incorporates the notion of address scope into its base architecture. This document specifies the architectural characteristics, expected behavior, textual representation, and usage of IPv6 addresses of different scopes.

Though the current address architecture specification [1] defines unicast site-local addresses, the IPv6 working group decided to deprecate the syntax and the usage [5], and is now investigating other forms of local IPv6 addressing. The usage of any new forms of local addresses will be documented elsewhere in the future. Thus, this document intentionally focuses on link-local and multicast scopes only.

2. Definitions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [2].

3. Basic Terminology

The terms link, interface, node, host, and router are defined in [3]. The definitions of unicast address scopes (link-local and global) and multicast address scopes (interface-local, link-local, etc.) are contained in [1].

4. Address Scope

Every IPv6 address other than the unspecified address has a specific scope, that is, a topological span within which the address may be used as a unique identifier for an interface or set of interfaces. The scope of an address is encoded as part of the address, as specified in [1].

For unicast addresses, this document discusses two defined scopes:

- o Link-local scope, for uniquely identifying interfaces within (i.e., attached to) a single link only.

- o Global scope, for uniquely identifying interfaces anywhere in the

Internet.

The IPv6 unicast loopback address, ::1, is treated as having link-local scope within an imaginary link to which a virtual "loopback interface" is attached.

The unspecified address, ::, is a special case. It does not have any scope, because it must never be assigned to any node according to [1]. Note, however, that an implementation might use an implementation dependent semantics for the unspecified address and may want to allow the unspecified address to have specific scopes. For example, implementations often use the unspecified address to represent "any" address in APIs. In such a case, implementations may want to regard the address in a particular scope to represent the notion of "any addresses in the scope." This document does not prohibit such a usage, as long as it is limited within the implementation.

[1] defines IPv6 addresses with embedded IPv4 addresses as part of global addresses. Thus, those addresses have global scope, with regards to the IPv6 scoped address architecture. However, an implementation may use those addresses as if they had other type of scopes for convenience. For instance, [6] assigns link-local scope to IPv4 auto-configured link-local addresses (the addresses from the prefix 169.254.0.0/16 [7]), and converts those addresses into IPv4-mapped IPv6 addresses in order to perform destination address selection among IPv4 and IPv6 addresses. This would implicitly mean the IPv4-mapped IPv6 addresses equivalent to the IPv4 auto-configuration link-local addresses have link-local scope. This document does not preclude such a usage, as long as it is limited within the implementation.

Anycast addresses [1] are allocated from the unicast address space and have the same scope properties as unicast addresses. All statements in this document regarding unicast apply equally to anycast.

For multicast addresses, there are fourteen possible scopes, ranging from interface-local to global (including link-local). The interface-local scope spans a single interface only; a multicast address of interface-local scope is useful only for loopback delivery of multicasts within a single node, for example, as a form of inter-process communication within a computer. Unlike the unicast loopback address, interface-local multicast addresses may be assigned to any interface.

There is a size relationship among scopes:

- for unicast scopes, link-local is a smaller scope than global.

- for multicast scopes, scopes with lesser values in the "scop" subfield of the multicast address (Section 2.7 of [1]) are smaller than scopes with greater values, with interface-local being the smallest and global being the largest.

However, two scopes of different size may cover the exact same region of topology. For example, a (multicast) site may consist of a single link, in which both link-local and site-local scope effectively cover the same topological span.

5. Scope Zones

A scope zone, or simply a zone, is a connected region of topology of a given scope. For example, the set of links connected by routers within a particular (multicast) site, and the interfaces attached to those links, comprise a single zone of multicast site-local scope. Note that a zone is a particular instance of a topological region (e.g., Alice's site or Bob's site), whereas a scope is the size of a topological region (i.e., a site or a link or a ...).

The zone to which a particular non-global address pertains is not encoded in the address itself, but rather is determined by context, such as the interface from which it is sent or received. Thus, addresses of a given (non-global) scope may be re-used in different zones of that scope. For example, two different physical links may each contain a node with link-local address fe80::1.

Zones of the different scopes are instantiated as follows:

- Each interface on a node comprises a single zone of interface-local scope (for multicast only).

- Each link, and the interfaces attached to that link, comprises a single zone of link-local scope (for both unicast and multicast).

- There is a single zone of global scope (for both unicast and multicast), comprising all the links and interfaces in the Internet.

- The boundaries of zones of scope other than interface-local, link-local, and global must be defined and configured by network administrators.

Zone boundaries are relatively static features, not changing in response to short-term changes in topology. Thus, the requirement that the topology within a zone be "connected" is intended to include

links and interfaces that may be only occasionally connected. For example, a residential node or network that obtains Internet access by dial-up to an employer's (multicast) site may be treated as part of the employer's (multicast) site-local zone even when the dial-up link is disconnected. Similarly, a failure of a router, interface, or link that causes a zone to become partitioned does not split that zone into multiple zones; rather, the different partitions are still considered to belong to the same zone.

Zones have the following additional properties:

o Zone boundaries cut through nodes, not links. (Note that the global zone has no boundary, and the boundary of an interface-local zone encloses just a single interface.)

o Zones of the same scope cannot overlap, i.e., they can have no links or interfaces in common.

o A zone of a given scope (less than global) falls completely within zones of larger scope, i.e., a smaller scope zone cannot include more topology than any larger scope zone with which it shares any links or interfaces.

o Each zone is required to be "convex" from a routing perspective, i.e., packets sent from one interface to any other interface in the same zone are never routed outside the zone.

Each interface belongs to exactly one zone of each possible scope. Note that this means an interface belongs to a scope zone regardless of what kind of unicast address the interface has or of which multicast groups the node joins on the interface.

6. Zone Indices

Considering the fact that the same non-global address may be in use in more than one zone of the same scope (e.g., the use of link-local address fe80::1 in two separate physical links), and that a node may have interfaces attached to different zones of the same scope (e.g., a router normally has multiple interfaces attached to different links), a node requires an internal means of identifying to which zone a non-global address belongs. This is accomplished by assigning, within the node, a distinct "zone index" to each zone of the same scope to which that node is attached, and allowing all internal uses of an address to be qualified by a zone index.

The assignment of zone indices is illustrated in the example in the figure below:

```
         ---------------------------------------------------------------
        | a node                                                        |
        |                                                               |
        |                                                               |
        |                                                               |
        |                                                               |
        |                                                               |
        |                                                               |
        |  /--link1--\ /--------link2--------\ /--link3--\ /--link4--\  |
        |                                                               |
        |  /--intf1--\ /--intf2--\ /--intf3--\ /--intf4--\ /--intf5--\  |
         ---------------------------------------------------------------
                :           |           |           |           |
                :           |           |           |           |
                :           |           |           |           |
            (imaginary    =================      a point-       a
             loopback        an Ethernet         to-point     tunnel
               link)                               link
```

Figure 1: Zone Indices Example

This example node has five interfaces:

A loopback interface to the imaginary loopback link (a phantom link that goes nowhere),

Two interfaces to the same Ethernet link,

An interface to a point-to-point link, and

A tunnel interface (e.g., the abstract endpoint of an IPv6-over-IPv6 tunnel [8], presumably established over either the Ethernet or the point-to-point link.)

It is thus attached to five interface-local zones, identified by the interface indices 1 through 5.

Because the two Ethernet interfaces are attached to the same link, the node is attached to only four link-local zones, identified by link indices 1 through 4.

Each zone index of a particular scope should contain an information to represent the scope type, so that all indices of all scopes are unique within the node and zone indices themselves can be used for a dedicated purpose. An entry of a Management Information Base (MIB) will be an example of the dedicated purpose. The actual representation to encode the scope type is implementation dependent and is out of scope of this document. Within this document, indices

are simply represented like "link index 2" for readability.

The zone indices are strictly local to the node. For example, the node on the other end of the point-to-point link may well be using entirely different interface and link index values for that link.

An implementation should also support the concept of a "default" zone for each scope. And, when supported, the index value zero at each scope SHOULD be reserved to mean "use the default zone". Unlike other zone indices, the default ID does not contain any scope type, and the scope type is determined by the address by which the default ID was accompanied. An implementation may additionally define a separate default zone for each scope type. Those default indices can also be used as the zone qualifier for an address for which the node is attached to only one zone, e.g., when using global addresses.

There is at present no way for a node to automatically determine which of its interfaces belong to the same zones, e.g., the same link or the same multicast scope zone larger than interface. In the future, protocols may be developed to determine that information. In the absence of such protocols, an implementation must provide a means for manual assignment and/or reassignment of zone indices. Furthermore, to avoid the need to perform manual configuration in most cases, an implementation should, by default, initially assign zone indices as follows, and only as follows:

o A unique interface index for each interface

o A unique link index for each interface

Then, manual configuration would be necessary only for the less common cases of nodes with multiple interfaces to a single link or interfaces to zones of different (multicast-only) scopes.

Thus, the default zone index assignments for the example node from Figure 1 would be as illustrated in Figure 2, below. Manual configuration would then be required to, for example, assign the same link index to the two Ethernet interfaces as shown in Figure 1.

```
      ---------------------------------------------------------------
     | a node                                                        |
     |                                                               |
     |                                                               |
     |                                                               |
     |                                                               |
     |                                                               |
     |  /--link1--\ /--link2--\ /--link3--\ /--link4--\ /--link5--\  |
     |                                                               |
```

```
        |  /--intf1--\ /--intf2--\ /--intf3--\ /--intf4--\ /--intf5--\  |
         ---------------------------------------------------------------
                ::           |           |           |           |
                :            |           |           |           |
                :            |           |           |           |
            (imaginary    =================      a point-       a
             loopback        an Ethernet         to-point     tunnel
               link)                               link
```

Figure 2: Example of Default Zone Indices

As well as initially assigning zone indices, as specified above, an implementation should automatically select a default zone for each scope for which there is more than one choice, to be used whenever an address is specified without a zone index (or with a zone index of zero). For instance, in the example shown in Figure 2, the implementation might automatically select intf2 and link2 as the default zones for each of those three scopes. (Perhaps the selection algorithm is to choose the first zone that includes an interface other than the loopback interface as the default for each scope.) A means must also be provided for manually assigning the default zone for a scope, overriding any automatic assignment.

Because the unicast loopback address, ::1, may not be assigned to any interface other than the loopback interface, it is recommended that, whenever ::1 is specified without a zone index, or with the default zone index, it be interpreted as belonging to the loopback link-local zone, regardless of which link-local zone has been selected as the default. If this is done, then in the common case of nodes with only a single non-loopback interface (e.g., a single Ethernet interface), it becomes possible to avoid any need to qualify link-local addresses with a zone index: the unqualified address ::1 would always refer to the link-local zone containing the loopback interface, and all other unqualified link-local addresses would refer to the link-local zone containing the non-loopback interface (as long as the default link-local zone were set to be the zone containing the non-loopback interface).

Because of the requirement that a zone of a given scope fall completely within zones of larger scope (see Section 5, above), if two interfaces are assigned to different zones of scope S, they must also be assigned to different zones of all scopes smaller than S. Thus, the manual assignment of distinct zone indices for one scope may require the automatic assignment of distinct zone indices for smaller scopes. For example, suppose that distinct multicast site-local indices 1 and 2 are manually assigned in Figure 1 and that site 1 contains link 1, 2, and 3, while site 2 only contains link 4. This configuration would then cause the automatic creation of

corresponding admin-local (i.e. multicast "scop" value 4) indices 1 and 2, because admin-local scope is smaller than site-local scope.

Taking all of the above considerations in account, the complete set of zone indices for our example node from Figure 1 with the additional configurations here is shown in Figure 3, below.

```
         ---------------------------------------------------------------
        | a node                                                        |
        |                                                               |
        |                                                               |
        |                                                               |
        |                                                               |
        |                                                               |
        |  /--------------------site1--------------------\ /--site2--\  |
        |                                                               |
        |  /-------------------admin1--------------------\ /-admin2--\  |
        |                                                               |
        |  /--link1--\ /--------link2--------\ /--link3--\ /--link4--\  |
        |                                                               |
        |  /--intf1--\ /--intf2--\ /--intf3--\ /--intf4--\ /--intf5--\  |
         ---------------------------------------------------------------
                :           |           |           |           |
                :           |           |           |           |
                :           |           |           |           |
            (imaginary    =================      a point-       a
             loopback       an Ethernet          to-point     tunnel
               link)                               link
```

Figure 3: Complete Zone Indices Example

Although the examples above show the zones being assigned index values sequentially for each scope, starting at one, the zone index values are arbitrary. An implementation may use any value it chooses to label a zone as long as it meets the requirement that the index value of each zone of all scopes be unique within the node and that zero SHOULD be reserved to represent the default zone. Implementations choosing to follow the recommended basic API [10] will want to restrict their index values to those that can be represented by the sin6_scope_id field of a sockaddr_in6 structure.

7. Sending Packets

When an upper-layer protocol sends a packet to a non-global destination address, it must have a means of identifying to the IPv6 layer the intended zone, for cases in which the node is attached to more than one zone of the destination address's scope.

Although identification of an outgoing interface is sufficient to identify an intended zone (because each interface is attached to no more than one zone of each scope), that is more specific than desired in many cases. For example, when sending to a link-local unicast address, from a node that has more than one interface to the intended link (though this is an unusual configuration), the upper layer protocol may not care which of those interfaces is used for the transmission, but rather would prefer to leave that choice to the routing function in the IP layer. Thus, the upper-layer requires the ability to specify a zone index, rather than an interface identifier, when sending to a non-global, non-loopback destination address.

8. Receiving Packets

When an upper-layer protocol receives a packet containing a non-global source or destination address, the zone to which that address pertains can be determined from the arrival interface, because the arrival interface can be attached to only one zone of the same scope as the address under consideration. However, it is recommended that the IP layer convey to the upper layer the correct zone indices for the arriving source and destination addresses, in addition to the arrival interface identifier.

9. Forwarding

When a router receives a packet addressed to a node other than itself, it must take the zone of the destination and source addresses into account as follows:

- The zone of the destination address is determined by the scope of the address and arrival interface of the packet. The next-hop interface is chosen by looking up the destination address in a (conceptual) routing table specific to that zone. That routing table is restricted to refer only to interfaces belonging to that zone.

- After the next-hop interface is chosen, the zone of the source address is considered. As with the destination address, the zone of the source address is determined by the scope of the address and arrival interface of the packet. If transmitting the packet on the chosen next-hop interface would cause the packet to leave the zone of the source address, i.e., cross a zone boundary of the scope of the source address, then the packet is discarded. Additionally, if the packet's destination address is a unicast address, an ICMP Destination Unreachable message [4] with Code 2 ("beyond scope of source address") is sent to the source of the original packet. Note: Code 2 is currently left as unassigned in [4]. But the IANA is going to re-assign the value for the new

purpose, and [4] will be revised with this change.

Note that even with the decision that unicast site-local addresses are deprecated, the above procedure still applies to link-local addresses. Thus, if a router receives a packet with a link-local destination address that is not one of the router's own link-local addresses on the arrival link, the router is expected to try to forward the packet to the destination on that link (subject to successful determination of the destination's link-layer address via the Neighbor Discovery protocol [9]). The forwarded packet may be transmitted back out the arrival interface, or out any other interface attached to the same link.

A node that receives a packet addressed to itself and containing a Routing Header with more than zero Segments Left (Section 4.4 of [3]) first checks the scope of the next address in the Routing Header. If the scope of the next address is smaller than the scope of the original destination address, the node MUST discard the packet. Otherwise, it swaps the original destination address with the next address in the Routing Header. Then the above forwarding rules apply as follows:

- The zone of the new destination address is determined by the scope of the next address and the arrival interface of the packet. The next-hop interface is chosen just like the first bullet of the rules above.

- After the next-hop interface is chosen, the zone of the source address is considered just like the second bullet of the rules above.

This check about the scope of the next address ensures that when a packet arrives at its final destination, if that destination is link-local then the receiving node can know that the packet originated on-link. As a result, this will help the receiving node send a "response" packet with the final destination of the received packet as the source address without breaking its source zone.

Note that it is possible, though generally inadvisable, to use a Routing Header to convey a non-global address across its associated zone boundary in the previously-used next address field. For example, consider a case where a link-border node (e.g., a router) receives a packet with the destination being a link-local address, and the source address a global address. If the packet contains a Routing Header where the next address is a global address, the next-hop interface to the global address may belong to a different link than that of the original destination. This is allowed, because the scope of the next address is not smaller than the scope of the original

destination.

10. Routing

Note: since unicast site-local addresses are deprecated, and link-local addresses do not need routing, the discussion in this section only applies to multicast scoped routing.

When a routing protocol determines that it is operating on a zone boundary, it MUST protect inter-zone integrity and maintain intra-zone connectivity.

In order to maintain connectivity, the routing protocol must be able to create forwarding information for the global groups as well as for all of the scoped groups for each of its attached zones. The most straightforward way of doing this is to create (conceptual) forwarding tables for each specific zone.

To protect inter-zone integrity, routers must be selective in the group information that is shared with neighboring routers. Routers routinely exchange routing information with neighboring routers. When a router is transmitting this routing information, it must not include any information about zones other than the zones assigned to the interface used to transmit the information.

```
                               *                                 *
                               *                                 *
                               *   ===========    Organization X *
                               *    |       |                    *
                               *    |       |                    *
                             +-*----|-------|------+             *
                             | *  intf1   intf2    |             *
                             | *                   |             *
                             | *             intf3 ---           *
                             | *                   |             *
                             | ***********************************
                             |                     |
                             |        Router       |
                             |                     |
               **********************       **********************
                             |       *     *       |
                  Org. Y   --- intf4  *   *  intf5 ---   Org. Z
                             |       *     *       |
               **********************       **********************
                             +---------------------+
```

Figure 4: Multi-Organization Multicast Router

As an example, the router in Figure 4 must exchange routing information on five interfaces. The information exchanged is as follows (for simplicity, multicast scopes smaller or larger than organization except global are not considered here):

- o Interface 1
  - * All global groups
  - * All organization groups learned from Interfaces 1, 2, and 3
- o Interface 2
  - * All global groups
  - * All organization groups learned from Interfaces 1, 2, and 3
- o Interface 3
  - * All global groups
  - * All organization groups learned from Interfaces 1, 2, and 3
- o Interface 4
  - * All global groups
  - * All organization groups learned from Interface 4
- o Interface 5
  - * All global groups
  - * All organization groups learned from Interface 5

By imposing route exchange rules, zone integrity is maintained by keeping all zone-specific routing information contained within the zone.

11. Textual Representation

As already mentioned, to specify an IPv6 non-global address without ambiguity, an intended scope zone should be specified as well. As a common notation to specify the scope zone, an implementation SHOULD support the following format.

```
<address>%<zone_id>
```

where

<address> is a literal IPv6 address,

<zone_id> is a string to identify the zone of the address, and

`%' is a delimeter character to distinguish between <address> and <zone_id>.

The following subsections describe detail definitions, concrete examples, and additional notes of the format.

11.1 Non-Global Addresses

The format applies to all kinds of unicast and multicast addresses of non-global scope except the unspecified address, which does not have a scope. The format is meaningless and should not be used for global addresses. The loopback address belongs to the trivial link, i.e., the link attached to the loopback interface, thus the format should not be used for the loopback address either. This document does not specify the usage of the format when the <address> is the unspecified address, since the address does not have a scope. This document, however, does not prohibit an implementation from using the format for those special addresses for implementation dependent purposes.

11.2 Zone Indices

In the textual representation, the <zone_id> part should be able to identify a particular zone of the address's scope. Although a zone index is expected to contain the scope type and to be unique among all scopes as described in Section 6 of this document, the <zone_id> part of this format does not have to contain the scope type because the <address> part should specify the appropriate scope. This also means the <zone_id> part does not have to be unique among all scopes.

With this loosened property, an implementation can use convenient representation as <zone_id>. For example, to represent link index 2, the implementation can simply use "2" as <zone_id>, which would be more readable than other representation that contains the scope type "link".

When an implementation interprets the format, it should construct the "full" zone ID, which contains the scope type, from the <zone_id> part and the scope type specified by the <address> part. (Remember that a zone index itself should contain the scope type as specified in Section 6.)

An implementation SHOULD support at least numerical indices as

<zone_id>, which are non-negative decimal integers. The default zone ID, which should typically be 0 (see Section 6), is included in the integers. When <zone_id> is the default, the delimiter character, "%", and <zone_id> can be omitted. Similarly, if a textual representation of an IPv6 address is given without a zone ID, it should be interpreted as <address>%<default ID> where <default ID> is the default zone ID of the scope that <address> has.

An implementation MAY support other kinds of non-null strings as <zone_id>. However, the strings must not conflict with the delimiter character. The precise format and semantics of such additional strings is implementation dependent.

One possible candidate of such strings would be interface names, since interfaces uniquely disambiguate any type of scopes. In particular, interface names can be used as "default identifiers" for interfaces and links, because there is, by default, a one-to-one mapping between interfaces and each of those scopes as described in Section 6.

An implementation could also use interface names as <zone_id> for larger scopes than links, but there might be some confusion in such use. For example, when more than one interface belongs to a same (multicast) site, a user would be confused about which interface should be used. Also, a mapping function from an address to a name would encounter a same kind of problem, when it prints an address with an interface name as a zone index. This document does not specify how these cases should be treated and leaves it implementation dependent.

It cannot be assumed that a same index is common to all nodes in a zone (see Section 6). Hence, the format MUST be used only within a node and MUST NOT be sent on a wire unless every node that interprets the format agrees on the semantics.

11.3 Examples

Here are examples. The following addresses

```
          fe80::1234 (on the 1st link of the node)
          ff02::5678 (on the 5th link of the node)
          ff08::9abc (on the 10th organization of the node)
```

would be represented as follows:

```
          fe80::1234%1
          ff02::5678%5
          ff08::9abc%10
```

(Here we assume a natual translation from a zone index to the <zone_id> part where the Nth zone of any scope is translated into "N".)

If we use interface names as <zone_id>, those addresses could also be represented as follows:

```
         fe80::1234%ne0
         ff02::5678%pvc1.3
         ff08::9abc%interface10
```

where the interface "ne0" belongs to the 1st link, "pvc1.3" belongs to the 5th link, and "interface10" belongs to the 10th organization.

11.4 Usage Examples

Applications that are supposed to be used in end hosts like telnet, ftp, and ssh, may not explicitly support the notion of address scope, especially of link-local addresses. However, an expert user (e.g. a network administrator) sometimes has to give even link-local addresses to such applications.

Here is a concrete example. Consider a multi-linked router, called "R1", that has at least two point-to-point interfaces (links). Each of the interfaces is connected to another router, called "R2" and "R3", respectively. Also assume that the point-to-point interfaces have link-local addresses only.

Now suppose that the routing system on R2 hangs up and has to be reinvoked. In this situation, we may not be able to use a global address of R2, because this is a routing trouble and we cannot expect that we have enough routes for global reachability to R2.

Hence, we have to login R1 first, and then try to login R2 using link-local addresses. In such a case, we have to give the link-local address of R2 to, for example, telnet. Here we assume the address is fe80::2.

Note that we cannot just type like

```
         % telnet fe80::2
```

here, since R1 has more than one link and hence the telnet command cannot detect which link it should try to use for connecting. Instead, we should type the link-local address with the link index as follows:

```
         % telnet fe80::2%3
```

where "3" after the delimiter character `%' conrresponds to the link index of the point-to-point link.

11.5 Related API

An extension to the recommended basic API defines how the format for non-global addresses should be treated in library functions that translate a nodename to an address, or vice versa [11].

11.6 Omitting Zone Indices

The format defined in this document does not intend to invalidate the original format for non-global addresses, that is, the format without the zone index portion. As described in Section 6, in some common cases with the notion of the default zone ID, there can be no ambiguity about scope zones. In such an environment, the implementation can omit the "%<zone_id>" part, and, as a result, it can act as if it did not support the extended format at all.

11.7 Combinations of Delimiter Characters

There are other kinds of delimiter characters defined for IPv6 addresses. In this subsection, we describe how they should be combined with the format for non-global addresses.

The IPv6 addressing architecture [1] also defines the syntax of IPv6 prefixes. If the address portion of a prefix is non-global and its scope zone should be disambiguated, the address portion SHOULD be in the format. For example, a link-local prefix fe80::/64 on the 2nd link can be represented as follows:

```
         fe80::%2/64
```

In this combination, it is important to place the zone index portion before the prefix length, when we consider parsing the format by a name-to-address library function [11]. That is, we can first separate the address with the zone index from the prefix length, and just pass the former to the library function.

The preferred format for literal IPv6 addresses in URL's are also defined [12]. When a user types the preferred format for an IPv6 non-global address whose zone should be explicitly specified, the user could use the format for the non-global address combined with the preferred format.

However, the typed URL is often sent on a wire, and it would cause confusion if an application did not strip the <zone_id> portion before sending. Note that the applications should not need to care

about which kind of addresses they're using, much less parse or strip out the <zone_id> portion of the address. Also, the format for non-global addresses might conflict with the URI syntax [13], since the syntax defines the delimiter character (`%') as the escape character.

Hence, this document does not specify how the format for non-global addresses should be combined with the preferred format for literal IPv6 addresses. As for the conflict issue with the URI format, it would be better to wait until the relationship between the preferred format and the URI syntax is clarified. In fact, the preferred format for IPv6 literal addresses itself has same kind of conflict. In any case, it is recommended to use an FQDN instead of a literal IPv6 address in a URL, whenever an FQDN is available.

12. Security Considerations

The routing section of this document specifies a set of guidelines that allow routers to prevent zone-specific information from leaking out of each zone. If, for example, multicast site boundary routers allow site routing information to be forwarded outside of the site, the integrity of the site could be compromised.

Since the use of the textual representation of non-global addresses is restricted within a single node, it does not create a security vulnerability from outside the node. However, a malicious node might send a packet that contains a textual IPv6 non-global address with a zone index, intending to deceive the receiving node about the zone of the non-global address. Thus, an implementation should be careful when it receives packets that contain textual non-global addresses as data.

13. Contributors

This document is a combination of several separate efforts. Atsushi Onoe took a significant role in one of them, and deeply contributed to the content of Section 11 as a co-author of a separate proposal.

14. Acknowledgements

Many members the IPv6 working group provided useful comments and feedback on this document. In particular, Margaret Wasserman and Bob Hinden led the working group to make a consensus on IPv6 local addressing. Richard Draves proposed an additional rule to process Routing header containing scoped addresses. Dave Thaler and Francis Dupont gave valuable suggestions to define semantics of zone indices in terms of related API. Pekka Savola reviewed a draft of the document very carefully, and made detailed comments including serious

problems.

Normative References

[1] Hinden, R. and S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture", RFC 3513, April 2003.

[2] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, BCP 14, March 1999.

[3] Deering, S. and R. Hinden, "Internet Protocol Version 6 (IPv6) Specification", RFC 2460, December 1998.

[4] Conta, A. and S. Deering, "Internet Control Message (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification", RFC 2463, December 1998.

Informative References

[5] Huitema, C. and B. Carpenter, "Deprecating Site Local Addresses", draft-ietf-ipv6-deprecate-site-local-02.txt (work in progress), November 2003.

[6] Draves, R., "Default Address Selection for Internet Protocol version 6 (IPv6)", RFC 3484, February 2003.

[7] Cheshire, S., Aboba, B. and E. Guttman, "Dynamic Configuration of Link-Local IPv4 Addresses", draft-ietf-zeroconf-ipv4-linklocal-12.txt (work in progress), January 2004.

[8] Conta, A., "Generic Packet Tunneling in IPv6 Specification", RFC 2473, December 1998.

[9] Narten, T., "Neighbor Discovery for IP Version 6 (IPv6)", RFC 2461, December 1998.

[10] Gilligan, R., Thomson, S., Bound, J., McCann, J. and W. Stevens, "Basic Socket Interface Extensions for IPv6", RFC 3493, February 2003.

[11] Gilligan, R., Thomson, S., Bound, J., McCann, J. and W. Stevens, "Scoped Address Extensions to the IPv6 Basic Socket API", Internet-Draft <draft-ietf-ipv6-scope-api-00.txt, July 2002.

[12] Hinden, R., Carpenter, B. and L. Masinter, "Preferred Format for Literal IPv6 Addresses in URL's", RFC 2732, December 1999.

[13] Berners-Lee, T., Fielding, R. and L. Masinter, "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998.

Authors' Addresses

Stephen E. Deering
Cisco Systems, Inc.
170 West Tasman Drive
San Jose, CA 95134-1706
USA

Brian Haberman
Caspian Networks
1 Park Drive, Suite 300
Research Triangle Park, NC 27709
USA

Phone: +1-919-949-4828
EMail: brian@innovationslab.net

Tatuya Jinmei
Corporate Research & Development Center, Toshiba Corporation
1 Komukai Toshiba-cho, Saiwai-ku
Kawasaki-shi, Kanagawa 212-8582
Japan

Phone: +81-44-549-2230
Fax: +81-44-520-1841
EMail: jinmei@isl.rdc.toshiba.co.jp

Erik Nordmark
Sun Microsystems Laboratories, Europe
180, avenue de l'Europe
SAINT ISMIER Cedex 38334
France

Phone: +33 (0)4 74 18 88 03
Fax: +33 (0)4 76 18 88 88
EMail: Erik.Nordmark@sun.com

Brian D. Zill
Microsoft Research
One Microsoft Way
Redmond, WA 98052-6399
USA

Phone: +1-425-703-3568
Fax: +1-425-936-7329
EMail: bzill@microsoft.com

Intellectual Property Statement

The IETF takes no position regarding the validity or scope of any intellectual property or other rights that might be claimed to pertain to the implementation or use of the technology described in this document or the extent to which any license under such rights might or might not be available; neither does it represent that it has made any effort to identify any such rights. Information on the IETF's procedures with respect to rights in standards-track and standards-related documentation can be found in BCP-11. Copies of claims of rights made available for publication and any assurances of licenses to be made available, or the result of an attempt made to obtain a general license or permission for the use of such proprietary rights by implementors or users of this specification can be obtained from the IETF Secretariat.

The IETF invites any interested party to bring to its attention any copyrights, patents or patent applications, or other proprietary rights which may cover technology that may be required to practice this standard. Please address the information to the IETF Executive Director.

Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.