IPng Working Group
Internet Draft
Document: draft-draves-ipngwg-simple-srcaddr-00.txt
Category: Standards Track

R. Draves
Microsoft Research

April 28, 1999

Simple Source Address Selection for IPv6

Status of this Memo

1. Abstract

This document describes a simple algorithm by which IPv6 implementations can choose an appropriate source address to use for communication with a specified destination address.

2. Conventions used in this document

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 RFC-2119 [2].

3. Introduction

The IPv6 addressing architecture [3] allows multiple addresses to be

assigned to interfaces. These addresses may have different reachability scopes (link-local, site-local, or global). Furthermore, addresses assigned via IPv6's auto-configuration mechanisms [4] may be "preferred" or "deprecated".

On occasion, an IPv6 implementation must choose from a set of available addresses an appropriate source address to use for a given destination address. This document specifies a simple set of rules for choosing a source address of appropriate scope and configuration status (preferred or deprecated). Furthermore, this document suggests a preferred method, longest matching prefix, for choosing among otherwise equivalent source addresses in the absence of better information.

This document does not address the more general problem of choosing the "best" destination address / source address pair for communication with another node, given a set of possible destination addresses and a set of possible source addresses.

This document does not specify a "strong host" or "weak host" model for source address selection [5, section 3.3.4.2]. It merely assumes that the implementation has a set of candidate source addresses from which one must be chosen. If the implementation uses the strong host model, this MAY be the set of addresses assigned to the outgoing interface that will be used for the destination address. If the implementation uses the weak host model, this MAY be the set of all addresses assigned to the node's interfaces.

The rules specified in this document MUST NOT be construed to override an application's explicit choice of source address.

4. Source Address Selection

This document specifies a pair-wise source address selection algorithm. Given a destination address and a pair of possible source addresses SA and SB (SA not equal to SB), it chooses a source address. Obviously, any pair-wise algorithm may be extended to select an address from a set of candidate source addresses.

When comparing scopes, we say that link-local scope is smaller than site-local scope, which is smaller than global scope.

Addresses that are manually configured (or otherwise not auto-configured according to [4]), we treat as having "preferred" configuration status.

The algorithm consists of four rules, which MUST be applied in order. If a rule chooses a source address, then the remaining rules are not relevant and MUST be ignored. Subsequent rules act as tie-breakers for earlier rules. If the four rules fail to choose a source address, some unspecified tie-breaker MUST be used.

Rule 1: If one of the source addresses is equal to the destination address, an implementation MUST choose that source address.

Rule 2: If the source addresses SA and SB have different scope, then an implementation MUST choose the source address as follows. Without loss of generality, assume that SA has smaller scope than SB. If the

destination address scope is smaller than or equal to SA's scope, then choose SA. Otherwise choose SB.

Rule 3: The two source addresses have the same scope. If one of the source addresses is "preferred" and one of them is "deprecated", an implementation MUST choose the one that is preferred.

Rule 4: The two source addresses have the same scope and the same configuration status (both preferred or both deprecated). If one of the source addresses has a longer prefix matching the destination address, an implementation SHOULD choose the source address with the longer matching prefix.

The fourth rule MAY be superceded if the implementation has other means of choosing among source addresses. For example, if the implementation somehow knows which source address will result in the "best" communications performance.

5. Multicast Destination Addresses

Multicast destination addresses have a 4-bit scope field that controls the propagation of the multicast packet. The IPv6 addressing architecture defines scope field values for node-local (0x1), link-local (0x2), site-local (0x5), organization-local (0x8), and global (0xE) scopes.

Application of the source address selection algorithm to a multicast destination address requires the comparison of a unicast source address scope with a multicast destination address scope. We map unicast link-local to multicast link-local, unicast site-local to multicast site-local, and unicast global scope to multicast global scope. This mapping implicitly conflates unicast site boundaries and multicast site boundaries.

6. IPv4-Compatible Addresses

For the purposes of this document, IPv4-compatible addresses have global scope and "preferred" configuration status.

7. Other Format Prefixes

This document does not specify source address selection in the presence of NSAP addresses, IPX addresses, or addresses with as-yet-undefined format prefixes.

5. Security Considerations

This document has no direct impact on Internet infrastructure security.

6. References

1 S. Bradner, "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

2 S. Bradner, "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

3 R. Hinden, S. Deering, "IP Version 6 Addressing Architecture", RFC 2373, July 1998.

4 S. Thompson, T. Narten, "IPv6 Stateless Address Autoconfiguration", RFC 2462 , December 1998.

5 R. Braden, editor, "Requirements for Internet Hosts - Communication Layers", RFC 1122 , October 1989.

7. Acknowledgments

The author would like to acknowledge the contributions of the IPng Working Group, and in particular, Robert Elz and Jack McCann.

8. Author's Address

Richard Draves
Microsoft Research
One Microsoft Way
Redmond, WA 98052
Email: richdr@microsoft.com

Full Copyright Statement