Network Working Group
Request for Comments: 2461
Obsoletes: 1970
Category: Standards Track

T. Narten
IBM
E. Nordmark
Sun Microsystems
W. Simpson
Daydreamer

December 1998

# Neighbor Discovery for IP Version 6 (IPv6)

## Status of this Memo

This document specifies an Internet standards track protocol for the Internet community, and requests discussion and suggestions for improvements. Please refer to the current edition of the "Internet Official Protocol Standards" (STD 1) for the standardization state and status of this protocol. Distribution of this memo is unlimited.

## Copyright Notice

Copyright (C) The Internet Society (1998). All Rights Reserved.

## Abstract

This document specifies the Neighbor Discovery protocol for IP Version 6. IPv6 nodes on the same link use Neighbor Discovery to discover each other's presence, to determine each other's link-layer addresses, to find routers and to maintain reachability information about the paths to active neighbors.

Table of Contents

1. INTRODUCTION..... 3
2. TERMINOLOGY..... 4
2.1. General..... 4
2.2. Link Types..... 7
2.3. Addresses..... 8
2.4. Requirements..... 9
3. PROTOCOL OVERVIEW..... 9
3.1. Comparison with IPv4..... 13
3.2. Supported Link Types..... 15
4. MESSAGE FORMATS..... 17
4.1. Router Solicitation Message Format..... 17
4.2. Router Advertisement Message Format..... 18
4.3. Neighbor Solicitation Message Format..... 21
4.4. Neighbor Advertisement Message Format..... 23
4.5. Redirect Message Format..... 26

4.6. Option Formats..... 28
4.6.1. Source/Target Link-layer Address..... 28
4.6.2. Prefix Information..... 29
4.6.3. Redirected Header..... 31
4.6.4. MTU..... 32
5. CONCEPTUAL MODEL OF A HOST..... 33
5.1. Conceptual Data Structures..... 33
5.2. Conceptual Sending Algorithm..... 35
5.3. Garbage Collection and Timeout Requirements..... 37
6. ROUTER AND PREFIX DISCOVERY..... 37
6.1. Message Validation..... 38
6.1.1. Validation of Router Solicitation Messages..... 38
6.1.2. Validation of Router Advertisement Messages..... 39
6.2. Router Specification..... 40
6.2.1. Router Configuration Variables..... 40
6.2.2. Becoming An Advertising Interface..... 44
6.2.3. Router Advertisement Message Content..... 44
6.2.4. Sending Unsolicited Router Advertisements..... 46
6.2.5. Ceasing To Be An Advertising Interface..... 46
6.2.6. Processing Router Solicitations..... 47
6.2.7. Router Advertisement Consistency..... 48
6.2.8. Link-local Address Change..... 49
6.3. Host Specification..... 50
6.3.1. Host Configuration Variables..... 50
6.3.2. Host Variables..... 50
6.3.3. Interface Initialization..... 51
6.3.4. Processing Received Router Advertisements..... 51
6.3.5. Timing out Prefixes and Default Routers..... 54
6.3.6. Default Router Selection..... 54
6.3.7. Sending Router Solicitations..... 55
7. ADDRESS RESOLUTION AND NEIGHBOR UNREACHABILITY DETECTION. 56
7.1. Message Validation..... 57
7.1.1. Validation of Neighbor Solicitations..... 57
7.1.2. Validation of Neighbor Advertisements..... 58
7.2. Address Resolution..... 58
7.2.1. Interface Initialization..... 59
7.2.2. Sending Neighbor Solicitations..... 59
7.2.3. Receipt of Neighbor Solicitations..... 60
7.2.4. Sending Solicited Neighbor Advertisements..... 61
7.2.5. Receipt of Neighbor Advertisements..... 62
7.2.6. Sending Unsolicited Neighbor Advertisements..... 64
7.2.7. Anycast Neighbor Advertisements..... 65
7.2.8. Proxy Neighbor Advertisements..... 65
7.3. Neighbor Unreachability Detection..... 66
7.3.1. Reachability Confirmation..... 66
7.3.2. Neighbor Cache Entry States..... 67
7.3.3. Node Behavior..... 68
8. REDIRECT FUNCTION..... 70

8.1. Validation of Redirect Messages..... 71
8.2. Router Specification..... 72
8.3. Host Specification..... 73
9. EXTENSIBILITY - OPTION PROCESSING..... 74
10. PROTOCOL CONSTANTS..... 75
11. SECURITY CONSIDERATIONS..... 76
12. RENUMBERING CONSIDERATIONS..... 78

References..... 80
Authors' Addresses..... 81
Appendix A: Multihomed Hosts..... 82
Appendix B: Future Extensions..... 84
Appendix C: State Machine for the Reachability State..... 85
Appendix D: Summary of ISROUTER Rules..... 88
Appendix E: Implementation Issues..... 89
Appendix E.1: Reachability confirmations..... 89
Appendix F: Changes since RFC 1970..... 91
Full Copyright Statement..... 93

## 1. INTRODUCTION

This specification defines the Neighbor Discovery (ND) protocol for Internet Protocol Version 6 (IPv6). Nodes (hosts and routers) use Neighbor Discovery to determine the link-layer addresses for neighbors known to reside on attached links and to quickly purge cached values that become invalid. Hosts also use Neighbor Discovery to find neighboring routers that are willing to forward packets on their behalf. Finally, nodes use the protocol to actively keep track of which neighbors are reachable and which are not, and to detect changed link-layer addresses. When a router or the path to a router fails, a host actively searches for functioning alternates.

Unless specified otherwise (in a document that covers operating IP over a particular link type) this document applies to all link types. However, because ND uses link-layer multicast for some of its services, it is possible that on some link types (e.g., NBMA links) alternative protocols or mechanisms to implement those services will be specified (in the appropriate document covering the operation of IP over a particular link type). The services described in this document that are not directly dependent on multicast, such as Redirects, Next-hop determination, Neighbor Unreachability Detection, etc., are expected to be provided as specified in this document. The details of how one uses ND on NBMA links is an area for further study.

The authors would like to acknowledge the contributions of the IPNGWG working group and, in particular, (in alphabetical order) Ran Atkinson, Jim Bound, Scott Bradner, Alex Conta, Stephen Deering,

Richard Draves, Francis Dupont, Robert Elz, Robert Gilligan, Robert Hinden, Allison Mankin, Dan McDonald, Charles Perkins, Matt Thomas, and Susan Thomson.

## 2. TERMINOLOGY

### 2.1. General

IP - Internet Protocol Version 6. The terms IPv4 and IPv6 are used only in contexts where necessary to avoid ambiguity.

ICMP - Internet Message Control Protocol for the Internet Protocol Version 6. The terms ICMPv4 and ICMPv6 are used only in contexts where necessary to avoid ambiguity.

node - a device that implements IP.

router - a node that forwards IP packets not explicitly addressed to itself.

host - any node that is not a router.

upper layer - a protocol layer immediately above IP. Examples are transport protocols such as TCP and UDP, control protocols such as ICMP, routing protocols such as OSPF, and internet or lower-layer protocols being "tunneled" over (i.e., encapsulated in) IP such as IPX, AppleTalk, or IP itself.

link - a communication facility or medium over which nodes can communicate at the link layer, i.e., the layer immediately below IP. Examples are Ethernets (simple or bridged), PPP links, X.25, Frame Relay, or ATM networks as well as internet (or higher) layer "tunnels", such as tunnels over IPv4 or IPv6 itself.

interface - a node’s attachment to a link.

neighbors - nodes attached to the same link.

address - an IP-layer identifier for an interface or a set of interfaces.

anycast address - 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 protocol's measure of distance). See [ADDR-ARCH].

Note that an anycast address is syntactically indistinguishable from a unicast address. Thus, nodes sending packets to anycast addresses don't generally know that an anycast address is being used. Throughout the rest of this document, references to unicast addresses also apply to anycast addresses in those cases where the node is unaware that a unicast address is actually an anycast address.

prefix - a bit string that consists of some number of initial bits of an address.

link-layer address
- a link-layer identifier for an interface. Examples include IEEE 802 addresses for Ethernet links and E.164 addresses for ISDN links.

on-link - an address that is assigned to an interface on a specified link. A node considers an address to be on-link if:

- it is covered by one of the link's prefixes, or
- a neighboring router specifies the address as the target of a Redirect message, or
- a Neighbor Advertisement message is received for the (target) address, or
- any Neighbor Discovery message is received from the address.

off-link - the opposite of "on-link"; an address that is not assigned to any interfaces on the specified link.

longest prefix match
- The process of determining which prefix (if any) in a set of prefixes covers a target address. A target address is covered by a prefix if all of the bits in the prefix match the left-most bits of the target address. When multiple prefixes cover an address, the longest prefix is the one that matches.

reachability
- whether or not the one-way "forward" path to a neighbor is functioning properly. In particular, whether packets sent to a neighbor are reaching the IP layer on the neighboring machine and are being processed properly by the receiving IP layer. For neighboring routers, reachability means that packets sent by a node's IP layer are delivered to the router's IP layer, and the router is indeed forwarding packets (i.e., it is configured as a router, not a host). For hosts, reachability means that packets sent by a node's IP layer are delivered to the neighbor host's IP layer.

packet - an IP header plus payload.

link MTU - the maximum transmission unit, i.e., maximum packet size in octets, that can be conveyed in one piece over a link.

target - an address about which address resolution information is sought, or an address which is the new first-hop when being redirected.

proxy - a router that responds to Neighbor Discovery query messages on behalf of another node. A router acting on behalf of a mobile node that has moved off-link could potentially act as a proxy for the mobile node.

ICMP destination unreachable indication
- an error indication returned to the original sender of a packet that cannot be delivered for the reasons outlined in [ICMPv6]. If the error occurs on a node other than the node originating the packet, an ICMP error message is generated. If the error occurs on the originating node, an implementation is not required to actually create and send an ICMP error packet to the source, as long as the upper-layer sender is notified through an appropriate mechanism (e.g., return value from a procedure call). Note, however, that an implementation may find it convenient in some cases to return errors to the sender by taking the offending packet, generating an ICMP error message, and then delivering it (locally) through the generic error handling routines.

random delay
- when sending out messages, it is sometimes necessary to delay a transmission for a random amount of time in order to prevent multiple nodes from transmitting at exactly the same time, or to prevent long-range periodic transmissions from synchronizing with each other [SYNC]. When a random component is required, a node calculates the actual delay in such a way that the computed delay forms a uniformly-distributed random value that falls between the specified minimum and maximum delay times. The implementor must take care to insure that the granularity of the calculated random component and the resolution of the timer used are both high enough to insure that the probability of multiple nodes delaying the same amount of time is small.

random delay seed
- If a pseudo-random number generator is used in calculating a random delay component, the generator should be initialized with a unique seed prior to being used. Note that it is not sufficient to use the interface token alone as the seed, since interface tokens will not always be unique. To reduce the probability that duplicate interface tokens cause the same seed to be used, the seed should be calculated from a variety of input sources (e.g., machine components) that are likely to be different even on identical "boxes". For example, the seed could be formed by combining the CPU's serial number with an interface token.

## 2.2. Link Types

Different link layers have different properties. The ones of concern to Neighbor Discovery are:

multicast - a link that supports a native mechanism at the link layer for sending packets to all (i.e., broadcast) or a subset of all neighbors.

point-to-point - a link that connects exactly two interfaces. A point-to-point link is assumed to have multicast capability and have a link-local address.

non-broadcast multi-access (NBMA)
- a link to which more than two interfaces can attach, but that does not support a native form of multicast or broadcast (e.g., X.25, ATM, frame relay, etc.).

Note that all link types (including NBMA) are expected to provide multicast service for IP (e.g., using multicast servers), but it is an issue for further study whether ND should use such facilities or an alternate mechanism that provides the equivalent ND services.

shared media - a link that allows direct communication among a number of nodes, but attached nodes are configured in such a way that they do not have complete prefix information for all on-link destinations. That is, at the IP level, nodes on the same link may not know that they are neighbors; by default, they communicate through a router. Examples are large (switched) public data networks such as SMDS and B-ISDN. Also known as "large clouds". See [SH-MEDIA].

variable MTU - a link that does not have a well-defined MTU (e.g., IEEE 802.5 token rings). Many links (e.g., Ethernet) have a standard MTU defined by the link-layer protocol or by the specific document describing how to run IP over the link layer.

asymmetric reachability
- a link where non-reflexive and/or non-transitive reachability is part of normal operation. (Non-reflexive reachability means packets from A reach B but packets from B don't reach A. Non-transitive reachability means packets from A reach B, and packets from B reach C, but packets from A don't reach C.) Many radio links exhibit these properties.

## 2.3. Addresses

Neighbor Discovery makes use of a number of different addresses defined in [ADDR-ARCH], including:

all-nodes multicast address
- the link-local scope address to reach all nodes. FF02::1

all-routers multicast address
- the link-local scope address to reach all routers. FF02::2

solicited-node multicast address
- a link-local scope multicast address that is computed as a function of the solicited target's address. The function is described in [ADDR-ARCH]. The function is chosen so that IP addresses which differ only in the high-order bits, e.g., due to multiple high-order prefixes associated with different providers, will map to the same solicited-node address thereby reducing the number of multicast addresses a node must join.

link-local address
- a unicast address having link-only scope that can be used to reach neighbors. All interfaces on routers MUST have a link-local address. Also, [ADDRCONF] requires that interfaces on hosts have a link-local address.

unspecified address
- a reserved address value that indicates the lack of an address (e.g., the address is unknown). It is never used as a destination address, but may be used as a source address if the sender does not (yet) know its own address (e.g., while verifying an address is unused during address autoconfiguration [ADDRCONF]). The unspecified address has a value of 0:0:0:0:0:0:0:0.

## 2.4. Requirements

The keywords MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL, when they appear in this document, are to be interpreted as described in [KEYWORDS].

This document also makes use of internal conceptual variables to describe protocol behavior and external variables that an implementation must allow system administrators to change. The specific variable names, how their values change, and how their settings influence protocol behavior are provided to demonstrate protocol behavior. An implementation is not required to have them in the exact form described here, so long as its external behavior is consistent with that described in this document.

# 3. PROTOCOL OVERVIEW

This protocol solves a set of problems related to the interaction between nodes attached to the same link. It defines mechanisms for solving each of the following problems:

Router Discovery: How hosts locate routers that reside on an attached link.

Prefix Discovery: How hosts discover the set of address prefixes that define which destinations are on-link for an attached link. (Nodes use prefixes to distinguish destinations that reside on-link from those only reachable through a router.)

Parameter Discovery: How a node learns such link parameters as the link MTU or such Internet parameters as the hop limit value to place in outgoing packets.

Address Autoconfiguration: How nodes automatically configure an address for an interface.

Address resolution: How nodes determine the link-layer address of an on-link destination (e.g., a neighbor) given only the destination's IP address.

Next-hop determination: The algorithm for mapping an IP destination address into the IP address of the neighbor to which traffic for the destination should be sent. The next-hop can be a router or the destination itself.

Neighbor Unreachability Detection: How nodes determine that a neighbor is no longer reachable. For neighbors used as routers, alternate default routers can be tried. For both routers and hosts, address resolution can be performed again.

Duplicate Address Detection: How a node determines that an address it wishes to use is not already in use by another node.

Redirect: How a router informs a host of a better first-hop node to reach a particular destination.

Neighbor Discovery defines five different ICMP packet types: A pair of Router Solicitation and Router Advertisement messages, a pair of Neighbor Solicitation and Neighbor Advertisements messages, and a Redirect message. The messages serve the following purpose:

Router Solicitation: When an interface becomes enabled, hosts may send out Router Solicitations that request routers to generate Router Advertisements immediately rather than at their next scheduled time.

Router Advertisement: Routers advertise their presence together with various link and Internet parameters either periodically, or in response to a Router Solicitation message. Router Advertisements contain prefixes that are used for on-link determination and/or address configuration, a suggested hop limit value, etc.

Neighbor Solicitation: Sent by a node to determine the link-layer address of a neighbor, or to verify that a neighbor is still reachable via a cached link-layer address. Neighbor Solicitations are also used for Duplicate Address Detection.

Neighbor Advertisement: A response to a Neighbor Solicitation message. A node may also send unsolicited Neighbor Advertisements to announce a link-layer address change.

Redirect: Used by routers to inform hosts of a better first hop for a destination.

On multicast-capable links, each router periodically multicasts a Router Advertisement packet announcing its availability. A host receives Router Advertisements from all routers, building a list of default routers. Routers generate Router Advertisements frequently enough that hosts will learn of their presence within a few minutes, but not frequently enough to rely on an absence of advertisements to detect router failure; a separate Neighbor Unreachability Detection algorithm provides failure detection.

Router Advertisements contain a list of prefixes used for on-link determination and/or autonomous address configuration; flags associated with the prefixes specify the intended uses of a particular prefix. Hosts use the advertised on-link prefixes to build and maintain a list that is used in deciding when a packet's destination is on-link or beyond a router. Note that a destination can be on-link even though it is not covered by any advertised on-link prefix. In such cases a router can send a Redirect informing the sender that the destination is a neighbor.

Router Advertisements (and per-prefix flags) allow routers to inform hosts how to perform Address Autoconfiguration. For example, routers can specify whether hosts should use stateful (DHCPv6) and/or autonomous (stateless) address configuration. The exact semantics and usage of the address configuration-related information is specified in [ADDRCONF].

Router Advertisement messages also contain Internet parameters such as the hop limit that hosts should use in outgoing packets and,

optionally, link parameters such as the link MTU. This facilitates centralized administration of critical parameters that can be set on routers and automatically propagated to all attached hosts.

Nodes accomplish address resolution by multicasting a Neighbor Solicitation that asks the target node to return its link-layer address. Neighbor Solicitation messages are multicast to the solicited-node multicast address of the target address. The target returns its link-layer address in a unicast Neighbor Advertisement message. A single request-response pair of packets is sufficient for both the initiator and the target to resolve each other's link-layer addresses; the initiator includes its link-layer address in the Neighbor Solicitation.

Neighbor Solicitation messages can also be used to determine if more than one node has been assigned the same unicast address. The use of Neighbor Solicitation messages for Duplicate Address Detection is specified in [ADDRCONF].

Neighbor Unreachability Detection detects the failure of a neighbor or the failure of the forward path to the neighbor. Doing so requires positive confirmation that packets sent to a neighbor are actually reaching that neighbor and being processed properly by its IP layer. Neighbor Unreachability Detection uses confirmation from two sources. When possible, upper-layer protocols provide a positive confirmation that a connection is making "forward progress", that is, previously sent data is known to have been delivered correctly (e.g., new acknowledgments were received recently). When positive confirmation is not forthcoming through such "hints", a node sends unicast Neighbor Solicitation messages that solicit Neighbor Advertisements as reachability confirmation from the next hop. To reduce unnecessary network traffic, probe messages are only sent to neighbors to which the node is actively sending packets.

In addition to addressing the above general problems, Neighbor Discovery also handles the following situations:

Link-layer address change - A node that knows its link-layer address has changed can multicast a few (unsolicited) Neighbor Advertisement packets to all nodes to quickly update cached link-layer addresses that have become invalid. Note that the sending of unsolicited advertisements is a performance enhancement only (e.g., unreliable). The Neighbor Unreachability Detection algorithm ensures that all nodes will reliably discover the new address, though the delay may be somewhat longer.

Inbound load balancing - Nodes with replicated interfaces may want to load balance the reception of incoming packets across multiple network interfaces on the same link. Such nodes have multiple link-layer addresses assigned to the same interface. For example, a single network driver could represent multiple network interface cards as a single logical interface having multiple link-layer addresses.

Load balancing is handled by allowing routers to omit the source link-layer address from Router Advertisement packets, thereby forcing neighbors to use Neighbor Solicitation messages to learn link-layer addresses of routers. Returned Neighbor Advertisement messages can then contain link-layer addresses that differ depending on who issued the solicitation.

Anycast addresses - Anycast addresses identify one of a set of nodes providing an equivalent service, and multiple nodes on the same link may be configured to recognize the same Anycast address. Neighbor Discovery handles anycasts by having nodes expect to receive multiple Neighbor Advertisements for the same target. All advertisements for anycast addresses are tagged as being non-Override advertisements. This invokes specific rules to determine which of potentially multiple advertisements should be used.

Proxy advertisements - A router willing to accept packets on behalf of a target address that is unable to respond to Neighbor Solicitations can issue non-Override Neighbor Advertisements. There is currently no specified use of proxy, but proxy advertising could potentially be used to handle cases like mobile nodes that have moved off-link. However, it is not intended as a general mechanism to handle nodes that, e.g., do not implement this protocol.

### 3.1. Comparison with IPv4

The IPv6 Neighbor Discovery protocol corresponds to a combination of the IPv4 protocols ARP [ARP], ICMP Router Discovery [RDISC], and ICMP Redirect [ICMPv4]. In IPv4 there is no generally agreed upon protocol or mechanism for Neighbor Unreachability Detection, although Hosts Requirements [HR-CL] does specify some possible algorithms for Dead Gateway Detection (a subset of the problems Neighbor Unreachability Detection tackles).

The Neighbor Discovery protocol provides a multitude of improvements over the IPv4 set of protocols:

Router Discovery is part of the base protocol set; there is no need for hosts to "snoop" the routing protocols.

Router advertisements carry link-layer addresses; no additional packet exchange is needed to resolve the router's link-layer address.

Router advertisements carry prefixes for a link; there is no need to have a separate mechanism to configure the "netmask".

Router advertisements enable Address Autoconfiguration.

Routers can advertise an MTU for hosts to use on the link, ensuring that all nodes use the same MTU value on links lacking a well-defined MTU.

Address resolution multicasts are "spread" over 4 billion (2^32) multicast addresses greatly reducing address resolution related interrupts on nodes other than the target. Moreover, non-IPv6 machines should not be interrupted at all.

Redirects contain the link-layer address of the new first hop; separate address resolution is not needed upon receiving a redirect.

Multiple prefixes can be associated with the same link. By default, hosts learn all on-link prefixes from Router Advertisements. However, routers may be configured to omit some or all prefixes from Router Advertisements. In such cases hosts assume that destinations are off-link and send traffic to routers. A router can then issue redirects as appropriate.

Unlike IPv4, the recipient of an IPv6 redirect assumes that the new next-hop is on-link. In IPv4, a host ignores redirects specifying a next-hop that is not on-link according to the link's network mask. The IPv6 redirect mechanism is analogous to the XRedirect facility specified in [SH-MEDIA]. It is expected to be useful on non-broadcast and shared media links in which it is undesirable or not possible for nodes to know all prefixes for on-link destinations.

Neighbor Unreachability Detection is part of the base significantly improving the robustness of packet delivery in the presence of failing routers, partially failing or partitioned links and nodes that change their link-layer addresses. For instance, mobile nodes can move off-link without losing any connectivity due to stale ARP caches.

Unlike ARP, Neighbor Discovery detects half-link failures (using Neighbor Unreachability Detection) and avoids sending traffic to neighbors with which two-way connectivity is absent.

Unlike in IPv4 Router Discovery the Router Advertisement messages do not contain a preference field. The preference field is not needed to handle routers of different "stability"; the Neighbor Unreachability Detection will detect dead routers and switch to a working one.

The use of link-local addresses to uniquely identify routers (for Router Advertisement and Redirect messages) makes it possible for hosts to maintain the router associations in the event of the site renumbering to use new global prefixes.

Using the Hop Limit equal to 255 trick Neighbor Discovery is immune to off-link senders that accidentally or intentionally send ND messages. In IPv4 off-link senders can send both ICMP Redirects and Router Advertisement messages.

Placing address resolution at the ICMP layer makes the protocol more media-independent than ARP and makes it possible to use standard IP authentication and security mechanisms as appropriate [IPv6-AUTH, IPv6-ESP].

## 3.2. Supported Link Types

Neighbor Discovery supports links with different properties. In the presence of certain properties only a subset of the ND protocol mechanisms are fully specified in this document:

point-to-point - Neighbor Discovery handles such links just like multicast links. (Multicast can be trivially provided on point to point links, and interfaces can be assigned link-local addresses.) Neighbor Discovery should be implemented as described in this document.

multicast - Neighbor Discovery should be implemented as described in this document.

non-broadcast multiple access (NBMA)
- Redirect, Neighbor Unreachability Detection and next-hop determination should be implemented as described in this document. Address resolution, and the mechanism for delivering Router Solicitations and Advertisements on NBMA links is not specified in this document. Note that if

hosts support manual configuration of a list of default routers, hosts can dynamically acquire the link-layer addresses for their neighbors from Redirect messages.

shared media - The Redirect message is modeled after the XRedirect message in [SH-MEDIA] in order to simplify use of the protocol on shared media links.

This specification does not address shared media issues that only relate to routers, such as:

- How routers exchange reachability information on a shared media link.

- How a router determines the link-layer address of a host, which it needs to send redirect messages to the host.

- How a router determines that it is the first-hop router for a received packet.

The protocol is extensible (through the definition of new options) so that other solutions might be possible in the future.

variable MTU - Neighbor Discovery allows routers to specify a MTU for the link, which all nodes then use. All nodes on a link must use the same MTU (or Maximum Receive Unit) in order for multicast to work properly. Otherwise when multicasting a sender, which can not know which nodes will receive the packet, could not determine a minimum packet size all receivers can process.

asymmetric reachability
- Neighbor Discovery detects the absence of symmetric reachability; a node avoids paths to a neighbor with which it does not have symmetric connectivity.

The Neighbor Unreachability Detection will typically identify such half-links and the node will refrain from using them.

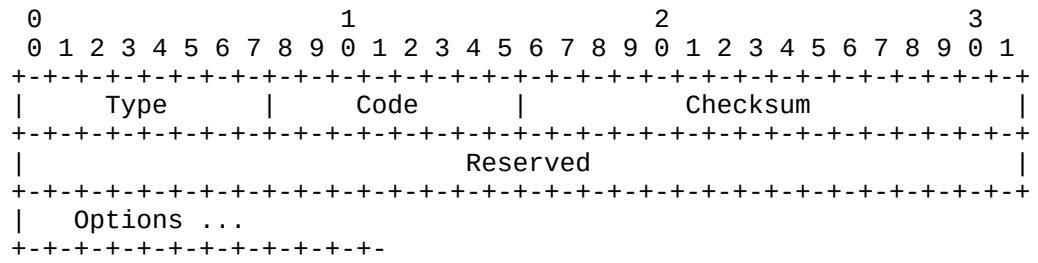
The protocol can presumably be extended in the future to find viable paths in environments that

lack reflexive and transitive connectivity.

# 4. MESSAGE FORMATS

## 4.1. Router Solicitation Message Format

Hosts send Router Solicitations in order to prompt routers to generate Router Advertisements quickly.




IP Fields:

Source Address
: An IP address assigned to the sending interface, or the unspecified address if no address is assigned to the sending interface.

Destination Address
: Typically the all-routers multicast address.

Hop Limit
: 255

Authentication Header
: If a Security Association for the IP Authentication Header exists between the sender and the destination address, then the sender SHOULD include this header.

ICMP Fields:

Type
: 133

Code
: 0

Checksum
: The ICMP checksum. See [ICMPv6].

Reserved
: This field is unused. It MUST be initialized to zero by the sender and MUST be ignored by the receiver.

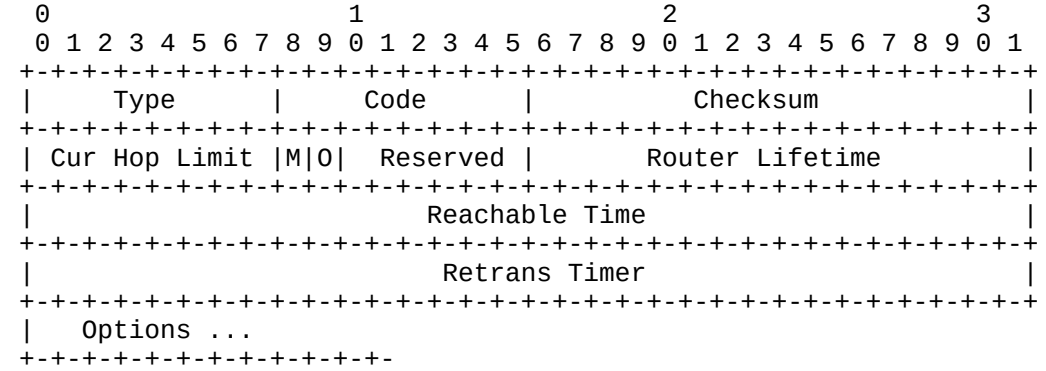
Valid Options:

Source link-layer address
The link-layer address of the sender, if known. MUST NOT be included if the Source Address is the unspecified address. Otherwise it SHOULD be included on link layers that have addresses.

Future versions of this protocol may define new option types. Receivers MUST silently ignore any options they do not recognize and continue processing the message.

## 4.2. Router Advertisement Message Format

Routers send out Router Advertisement message periodically, or in response to a Router Solicitation.




IP Fields:

Source Address
MUST be the link-local address assigned to the interface from which this message is sent.

Destination Address
Typically the Source Address of an invoking Router Solicitation or the all-nodes multicast address.

Hop Limit 255

Authentication Header
If a Security Association for the IP Authentication Header exists between the sender and the destination address, then the sender SHOULD include this header.

ICMP Fields:

| | |
|---|---|
| Type | 134 |
| Code | 0 |
| Checksum | The ICMP checksum. See [ICMPv6]. |
| Cur Hop Limit | 8-bit unsigned integer. The default value that should be placed in the Hop Count field of the IP header for outgoing IP packets. A value of zero means unspecified (by this router). |
| M | 1-bit "Managed address configuration" flag. When set, hosts use the administered (stateful) protocol for address autoconfiguration in addition to any addresses autoconfigured using stateless address autoconfiguration. The use of this flag is described in [ADDRCONF]. |
| O | 1-bit "Other stateful configuration" flag. When set, hosts use the administered (stateful) protocol for autoconfiguration of other (non-address) information. The use of this flag is described in [ADDRCONF]. |
| Reserved | A 6-bit unused field. It MUST be initialized to zero by the sender and MUST be ignored by the receiver. |
| Router Lifetime | 16-bit unsigned integer. The lifetime associated with the default router in units of seconds. The maximum value corresponds to 18.2 hours. A Lifetime of 0 indicates that the router is not a default router and SHOULD NOT appear on the default router list. The Router Lifetime applies only to the router's usefulness as a default router; it does not apply to information contained in other message fields or options. Options that need time limits for their information include their own lifetime fields. |
| Reachable Time | 32-bit unsigned integer. The time, in milliseconds, that a node assumes a neighbor is reachable after having received a reachability confirmation. Used by the Neighbor Unreachability Detection algorithm (see Section 7.3). A value of |

zero means unspecified (by this router).

Retrans Timer 32-bit unsigned integer. The time, in milliseconds, between retransmitted Neighbor Solicitation messages. Used by address resolution and the Neighbor Unreachability Detection algorithm (see Sections 7.2 and 7.3). A value of zero means unspecified (by this router).

Possible options:

Source link-layer address
The link-layer address of the interface from which the Router Advertisement is sent. Only used on link layers that have addresses. A router MAY omit this option in order to enable inbound load sharing across multiple link-layer addresses.

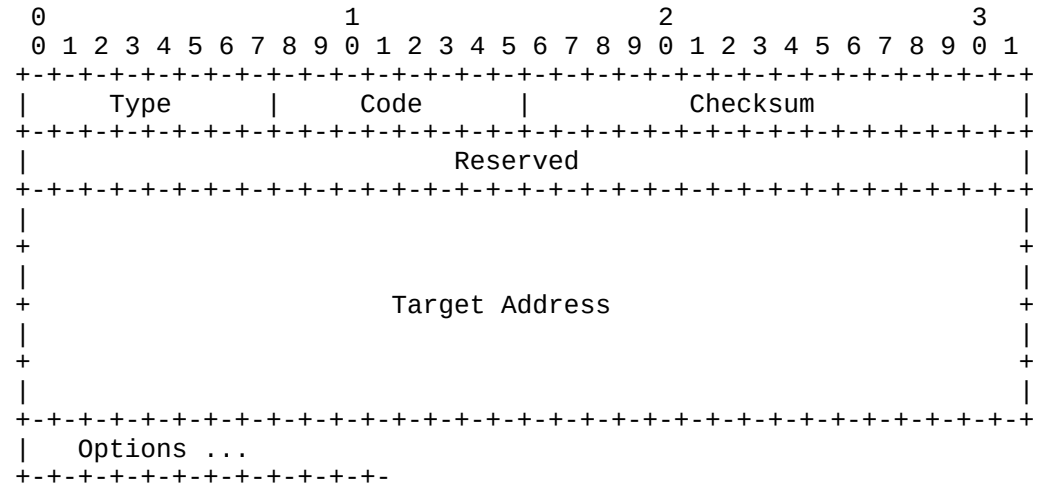
MTU SHOULD be sent on links that have a variable MTU (as specified in the document that describes how to run IP over the particular link type). MAY be sent on other links.

Prefix Information
These options specify the prefixes that are on-link and/or are used for address autoconfiguration. A router SHOULD include all its on-link prefixes (except the link-local prefix) so that multihomed hosts have complete prefix information about on-link destinations for the links to which they attach. If complete information is lacking, a multihomed host may not be able to choose the correct outgoing interface when sending traffic to its neighbors.

Future versions of this protocol may define new option types. Receivers MUST silently ignore any options they do not recognize and continue processing the message.

### 4.3. Neighbor Solicitation Message Format

Nodes send Neighbor Solicitations to request the link-layer address of a target node while also providing their own link-layer address to the target. Neighbor Solicitations are multicast when the node needs to resolve an address and unicast when the node seeks to verify the reachability of a neighbor.




IP Fields:

Source Address
: Either an address assigned to the interface from which this message is sent or (if Duplicate Address Detection is in progress [ADDRCONF]) the unspecified address.

Destination Address
: Either the solicited-node multicast address corresponding to the target address, or the target address.

Hop Limit
: 255

Authentication Header
: If a Security Association for the IP Authentication Header exists between the sender and the destination address, then the sender SHOULD include this header.

ICMP Fields:

Type 135

Code 0

Checksum The ICMP checksum. See [ICMPv6].

Reserved This field is unused. It MUST be initialized to zero by the sender and MUST be ignored by the receiver.

Target Address
The IP address of the target of the solicitation. It MUST NOT be a multicast address.

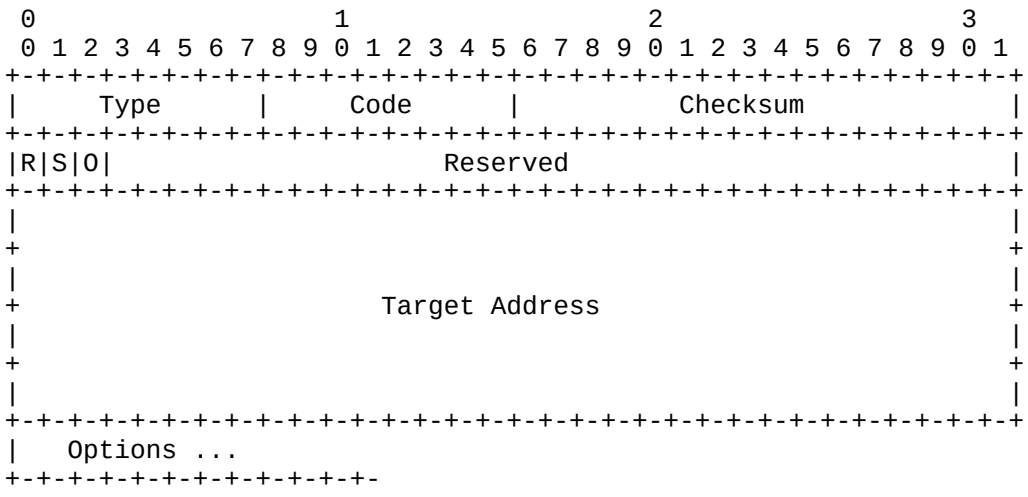
Possible options:

Source link-layer address
The link-layer address for the sender. MUST NOT be included when the source IP address is the unspecified address. Otherwise, on link layers that have addresses this option MUST be included in multicast solicitations and SHOULD be included in unicast solicitations.

Future versions of this protocol may define new option types. Receivers MUST silently ignore any options they do not recognize and continue processing the message.

### 4.4. Neighbor Advertisement Message Format

A node sends Neighbor Advertisements in response to Neighbor Solicitations and sends unsolicited Neighbor Advertisements in order to (unreliably) propagate new information quickly.




IP Fields:

Source Address
: An address assigned to the interface from which the advertisement is sent.

Destination Address
: For solicited advertisements, the Source Address of an invoking Neighbor Solicitation or, if the solicitation's Source Address is the unspecified address, the all-nodes multicast address.

  For unsolicited advertisements typically the all-nodes multicast address.

Hop Limit
: 255

Authentication Header
: If a Security Association for the IP Authentication Header exists between the sender and the destination address, then the sender SHOULD include this header.

ICMP Fields:

Type 136

Code 0

Checksum The ICMP checksum. See [ICMPv6].

R Router flag. When set, the R-bit indicates that the sender is a router. The R-bit is used by Neighbor Unreachability Detection to detect a router that changes to a host.

S Solicited flag. When set, the S-bit indicates that the advertisement was sent in response to a Neighbor Solicitation from the Destination address. The S-bit is used as a reachability confirmation for Neighbor Unreachability Detection. It MUST NOT be set in multicast advertisements or in unsolicited unicast advertisements.

O Override flag. When set, the O-bit indicates that the advertisement should override an existing cache entry and update the cached link-layer address. When it is not set the advertisement will not update a cached link-layer address though it will update an existing Neighbor Cache entry for which no link-layer address is known. It SHOULD NOT be set in solicited advertisements for anycast addresses and in solicited proxy advertisements. It SHOULD be set in other solicited advertisements and in unsolicited advertisements.

Reserved 29-bit unused field. It MUST be initialized to zero by the sender and MUST be ignored by the receiver.

Target Address
For solicited advertisements, the Target Address field in the Neighbor Solicitation message that prompted this advertisement. For an unsolicited advertisement, the address whose link-layer address has changed. The Target Address MUST NOT be a multicast address.

Possible options:

Target link-layer address
: The link-layer address for the target, i.e., the sender of the advertisement. This option MUST be included on link layers that have addresses when responding to multicast solicitations. When responding to a unicast Neighbor Solicitation this option SHOULD be included.

  The option MUST be included for multicast solicitations in order to avoid infinite Neighbor Solicitation "recursion" when the peer node does not have a cache entry to return a Neighbor Advertisements message. When responding to unicast solicitations, the option can be omitted since the sender of the solicitation has the correct link-layer address; otherwise it would not have be able to send the unicast solicitation in the first place. However, including the link-layer address in this case adds little overhead and eliminates a potential race condition where the sender deletes the cached link-layer address prior to receiving a response to a previous solicitation.

Future versions of this protocol may define new option types. Receivers MUST silently ignore any options they do not recognize and continue processing the message.

## 4.5. Redirect Message Format

Routers send Redirect packets to inform a host of a better first-hop node on the path to a destination. Hosts can be redirected to a better first-hop router but can also be informed by a redirect that the destination is in fact a neighbor. The latter is accomplished by setting the ICMP Target Address equal to the ICMP Destination Address.

IP Fields:

Source Address
: MUST be the link-local address assigned to the interface from which this message is sent.

Destination Address
: The Source Address of the packet that triggered the redirect.

Hop Limit
: 255

Authentication Header
If a Security Association for the IP Authentication Header exists between the sender and the destination address, then the sender SHOULD include this header.

ICMP Fields:

Type 137

Code 0

Checksum The ICMP checksum. See [ICMPv6].

Reserved This field is unused. It MUST be initialized to zero by the sender and MUST be ignored by the receiver.

Target Address An IP address that is a better first hop to use for the ICMP Destination Address. When the target is the actual endpoint of communication, i.e., the destination is a neighbor, the Target Address field MUST contain the same value as the ICMP Destination Address field. Otherwise the target is a better first-hop router and the Target Address MUST be the router's link-local address so that hosts can uniquely identify routers.

Destination Address
The IP address of the destination which is redirected to the target.

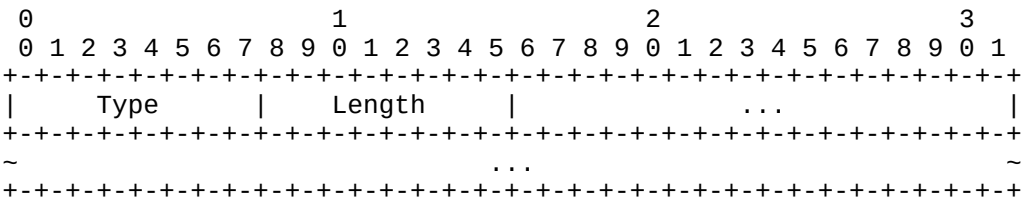
Possible options:

Target link-layer address
The link-layer address for the target. It SHOULD be included (if known). Note that on NBMA links, hosts may rely on the presence of the Target Link-Layer Address option in Redirect messages as the means for determining the link-layer addresses of neighbors. In such cases, the option MUST be included in Redirect messages.

Redirected Header
As much as possible of the IP packet that triggered the sending of the Redirect without making the redirect packet exceed 1280 octets.

### 4.6. Option Formats

Neighbor Discovery messages include zero or more options, some of which may appear multiple times in the same message. All options are of the form:



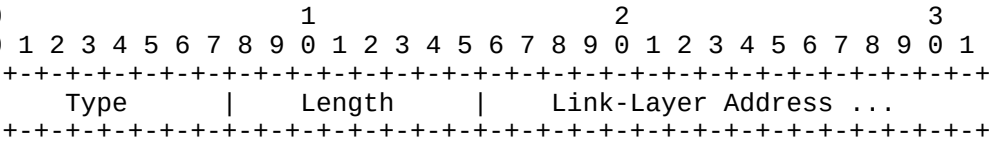

Fields:

Type — 8-bit identifier of the type of option. The options defined in this document are:

| Option Name | Type |
|---|---|
| Source Link-Layer Address | 1 |
| Target Link-Layer Address | 2 |
| Prefix Information | 3 |
| Redirected Header | 4 |
| MTU | 5 |

Length — 8-bit unsigned integer. The length of the option (including the type and length fields) in units of 8 octets. The value 0 is invalid. Nodes MUST silently discard an ND packet that contains an option with length zero.

#### 4.6.1. Source/Target Link-layer Address




Fields:

Type

1 for Source Link-layer Address
2 for Target Link-layer Address

Length The length of the option (including the type and length fields) in units of 8 octets. For example, the length for IEEE 802 addresses is 1 [IPv6-ETHER].

Link-Layer Address
The variable length link-layer address.

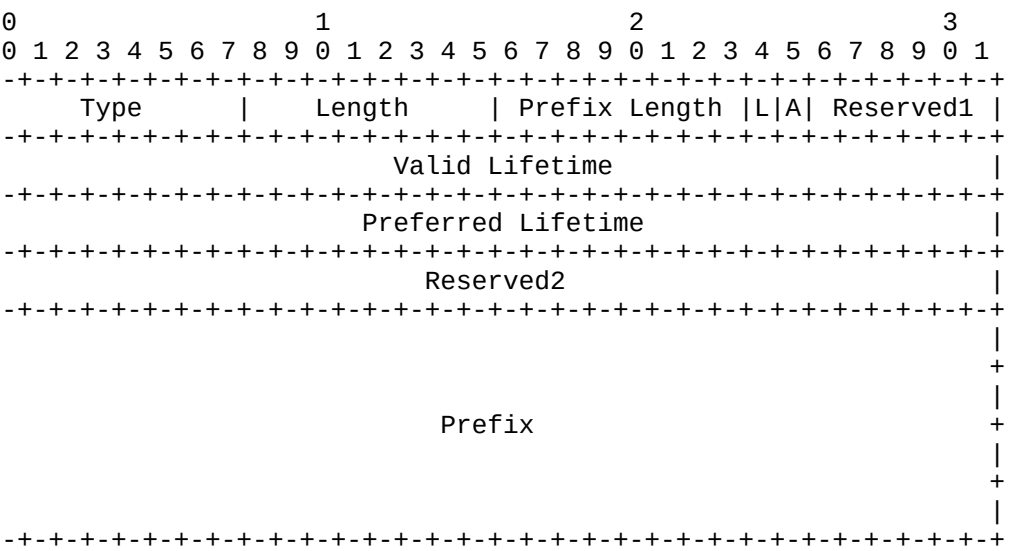
The content and format of this field (including byte and bit ordering) is expected to be specified in specific documents that describe how IPv6 operates over different link layers. For instance, [IPv6-ETHER].

Description
The Source Link-Layer Address option contains the link-layer address of the sender of the packet. It is used in the Neighbor Solicitation, Router Solicitation, and Router Advertisement packets.

The Target Link-Layer Address option contains the link-layer address of the target. It is used in Neighbor Advertisement and Redirect packets.

These options MUST be silently ignored for other Neighbor Discovery messages.

### 4.6.2. Prefix Information



Fields:

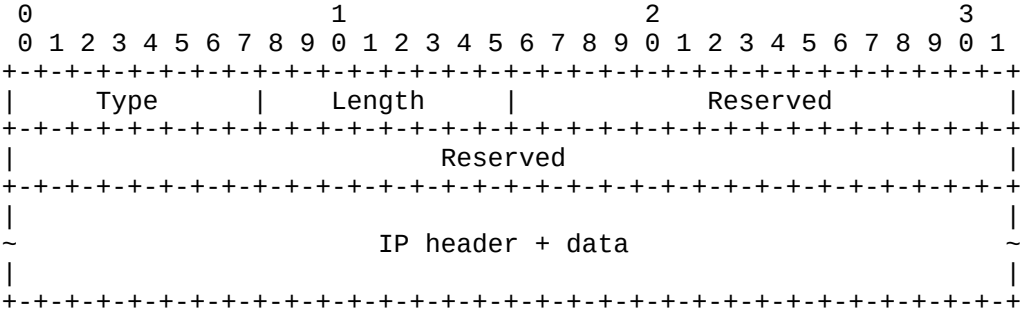
| | |
|---|---|
| Type | 3 |
| Length | 4 |
| Prefix Length | 8-bit unsigned integer. The number of leading bits in the Prefix that are valid. The value ranges from 0 to 128. |
| L | 1-bit on-link flag. When set, indicates that this prefix can be used for on-link determination. When not set the advertisement makes no statement about on-link or off-link properties of the prefix. For instance, the prefix might be used for address configuration with some of the addresses belonging to the prefix being on-link and others being off-link. |
| A | 1-bit autonomous address-configuration flag. When set indicates that this prefix can be used for autonomous address configuration as specified in [ADDRCONF]. |
| Reserved1 | 6-bit unused field. It MUST be initialized to zero by the sender and MUST be ignored by the receiver. |
| Valid Lifetime | 32-bit unsigned integer. The length of time in seconds (relative to the time the packet is sent) that the prefix is valid for the purpose of on-link determination. A value of all one bits (0xffffffff) represents infinity. The Valid Lifetime is also used by [ADDRCONF]. |
| Preferred Lifetime | 32-bit unsigned integer. The length of time in seconds (relative to the time the packet is sent) that addresses generated from the prefix via stateless address autoconfiguration remain preferred [ADDRCONF]. A value of all one bits (0xffffffff) represents infinity. See [ADDRCONF]. |
| Reserved2 | This field is unused. It MUST be initialized to zero by the sender and MUST be ignored by the receiver. |

Prefix An IP address or a prefix of an IP address. The Prefix Length field contains the number of valid leading bits in the prefix. The bits in the prefix after the prefix length are reserved and MUST be initialized to zero by the sender and ignored by the receiver. A router SHOULD NOT send a prefix option for the link-local prefix and a host SHOULD ignore such a prefix option.

Description
The Prefix Information option provide hosts with on-link prefixes and prefixes for Address Autoconfiguration.

The Prefix Information option appears in Router Advertisement packets and MUST be silently ignored for other messages.

### 4.6.3. Redirected Header




Fields:

Type 4

Length The length of the option in units of 8 octets.

Reserved These fields are unused. They MUST be initialized to zero by the sender and MUST be ignored by the receiver.

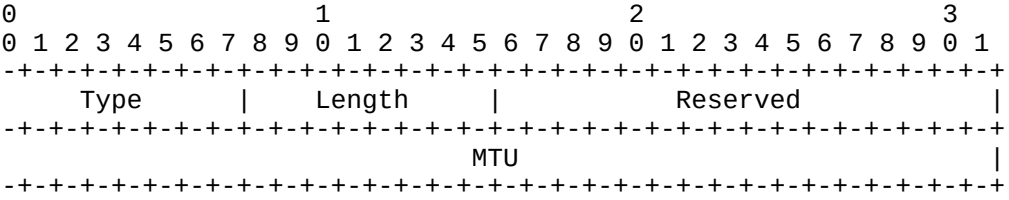
IP header + data
The original packet truncated to ensure that the size of the redirect message does not exceed 1280 octets.

Description

The Redirected Header option is used in Redirect messages and contains all or part of the packet that is being redirected.

This option MUST be silently ignored for other Neighbor Discovery messages.

### 4.6.4. MTU




Fields:

Type 5

Length 1

Reserved This field is unused. It MUST be initialized to zero by the sender and MUST be ignored by the receiver.

MTU 32-bit unsigned integer. The recommended MTU for the link.

Description

The MTU option is used in Router Advertisement messages to insure that all nodes on a link use the same MTU value in those cases where the link MTU is not well known.

This option MUST be silently ignored for other Neighbor Discovery messages.

In configurations in which heterogeneous technologies are bridged together, the maximum supported MTU may differ from one segment to another. If the bridges do not generate ICMP Packet Too Big messages, communicating nodes will be unable to use Path MTU to dynamically determine the appropriate MTU on a per-neighbor basis. In such cases, routers use the MTU option to specify

the maximum MTU value that is supported by all segments.

## 5. CONCEPTUAL MODEL OF A HOST

This section describes a conceptual model of one possible data structure organization that hosts (and to some extent routers) will maintain in interacting with neighboring nodes. The described organization is provided to facilitate the explanation of how the Neighbor Discovery protocol should behave. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this document.

This model is only concerned with the aspects of host behavior directly related to Neighbor Discovery. In particular, it does not concern itself with such issues as source address selection or the selecting of an outgoing interface on a multihomed host.

### 5.1. Conceptual Data Structures

Hosts will need to maintain the following pieces of information for each interface:

Neighbor Cache

- A set of entries about individual neighbors to which traffic has been sent recently. Entries are keyed on the neighbor's on-link unicast IP address and contain such information as its link-layer address, a flag indicating whether the neighbor is a router or a host (called IsRouter in this document), a pointer to any queued packets waiting for address resolution to complete, etc.

  A Neighbor Cache entry also contains information used by the Neighbor Unreachability Detection algorithm, including the reachability state, the number of unanswered probes, and the time the next Neighbor Unreachability Detection event is scheduled to take place.

Destination Cache

- A set of entries about destinations to which traffic has been sent recently. The Destination Cache includes both on-link and off-link destinations and provides a level of indirection into the Neighbor Cache; the Destination Cache maps a destination IP address to the IP address of the next-hop neighbor. This cache is updated with

information learned from Redirect messages. Implementations may find it convenient to store additional information not directly related to Neighbor Discovery in Destination Cache entries, such as the Path MTU (PMTU) and round trip timers maintained by transport protocols.

Prefix List - A list of the prefixes that define a set of addresses that are on-link. Prefix List entries are created from information received in Router Advertisements. Each entry has an associated invalidation timer value (extracted from the advertisement) used to expire prefixes when they become invalid. A special "infinity" timer value specifies that a prefix remains valid forever, unless a new (finite) value is received in a subsequent advertisement.

The link-local prefix is considered to be on the prefix list with an infinite invalidation timer regardless of whether routers are advertising a prefix for it. Received Router Advertisements SHOULD NOT modify the invalidation timer for the link-local prefix.

Default Router List
- A list of routers to which packets may be sent. Router list entries point to entries in the Neighbor Cache; the algorithm for selecting a default router favors routers known to be reachable over those whose reachability is suspect. Each entry also has an associated invalidation timer value (extracted from Router Advertisements) used to delete entries that are no longer advertised.

Note that the above conceptual data structures can be implemented using a variety of techniques. One possible implementation is to use a single longest-match routing table for all of the above data structures. Regardless of the specific implementation, it is critical that the Neighbor Cache entry for a router is shared by all Destination Cache entries using that router in order to prevent redundant Neighbor Unreachability Detection probes.

Note also that other protocols (e.g., IPv6 Mobility) might add additional conceptual data structures. An implementation is at liberty to implement such data structures in any way it pleases. For example, an implementation could merge all conceptual data structures into a single routing table.

The Neighbor Cache contains information maintained by the Neighbor Unreachability Detection algorithm. A key piece of information is a neighbor's reachability state, which is one of five possible values. The following definitions are informal; precise definitions can be found in Section 7.3.2.

INCOMPLETE Address resolution is in progress and the link-layer address of the neighbor has not yet been determined.

REACHABLE Roughly speaking, the neighbor is known to have been reachable recently (within tens of seconds ago).

STALE The neighbor is no longer known to be reachable but until traffic is sent to the neighbor, no attempt should be made to verify its reachability.

DELAY The neighbor is no longer known to be reachable, and traffic has recently been sent to the neighbor. Rather than probe the neighbor immediately, however, delay sending probes for a short while in order to give upper layer protocols a chance to provide reachability confirmation.

PROBE The neighbor is no longer known to be reachable, and unicast Neighbor Solicitation probes are being sent to verify reachability.

## 5.2. Conceptual Sending Algorithm

When sending a packet to a destination, a node uses a combination of the Destination Cache, the Prefix List, and the Default Router List to determine the IP address of the appropriate next hop, an operation known as "next-hop determination". Once the IP address of the next hop is known, the Neighbor Cache is consulted for link-layer information about that neighbor.

Next-hop determination for a given unicast destination operates as follows. The sender performs a longest prefix match against the Prefix List to determine whether the packet's destination is on- or off-link. If the destination is on-link, the next-hop address is the same as the packet's destination address. Otherwise, the sender selects a router from the Default Router List (following the rules described in Section 6.3.6). If the Default Router List is empty, the sender assumes that the destination is on-link.

For efficiency reasons, next-hop determination is not performed on every packet that is sent. Instead, the results of next-hop determination computations are saved in the Destination Cache (which

also contains updates learned from Redirect messages). When the sending node has a packet to send, it first examines the Destination Cache. If no entry exists for the destination, next-hop determination is invoked to create a Destination Cache entry.

Once the IP address of the next-hop node is known, the sender examines the Neighbor Cache for link-layer information about that neighbor. If no entry exists, the sender creates one, sets its state to INCOMPLETE, initiates Address Resolution, and then queues the data packet pending completion of address resolution. For multicast-capable interfaces Address Resolution consists of sending a Neighbor Solicitation message and waiting for a Neighbor Advertisement. When a Neighbor Advertisement response is received, the link-layer addresses is entered in the Neighbor Cache entry and the queued packet is transmitted. The address resolution mechanism is described in detail in Section 7.2.

For multicast packets the next-hop is always the (multicast) destination address and is considered to be on-link. The procedure for determining the link-layer address corresponding to a given IP multicast address can be found in a separate document that covers operating IP over a particular link type (e.g., [IPv6-ETHER]).

Each time a Neighbor Cache entry is accessed while transmitting a unicast packet, the sender checks Neighbor Unreachability Detection related information according to the Neighbor Unreachability Detection algorithm (Section 7.3). This unreachability check might result in the sender transmitting a unicast Neighbor Solicitation to verify that the neighbor is still reachable.

Next-hop determination is done the first time traffic is sent to a destination. As long as subsequent communication to that destination proceeds successfully, the Destination Cache entry continues to be used. If at some point communication ceases to proceed, as determined by the Neighbor Unreachability Detection algorithm, next-hop determination may need to be performed again. For example, traffic through a failed router should be switched to a working router. Likewise, it may be possible to reroute traffic destined for a mobile node to a "mobility agent".

Note that when a node redoes next-hop determination there is no need to discard the complete Destination Cache entry. In fact, it is generally beneficial to retain such cached information as the PMTU and round trip timer values that may also be kept in the Destination Cache entry.

Routers and multihomed hosts have multiple interfaces. The remainder of this document assumes that all sent and received Neighbor Discovery messages refer to the interface of appropriate context. For example, when responding to a Router Solicitation, the corresponding Router Advertisement is sent out the interface on which the solicitation was received.

### 5.3. Garbage Collection and Timeout Requirements

The conceptual data structures described above use different mechanisms for discarding potentially stale or unused information.

From the perspective of correctness there is no need to periodically purge Destination and Neighbor Cache entries. Although stale information can potentially remain in the cache indefinitely, the Neighbor Unreachability Detection algorithm ensures that stale information is purged quickly if it is actually being used.

To limit the storage needed for the Destination and Neighbor Caches, a node may need to garbage-collect old entries. However, care must be taken to insure that sufficient space is always present to hold the working set of active entries. A small cache may result in an excessive number of Neighbor Discovery messages if entries are discarded and rebuilt in quick succession. Any LRU-based policy that only reclaims entries that have not been used in some time (e.g., ten minutes or more) should be adequate for garbage-collecting unused entries.

A node should retain entries in the Default Router List and the Prefix List until their lifetimes expire. However, a node may garbage collect entries prematurely if it is low on memory. If not all routers are kept on the Default Router list, a node should retain at least two entries in the Default Router List (and preferably more) in order to maintain robust connectivity for off-link destinations.

When removing an entry from the Prefix List there is no need to purge any entries from the Destination or Neighbor Caches. Neighbor Unreachability Detection will efficiently purge any entries in these caches that have become invalid. When removing an entry from the Default Router List, however, any entries in the Destination Cache that go through that router must perform next-hop determination again to select a new default router.

## 6. ROUTER AND PREFIX DISCOVERY

This section describes router and host behavior related to the Router Discovery portion of Neighbor Discovery. Router Discovery is used to locate neighboring routers as well as learn prefixes and

configuration parameters related to address autoconfiguration.

Prefix Discovery is the process through which hosts learn the ranges of IP addresses that reside on-link and can be reached directly without going through a router. Routers send Router Advertisements that indicate whether the sender is willing to be a default router. Router Advertisements also contain Prefix Information options that list the set of prefixes that identify on-link IP addresses.

Stateless Address Autoconfiguration must also obtain subnet prefixes as part of configuring addresses. Although the prefixes used for address autoconfiguration are logically distinct from those used for on-link determination, autoconfiguration information is piggybacked on Router Discovery messages to reduce network traffic. Indeed, the same prefixes can be advertised for on-link determination and address autoconfiguration by specifying the appropriate flags in the Prefix Information options. See [ADDRCONF] for details on how autoconfiguration information is processed.

## 6.1. Message Validation

### 6.1.1. Validation of Router Solicitation Messages

Hosts MUST silently discard any received Router Solicitation Messages.

A router MUST silently discard any received Router Solicitation messages that do not satisfy all of the following validity checks:

- The IP Hop Limit field has a value of 255, i.e., the packet could not possibly have been forwarded by a router.
- If the message includes an IP Authentication Header, the message authenticates correctly.
- ICMP Checksum is valid.
- ICMP Code is 0.
- ICMP length (derived from the IP length) is 8 or more octets.
- All included options have a length that is greater than zero.
- If the IP source address is the unspecified address, there is no source link-layer address option in the message.

The contents of the Reserved field, and of any unrecognized options, MUST be ignored. Future, backward-compatible changes to the protocol may specify the contents of the Reserved field or add new options; backward-incompatible changes may use different Code values.

The contents of any defined options that are not specified to be used with Router Solicitation messages MUST be ignored and the packet processed as normal. The only defined option that may appear is the Source Link-Layer Address option.

A solicitation that passes the validity checks is called a "valid solicitation".

### 6.1.2. Validation of Router Advertisement Messages

A node MUST silently discard any received Router Advertisement messages that do not satisfy all of the following validity checks:

- IP Source Address is a link-local address. Routers must use their link-local address as the source for Router Advertisement and Redirect messages so that hosts can uniquely identify routers.

- The IP Hop Limit field has a value of 255, i.e., the packet could not possibly have been forwarded by a router.

- If the message includes an IP Authentication Header, the message authenticates correctly.

- ICMP Checksum is valid.

- ICMP Code is 0.

- ICMP length (derived from the IP length) is 16 or more octets.

- All included options have a length that is greater than zero.

The contents of the Reserved field, and of any unrecognized options, MUST be ignored. Future, backward-compatible changes to the protocol may specify the contents of the Reserved field or add new options; backward-incompatible changes may use different Code values.

The contents of any defined options that are not specified to be used with Router Advertisement messages MUST be ignored and the packet processed as normal. The only defined options that may appear are the Source Link-Layer Address, Prefix Information and MTU options.

An advertisement that passes the validity checks is called a "valid advertisement".

## 6.2. Router Specification

### 6.2.1. Router Configuration Variables

A router MUST allow for the following conceptual variables to be configured by system management. The specific variable names are used for demonstration purposes only, and an implementation is not required to have them, so long as its external behavior is consistent with that described in this document. Default values are specified to simplify configuration in common cases.

The default values for some of the variables listed below may be overridden by specific documents that describe how IPv6 operates over different link layers. This rule simplifies the configuration of Neighbor Discovery over link types with widely differing performance characteristics.

For each multicast interface:

AdvSendAdvertisements
: A flag indicating whether or not the router sends periodic Router Advertisements and responds to Router Solicitations.

  Default: FALSE

  Note that AdvSendAdvertisements MUST be FALSE by default so that a node will not accidentally start acting as a router unless it is explicitly configured by system management to send Router Advertisements.

MaxRtrAdvInterval
: The maximum time allowed between sending unsolicited multicast Router Advertisements from the interface, in seconds. MUST be no less than 4 seconds and no greater than 1800 seconds.

  Default: 600 seconds

MinRtrAdvInterval
: The minimum time allowed between sending unsolicited multicast Router Advertisements from the interface, in seconds. MUST be no less than 3 seconds and no greater than .75 *

MaxRtrAdvInterval.

Default: 0.33 * MaxRtrAdvInterval

AdvManagedFlag
The TRUE/FALSE value to be placed in the "Managed address configuration" flag field in the Router Advertisement. See [ADDRCONF].

Default: FALSE

AdvOtherConfigFlag
The TRUE/FALSE value to be placed in the "Other stateful configuration" flag field in the Router Advertisement. See [ADDRCONF].

Default: FALSE

AdvLinkMTU The value to be placed in MTU options sent by the router. A value of zero indicates that no MTU options are sent.

Default: 0

AdvReachableTime
The value to be placed in the Reachable Time field in the Router Advertisement messages sent by the router. The value zero means unspecified (by this router). MUST be no greater than 3,600,000 milliseconds (1 hour).

Default: 0

AdvRetransTimer The value to be placed in the Retrans Timer field in the Router Advertisement messages sent by the router. The value zero means unspecified (by this router).

Default: 0

AdvCurHopLimit
The default value to be placed in the Cur Hop Limit field in the Router Advertisement messages sent by the router. The value should be set to that current diameter of the Internet. The value zero means unspecified (by this router).

Default: The value specified in the "Assigned Numbers" RFC [ASSIGNED] that was in effect at the time of implementation.

AdvDefaultLifetime
The value to be placed in the Router Lifetime field of Router Advertisements sent from the interface, in seconds. MUST be either zero or between MaxRtrAdvInterval and 9000 seconds. A value of zero indicates that the router is not to be used as a default router.

Default: 3 * MaxRtrAdvInterval

AdvPrefixList
A list of prefixes to be placed in Prefix Information options in Router Advertisement messages sent from the interface.

Default: all prefixes that the router advertises via routing protocols as being on-link for the interface from which the advertisement is sent. The link-local prefix SHOULD NOT be included in the list of advertised prefixes.

Each prefix has an associated:

AdvValidLifetime
The value to be placed in the Valid Lifetime in the Prefix Information option, in seconds. The designated value of all 1's (0xffffffff) represents infinity. Implementations MUST allow AdvValidLifetime to be specified in two ways:

- a time that decrements in real time, that is, one that will result in a Lifetime of zero at the specified time in the future, or

- a fixed time that stays the same in consecutive advertisements.

Default: 2592000 seconds (30 days), fixed (i.e., stays the same in consecutive advertisements).

AdvOnLinkFlag
The value to be placed in the on-link flag ("L-bit") field in the Prefix Information option.

Default: TRUE

Automatic address configuration [ADDRCONF] defines additional information associated with each the prefixes:

AdvPreferredLifetime
The value to be placed in the Preferred Lifetime in the Prefix Information option, in seconds. The designated value of all 1's (0xffffffff) represents infinity. See [ADDRCONF] for details on how this value is used. Implementations MUST allow AdvPreferredLifetime to be specified in two ways:

- a time that decrements in real time, that is, one that will result in a Lifetime of zero at a specified time in the future, or

- a fixed time that stays the same in consecutive advertisements.

Default: 604800 seconds (7 days), fixed (i.e., stays the same in consecutive advertisements).

AdvAutonomousFlag
The value to be placed in the Autonomous Flag field in the Prefix Information option. See [ADDRCONF].

Default: TRUE

The above variables contain information that is placed in outgoing Router Advertisement messages. Hosts use the received information to initialize a set of analogous variables that control their external behavior (see Section 6.3.2). Some of these host variables (e.g., CurHopLimit, RetransTimer, and ReachableTime) apply to all nodes including routers. In practice, these variables may not actually be present on routers, since their contents can be derived from the variables described above. However, external router behavior MUST be

the same as host behavior with respect to these variables. In particular, this includes the occasional randomization of the ReachableTime value as described in Section 6.3.2.

Protocol constants are defined in Section 10.

### 6.2.2. Becoming An Advertising Interface

The term "advertising interface" refers to any functioning and enabled multicast interface that has at least one unicast IP address assigned to it and whose corresponding AdvSendAdvertisements flag is TRUE. A router MUST NOT send Router Advertisements out any interface that is not an advertising interface.

An interface may become an advertising interface at times other than system startup. For example:

- changing the AdvSendAdvertisements flag on an enabled interface from FALSE to TRUE, or

- administratively enabling the interface, if it had been administratively disabled, and its AdvSendAdvertisements flag is TRUE, or

- enabling IP forwarding capability (i.e., changing the system from being a host to being a router), when the interface's AdvSendAdvertisements flag is TRUE.

A router MUST join the all-routers multicast address on an advertising interface. Routers respond to Router Solicitations sent to the all-routers address and verify the consistency of Router Advertisements sent by neighboring routers.

### 6.2.3. Router Advertisement Message Content

A router sends periodic as well as solicited Router Advertisements out its advertising interfaces. Outgoing Router Advertisements are filled with the following values consistent with the message format given in Section 4.2:

- In the Router Lifetime field: the interface's configured AdvDefaultLifetime.

- In the M and O flags: the interface's configured AdvManagedFlag and AdvOtherConfigFlag, respectively. See [ADDRCONF].

- In the Cur Hop Limit field: the interface's configured CurHopLimit.

- In the Reachable Time field: the interface's configured AdvReachableTime.

- In the Retrans Timer field: the interface's configured AdvRetransTimer.

- In the options:

  o Source Link-Layer Address option: link-layer address of the sending interface. This option MAY be omitted to facilitate in-bound load balancing over replicated interfaces.

  o MTU option: the interface's configured AdvLinkMTU value if the value is non-zero. If AdvLinkMTU is zero the MTU option is not sent.

  o Prefix Information options: one Prefix Information option for each prefix listed in AdvPrefixList with the option fields set from the information in the AdvPrefixList entry as follows:

    - In the "on-link" flag: the entry's AdvOnLinkFlag.

    - In the Valid Lifetime field: the entry's AdvValidLifetime.

    - In the "Autonomous address configuration" flag: the entry's AdvAutonomousFlag.

    - In the Preferred Lifetime field: the entry's AdvPreferredLifetime.

A router might want to send Router Advertisements without advertising itself as a default router. For instance, a router might advertise prefixes for address autoconfiguration while not wishing to forward packets. Such a router sets the Router Lifetime field in outgoing advertisements to zero.

A router MAY choose not to include some or all options when sending unsolicited Router Advertisements. For example, if prefix lifetimes are much longer than AdvDefaultLifetime, including them every few advertisements may be sufficient. However, when responding to a Router Solicitation or while sending the first few initial unsolicited advertisements, a router SHOULD include all options so that all information (e.g., prefixes) is propagated quickly during system initialization.

If including all options causes the size of an advertisement to exceed the link MTU, multiple advertisements can be sent, each containing a subset of the options.

### 6.2.4. Sending Unsolicited Router Advertisements

A host MUST NOT send Router Advertisement messages at any time.

Unsolicited Router Advertisements are not strictly periodic: the interval between subsequent transmissions is randomized to reduce the probability of synchronization with the advertisements from other routers on the same link [SYNC]. Each advertising interface has its own timer. Whenever a multicast advertisement is sent from an interface, the timer is reset to a uniformly-distributed random value between the interface's configured MinRtrAdvInterval and MaxRtrAdvInterval; expiration of the timer causes the next advertisement to be sent and a new random value to be chosen.

For the first few advertisements (up to MAX_INITIAL_RTR_ADVERTISEMENTS) sent from an interface when it becomes an advertising interface, if the randomly chosen interval is greater than MAX_INITIAL_RTR_ADVERT_INTERVAL, the timer SHOULD be set to MAX_INITIAL_RTR_ADVERT_INTERVAL instead. Using a smaller interval for the initial advertisements increases the likelihood of a router being discovered quickly when it first becomes available, in the presence of possible packet loss.

The information contained in Router Advertisements may change through actions of system management. For instance, the lifetime of advertised prefixes may change, new prefixes could be added, a router could cease to be a router (i.e., switch from being a router to being a host), etc. In such cases, the router MAY transmit up to MAX_INITIAL_RTR_ADVERTISEMENTS unsolicited advertisements, using the same rules as when an interface becomes an advertising interface.

### 6.2.5. Ceasing To Be An Advertising Interface

An interface may cease to be an advertising interface, through actions of system management such as:

- changing the AdvSendAdvertisements flag of an enabled interface from TRUE to FALSE, or
- administratively disabling the interface, or
- shutting down the system.

In such cases the router SHOULD transmit one or more (but not more than MAX_FINAL_RTR_ADVERTISEMENTS) final multicast Router Advertisements on the interface with a Router Lifetime field of zero. In the case of a router becoming a host, the system SHOULD also depart from the all-routers IP multicast group on all interfaces on which the router supports IP multicast (whether or not they had been advertising interfaces). In addition, the host MUST insure that subsequent Neighbor Advertisement messages sent from the interface have the Router flag set to zero.

Note that system management may disable a router's IP forwarding capability (i.e., changing the system from being a router to being a host), a step that does not necessarily imply that the router's interfaces stop being advertising interfaces. In such cases, subsequent Router Advertisements MUST set the Router Lifetime field to zero.

### 6.2.6. Processing Router Solicitations

A host MUST silently discard any received Router Solicitation messages.

In addition to sending periodic, unsolicited advertisements, a router sends advertisements in response to valid solicitations received on an advertising interface. A router MAY choose to unicast the response directly to the soliciting host's address (if the solicitation's source address is not the unspecified address), but the usual case is to multicast the response to the all-nodes group. In the latter case, the interface's interval timer is reset to a new random value, as if an unsolicited advertisement had just been sent (see Section 6.2.4).

In all cases, Router Advertisements sent in response to a Router Solicitation MUST be delayed by a random time between 0 and MAX_RA_DELAY_TIME seconds. (If a single advertisement is sent in response to multiple solicitations, the delay is relative to the first solicitation.) In addition, consecutive Router Advertisements sent to the all-nodes multicast address MUST be rate limited to no more than one advertisement every MIN_DELAY_BETWEEN_RAS seconds.

A router might process Router Solicitations as follows:

- Upon receipt of a Router Solicitation, compute a random delay within the range 0 through MAX_RA_DELAY_TIME. If the computed value corresponds to a time later than the time the next multicast Router Advertisement is scheduled to be sent, ignore the random delay and send the advertisement at the already-scheduled time.

- If the router sent a multicast Router Advertisement (solicited or unsolicited) within the last MIN_DELAY_BETWEEN_RAS seconds, schedule the advertisement to be sent at a time corresponding to MIN_DELAY_BETWEEN_RAS plus the random value after the previous advertisement was sent. This ensures that the multicast Router Advertisements are rate limited.

- Otherwise, schedule the sending of a Router Advertisement at the time given by the random value.

Note that a router is permitted to send multicast Router Advertisements more frequently than indicated by the MinRtrAdvInterval configuration variable so long as the more frequent advertisements are responses to Router Solicitations. In all cases, however, unsolicited multicast advertisements MUST NOT be sent more frequently than indicated by MinRtrAdvInterval.

Router Solicitations in which the Source Address is the unspecified address MUST NOT update the router's Neighbor Cache; solicitations with a proper source address update the Neighbor Cache as follows. If the router already has a Neighbor Cache entry for the solicitation's sender, the solicitation contains a Source Link-Layer Address option, and the received link-layer address differs from that already in the cache, the link-layer address SHOULD be updated in the appropriate Neighbor Cache entry, and its reachability state MUST also be set to STALE. If there is no existing Neighbor Cache entry for the solicitation's sender, the router creates one, installs the link-layer address and sets its reachability state to STALE as specified in Section 7.3.3. Whether or not a Source Link-Layer Address option is provided, if a Neighbor Cache entry for the solicitation's sender exists (or is created) the entry's IsRouter flag MUST be set to FALSE.

### 6.2.7. Router Advertisement Consistency

Routers SHOULD inspect valid Router Advertisements sent by other routers and verify that the routers are advertising consistent information on a link. Detected inconsistencies indicate that one or more routers might be misconfigured and SHOULD be logged to system or network management. The minimum set of information to check includes:

- Cur Hop Limit values (except for the unspecified value of zero).

- Values of the M or O flags.

- Reachable Time values (except for the unspecified value of zero).

- Retrans Timer values (except for the unspecified value of zero).

- Values in the MTU options.

- Preferred and Valid Lifetimes for the same prefix. If AdvPreferredLifetime and/or AdvValidLifetime decrement in real time as specified in section 6.2.7 then the comparison of the lifetimes can not compare the content of the fields in the Router Advertisement but must instead compare the time at which the prefix will become deprecated and invalidated, respectively. Due to link propagation delays and potentially poorly synchronized clocks between the routers such comparison SHOULD allow some time skew.

Note that it is not an error for different routers to advertise different sets of prefixes. Also, some routers might leave some fields as unspecified, i.e., with the value zero, while other routers specify values. The logging of errors SHOULD be restricted to conflicting information that causes hosts to switch from one value to another with each received advertisement.

Any other action on reception of Router Advertisement messages by a router is beyond the scope of this document.

### 6.2.8. Link-local Address Change

The link-local address on a router SHOULD change rarely, if ever. Nodes receiving Neighbor Discovery messages use the source address to identify the sender. If multiple packets from the same router contain different source addresses, nodes will assume they come from different routers, leading to undesirable behavior. For example, a node will ignore Redirect messages that are believed to have been sent by a router other than the current first-hop router. Thus the source address used in Router Advertisements sent by a particular router must be identical to the target address in a Redirect message when redirecting to that router.

Using the link-local address to uniquely identify routers on the link has the benefit that the address a router is known by should not change when a site renumbers.

If a router changes the link-local address for one of its interfaces, it SHOULD inform hosts of this change. The router SHOULD multicast a few Router Advertisements from the old link-local address with the Router Lifetime field set to zero and also multicast a few Router Advertisements from the new link-local address. The overall effect should be the same as if one interface ceases being an advertising interface, and a different one starts being an advertising interface.

## 6.3. Host Specification

### 6.3.1. Host Configuration Variables

None.

### 6.3.2. Host Variables

A host maintains certain Neighbor Discovery related variables in addition to the data structures defined in Section 5.1. The specific variable names are used for demonstration purposes only, and an implementation is not required to have them, so long as its external behavior is consistent with that described in this document.

These variables have default values that are overridden by information received in Router Advertisement messages. The default values are used when there is no router on the link or when all received Router Advertisements have left a particular value unspecified.

The default values in this specification may be overridden by specific documents that describe how IP operates over different link layers. This rule allows Neighbor Discovery to operate over links with widely varying performance characteristics.

For each interface:

LinkMTU — The MTU of the link.
Default: The valued defined in the specific document that describes how IPv6 operates over the particular link layer (e.g., [IPv6-ETHER]).

CurHopLimit — The default hop limit to be used when sending (unicast) IP packets.

Default: The value specified in the "Assigned Numbers" RFC [ASSIGNED] that was in effect at the time of implementation.

BaseReachableTime
A base value used for computing the random ReachableTime value.

Default: REACHABLE_TIME milliseconds.

ReachableTime — The time a neighbor is considered reachable after receiving a reachability confirmation.

This value should be a uniformly-distributed random value between MIN_RANDOM_FACTOR and MAX_RANDOM_FACTOR times BaseReachableTime milliseconds. A new random value should be calculated when BaseReachableTime changes (due to Router Advertisements) or at least every few hours even if no Router Advertisements are received.

RetransTimer The time between retransmissions of Neighbor Solicitation messages to a neighbor when resolving the address or when probing the reachability of a neighbor.

Default: RETRANS_TIMER milliseconds

### 6.3.3. Interface Initialization

The host joins the all-nodes multicast address on all multicast-capable interfaces.

### 6.3.4. Processing Received Router Advertisements

When multiple routers are present, the information advertised collectively by all routers may be a superset of the information contained in a single Router Advertisement. Moreover, information may also be obtained through other dynamic means, such as stateful autoconfiguration. Hosts accept the union of all received information; the receipt of a Router Advertisement MUST NOT invalidate all information received in a previous advertisement or from another source. However, when received information for a specific parameter (e.g., Link MTU) or option (e.g., Lifetime on a specific Prefix) differs from information received earlier, and the parameter/option can only have one value, the most recently-received information is considered authoritative.

Some Router Advertisement fields (e.g., Cur Hop Limit, Reachable Time and Retrans Timer) may contain a value denoting unspecified. In such cases, the parameter should be ignored and the host should continue using whatever value it is already using. In particular, a host MUST NOT interpret the unspecified value as meaning change back to the default value that was in use before the first Router Advertisement was received. This rule prevents hosts from continually changing an internal variable when one router advertises a specific value, but other routers advertise the unspecified value.

On receipt of a valid Router Advertisement, a host extracts the source address of the packet and does the following:

- If the address is not already present in the host's Default Router List, and the advertisement's Router Lifetime is non-zero, create a new entry in the list, and initialize its invalidation timer value from the advertisement's Router Lifetime field.

- If the address is already present in the host's Default Router List as a result of a previously-received advertisement, reset its invalidation timer to the Router Lifetime value in the newly-received advertisement.

- If the address is already present in the host's Default Router List and the received Router Lifetime value is zero, immediately time-out the entry as specified in Section 6.3.5.

To limit the storage needed for the Default Router List, a host MAY choose not to store all of the router addresses discovered via advertisements. However, a host MUST retain at least two router addresses and SHOULD retain more. Default router selections are made whenever communication to a destination appears to be failing. Thus, the more routers on the list, the more likely an alternative working router can be found quickly (e.g., without having to wait for the next advertisement to arrive).

If the received Cur Hop Limit value is non-zero the host SHOULD set its CurHopLimit variable to the received value.

If the received Reachable Time value is non-zero the host SHOULD set its BaseReachableTime variable to the received value. If the new value differs from the previous value, the host SHOULD recompute a new random ReachableTime value. ReachableTime is computed as a uniformly-distributed random value between MIN_RANDOM_FACTOR and MAX_RANDOM_FACTOR times the BaseReachableTime. Using a random component eliminates the possibility Neighbor Unreachability Detection messages synchronize with each other.

In most cases, the advertised Reachable Time value will be the same in consecutive Router Advertisements and a host's BaseReachableTime rarely changes. In such cases, an implementation SHOULD insure that a new random value gets recomputed at least once every few hours.

The RetransTimer variable SHOULD be copied from the Retrans Timer field, if the received value is non-zero.

After extracting information from the fixed part of the Router Advertisement message, the advertisement is scanned for valid options. If the advertisement contains a Source Link-Layer Address option the link-layer address SHOULD be recorded in the Neighbor

Cache entry for the router (creating an entry if necessary) and the IsRouter flag in the Neighbor Cache entry MUST be set to TRUE. If no Source Link-Layer Address is included, but a corresponding Neighbor Cache entry exists, its IsRouter flag MUST be set to TRUE. The IsRouter flag is used by Neighbor Unreachability Detection to determine when a router changes to being a host (i.e., no longer capable of forwarding packets). If a Neighbor Cache entry is created for the router its reachability state MUST be set to STALE as specified in Section 7.3.3. If a cache entry already exists and is updated with a different link-layer address the reachability state MUST also be set to STALE.

If the MTU option is present, hosts SHOULD copy the option's value into LinkMTU so long as the value is greater than or equal to the minimum link MTU [IPv6] and does not exceed the default LinkMTU value specified in the link type specific document (e.g., [IPv6-ETHER]).

Prefix Information options that have the "on-link" (L) flag set indicate a prefix identifying a range of addresses that should be considered on-link. Note, however, that a Prefix Information option with the on-link flag set to zero conveys no information concerning on-link determination and MUST NOT be interpreted to mean that addresses covered by the prefix are off-link. The only way to cancel a previous on-link indication is to advertise that prefix with the L-bit set and the Lifetime set to zero. The default behavior (see Section 5.2) when sending a packet to an address for which no information is known about the on-link status of the address is to forward the packet to a default router; the reception of a Prefix Information option with the "on-link " (L) flag set to zero does not change this behavior. The reasons for an address being treated as on-link is specified in the definition of "on-link" in Section 2.1. Prefixes with the on-link flag set to zero would normally have the autonomous flag set and be used by [ADDRCONF].

For each Prefix Information option with the on-link flag set, a host does the following:

- If the prefix is the link-local prefix, silently ignore the Prefix Information option.

- If the prefix is not already present in the Prefix List, and the Prefix Information option's Valid Lifetime field is non-zero, create a new entry for the prefix and initialize its invalidation timer to the Valid Lifetime value in the Prefix Information option.

- If the prefix is already present in the host's Prefix List as the result of a previously-received advertisement, reset its

invalidation timer to the Valid Lifetime value in the Prefix Information option. If the new Lifetime value is zero, time-out the prefix immediately (see Section 6.3.5).

- If the Prefix Information option's Valid Lifetime field is zero, and the prefix is not present in the host's Prefix List, silently ignore the option.

Stateless address autoconfiguration [ADDRCONF] may in some circumstances increase the Valid Lifetime of a prefix or ignore it completely in order to prevent a particular denial of service attack. However, since the effect of the same denial of service targeted at the on-link prefix list is not catastrophic (hosts would send packets to a default router and receive a redirect rather than sending packets directly to a neighbor) the Neighbor Discovery protocol does not impose such a check on the prefix lifetime values.

Note: Implementations can choose to process the on-link aspects of the prefixes separately from the address autoconfiguration aspects of the prefixes by, e.g., passing a copy of each valid Router Advertisement message to both an "on-link" and an "addrconf" function. Each function can then operate independently on the prefixes that have the appropriate flag set.

### 6.3.5. Timing out Prefixes and Default Routers

Whenever the invalidation timer expires for a Prefix List entry, that entry is discarded. No existing Destination Cache entries need be updated, however. Should a reachability problem arise with an existing Neighbor Cache entry, Neighbor Unreachability Detection will perform any needed recovery.

Whenever the Lifetime of an entry in the Default Router List expires, that entry is discarded. When removing a router from the Default Router list, the node MUST update the Destination Cache in such a way that all entries using the router perform next-hop determination again rather than continue sending traffic to the (deleted) router.

### 6.3.6. Default Router Selection

The algorithm for selecting a router depends in part on whether or not a router is known to be reachable. The exact details of how a node keeps track of a neighbor's reachability state are covered in Section 7.3. The algorithm for selecting a default router is invoked during next-hop determination when no Destination Cache entry exists for an off-link destination or when communication through an existing router appears to be failing. Under normal conditions, a router would be selected the first time traffic is sent to a destination,

with subsequent traffic for that destination using the same router as indicated in the Destination Cache modulo any changes to the Destination Cache caused by Redirect messages.

The policy for selecting routers from the Default Router List is as follows:

1) Routers that are reachable or probably reachable (i.e., in any state other than INCOMPLETE) SHOULD be preferred over routers whose reachability is unknown or suspect (i.e., in the INCOMPLETE state, or for which no Neighbor Cache entry exists). An implementation may choose to always return the same router or cycle through the router list in a round-robin fashion as long as it always returns a reachable or a probably reachable router when one is available.

2) When no routers on the list are known to be reachable or probably reachable, routers SHOULD be selected in a round-robin fashion, so that subsequent requests for a default router do not return the same router until all other routers have been selected.

   Cycling through the router list in this case ensures that all available routers are actively probed by the Neighbor Unreachability Detection algorithm. A request for a default router is made in conjunction with the sending of a packet to a router, and the selected router will be probed for reachability as a side effect.

3) If the Default Router List is empty, assume that all destinations are on-link as specified in Section 5.2.

### 6.3.7. Sending Router Solicitations

When an interface becomes enabled, a host may be unwilling to wait for the next unsolicited Router Advertisement to locate default routers or learn prefixes. To obtain Router Advertisements quickly, a host SHOULD transmit up to MAX_RTR_SOLICITATIONS Router Solicitation messages each separated by at least RTR_SOLICITATION_INTERVAL seconds. Router Solicitations may be sent after any of the following events:

- The interface is initialized at system startup time.

- The interface is reinitialized after a temporary interface failure or after being temporarily disabled by system management.

- The system changes from being a router to being a host, by having its IP forwarding capability turned off by system management.

- The host attaches to a link for the first time.

- The host re-attaches to a link after being detached for some time.

A host sends Router Solicitations to the All-Routers multicast address. The IP source address is set to either one of the interface's unicast addresses or the unspecified address. The Source Link-Layer Address option SHOULD be set to the host's link-layer address, if the IP source address is not the unspecified address.

Before a host sends an initial solicitation, it SHOULD delay the transmission for a random amount of time between 0 and MAX_RTR_SOLICITATION_DELAY. This serves to alleviate congestion when many hosts start up on a link at the same time, such as might happen after recovery from a power failure. If a host has already performed a random delay since the interface became (re)enabled (e.g., as part of Duplicate Address Detection [ADDRCONF]) there is no need to delay again before sending the first Router Solicitation message.

Once the host sends a Router Solicitation, and receives a valid Router Advertisement with a non-zero Router Lifetime, the host MUST desist from sending additional solicitations on that interface, until the next time one of the above events occurs. Moreover, a host SHOULD send at least one solicitation in the case where an advertisement is received prior to having sent a solicitation. Unsolicited Router Advertisements may be incomplete (see Section 6.2.3); solicited advertisements are expected to contain complete information.

If a host sends MAX_RTR_SOLICITATIONS solicitations, and receives no Router Advertisements after having waited MAX_RTR_SOLICITATION_DELAY seconds after sending the last solicitation, the host concludes that there are no routers on the link for the purpose of [ADDRCONF]. However, the host continues to receive and process Router Advertisements messages in the event that routers appear on the link.

## 7. ADDRESS RESOLUTION AND NEIGHBOR UNREACHABILITY DETECTION

This section describes the functions related to Neighbor Solicitation and Neighbor Advertisement messages and includes descriptions of address resolution and the Neighbor Unreachability Detection algorithm.

Neighbor Solicitation and Advertisement messages are also used for Duplicate Address Detection as specified by [ADDRCONF]. In particular, Duplicate Address Detection sends Neighbor Solicitation messages with an unspecified source address targeting its own "tentative" address. Such messages trigger nodes already using the address to respond with a multicast Neighbor Advertisement indicating that the address is in use.

## 7.1. Message Validation

### 7.1.1. Validation of Neighbor Solicitations

A node MUST silently discard any received Neighbor Solicitation messages that do not satisfy all of the following validity checks:

- The IP Hop Limit field has a value of 255, i.e., the packet could not possibly have been forwarded by a router.
- If the message includes an IP Authentication Header, the message authenticates correctly.
- ICMP Checksum is valid.
- ICMP Code is 0.
- ICMP length (derived from the IP length) is 24 or more octets.
- Target Address is not a multicast address.
- All included options have a length that is greater than zero.
- If the IP source address is the unspecified address, the IP destination address is a solicited-node multicast address.
- If the IP source address is the unspecified address, there is no source link-layer address option in the message.

The contents of the Reserved field, and of any unrecognized options, MUST be ignored. Future, backward-compatible changes to the protocol may specify the contents of the Reserved field or add new options; backward-incompatible changes may use different Code values.

The contents of any defined options that are not specified to be used with Neighbor Solicitation messages MUST be ignored and the packet processed as normal. The only defined option that may appear is the Source Link-Layer Address option.

A Neighbor Solicitation that passes the validity checks is called a "valid solicitation".

### 7.1.2. Validation of Neighbor Advertisements

A node MUST silently discard any received Neighbor Advertisement messages that do not satisfy all of the following validity checks:

- The IP Hop Limit field has a value of 255, i.e., the packet could not possibly have been forwarded by a router.
- If the message includes an IP Authentication Header, the message authenticates correctly.
- ICMP Checksum is valid.
- ICMP Code is 0.
- ICMP length (derived from the IP length) is 24 or more octets.
- Target Address is not a multicast address.
- If the IP Destination Address is a multicast address the Solicited flag is zero.
- All included options have a length that is greater than zero.

The contents of the Reserved field, and of any unrecognized options, MUST be ignored. Future, backward-compatible changes to the protocol may specify the contents of the Reserved field or add new options; backward-incompatible changes may use different Code values.

The contents of any defined options that are not specified to be used with Neighbor Advertisement messages MUST be ignored and the packet processed as normal. The only defined option that may appear is the Target Link-Layer Address option.

A Neighbor Advertisements that passes the validity checks is called a "valid advertisement".

## 7.2. Address Resolution

Address resolution is the process through which a node determines the link-layer address of a neighbor given only its IP address. Address resolution is performed only on addresses that are determined to be on-link and for which the sender does not know the corresponding link-layer address. Address resolution is never performed on multicast addresses.

### 7.2.1. Interface Initialization

When a multicast-capable interface becomes enabled the node MUST join the all-nodes multicast address on that interface, as well as the solicited-node multicast address corresponding to each of the IP addresses assigned to the interface.

The set of addresses assigned to an interface may change over time. New addresses might be added and old addresses might be removed [ADDRCONF]. In such cases the node MUST join and leave the solicited-node multicast address corresponding to the new and old addresses, respectively. Note that multiple unicast addresses may map into the same solicited-node multicast address; a node MUST NOT leave the solicited-node multicast group until all assigned addresses corresponding to that multicast address have been removed.

### 7.2.2. Sending Neighbor Solicitations

When a node has a unicast packet to send to a neighbor, but does not know the neighbor's link-layer address, it performs address resolution. For multicast-capable interfaces this entails creating a Neighbor Cache entry in the INCOMPLETE state and transmitting a Neighbor Solicitation message targeted at the neighbor. The solicitation is sent to the solicited-node multicast address corresponding to the target address.

If the source address of the packet prompting the solicitation is the same as one of the addresses assigned to the outgoing interface, that address SHOULD be placed in the IP Source Address of the outgoing solicitation. Otherwise, any one of the addresses assigned to the interface should be used. Using the prompting packet's source address when possible insures that the recipient of the Neighbor Solicitation installs in its Neighbor Cache the IP address that is highly likely to be used in subsequent return traffic belonging to the prompting packet's "connection".

If the solicitation is being sent to a solicited-node multicast address, the sender MUST include its link-layer address (if it has one) as a Source Link-Layer Address option. Otherwise, the sender SHOULD include its link-layer address (if it has one) as a Source Link-Layer Address option. Including the source link-layer address in a multicast solicitation is required to give the target an address to which it can send the Neighbor Advertisement. On unicast solicitations, an implementation MAY omit the Source Link-Layer Address option. The assumption here is that if the sender has a peer's link-layer address in its cache, there is a high probability that the peer will also have an entry in its cache for the sender. Consequently, it need not be sent.

While waiting for address resolution to complete, the sender MUST, for each neighbor, retain a small queue of packets waiting for address resolution to complete. The queue MUST hold at least one packet, and MAY contain more. However, the number of queued packets per neighbor SHOULD be limited to some small value. When a queue overflows, the new arrival SHOULD replace the oldest entry. Once address resolution completes, the node transmits any queued packets.

While awaiting a response, the sender SHOULD retransmit Neighbor Solicitation messages approximately every RetransTimer milliseconds, even in the absence of additional traffic to the neighbor. Retransmissions MUST be rate-limited to at most one solicitation per neighbor every RetransTimer milliseconds.

If no Neighbor Advertisement is received after MAX_MULTICAST_SOLICIT solicitations, address resolution has failed. The sender MUST return ICMP destination unreachable indications with code 3 (Address Unreachable) for each packet queued awaiting address resolution.

### 7.2.3. Receipt of Neighbor Solicitations

A valid Neighbor Solicitation that does not meet any the following requirements MUST be silently discarded:

- The Target Address is a "valid" unicast or anycast address assigned to the receiving interface [ADDRCONF],

- The Target Address is a unicast address for which the node is offering proxy service, or

- The Target Address is a "tentative" address on which Duplicate Address Detection is being performed [ADDRCONF].

If the Target Address is tentative, the Neighbor Solicitation should be processed as described in [ADDRCONF]. Otherwise, the following description applies. If the Source Address is not the unspecified address and, on link layers that have addresses, the solicitation includes a Source Link-Layer Address option, then the recipient SHOULD create or update the Neighbor Cache entry for the IP Source Address of the solicitation. If an entry does not already exist, the node SHOULD create a new one and set its reachability state to STALE as specified in Section 7.3.3. If an entry already exists, and the cached link-layer address differs from the one in the received Source Link-Layer option, the cached address should be replaced by the received address and the entry’s reachability state MUST be set to STALE.

If a Neighbor Cache entry is created the IsRouter flag SHOULD be set to FALSE. This will be the case even if the Neighbor Solicitation is sent by a router since the Neighbor Solicitation messages do not contain an indication of whether or not the sender is a router. In the event that the sender is a router, subsequent Neighbor Advertisement or Router Advertisement messages will set the correct IsRouter value. If a Neighbor Cache entry already exists its IsRouter flag MUST NOT be modified.

If the Source Address is the unspecified address the node MUST NOT create or update the Neighbor Cache entry.

After any updates to the Neighbor Cache, the node sends a Neighbor Advertisement response as described in the next section.

### 7.2.4. Sending Solicited Neighbor Advertisements

A node sends a Neighbor Advertisement in response to a valid Neighbor Solicitation targeting one of the node's assigned addresses. The Target Address of the advertisement is copied from the Target Address of the solicitation. If the solicitation's IP Destination Address is not a multicast address, the Target Link-Layer Address option MAY be omitted; the neighboring node's cached value must already be current in order for the solicitation to have been received. If the solicitation's IP Destination Address is a multicast address, the Target Link-Layer option MUST be included in the advertisement. Furthermore, if the node is a router, it MUST set the Router flag to one; otherwise it MUST set the flag to zero.

If the Target Address is either an anycast address or a unicast address for which the node is providing proxy service, or the Target Link-Layer Address option is not included, the Override flag SHOULD be set to zero. Otherwise, the Override flag SHOULD be set to one. Proper setting of the Override flag ensures that nodes give preference to non-proxy advertisements, even when received after proxy advertisements, and also ensures that the first advertisement for an anycast address "wins".

If the source of the solicitation is the unspecified address, the node MUST set the Solicited flag to zero and multicast the advertisement to the all-nodes address. Otherwise, the node MUST set the Solicited flag to one and unicast the advertisement to the Source Address of the solicitation.

If the Target Address is an anycast address the sender SHOULD delay sending a response for a random time between 0 and MAX_ANYCAST_DELAY_TIME seconds.

Because unicast Neighbor Solicitations are not required to include a Source Link-Layer Address, it is possible that a node sending a solicited Neighbor Advertisement does not have a corresponding link-layer address for its neighbor in its Neighbor Cache. In such situations, a node will first have to use Neighbor Discovery to determine the link-layer address of its neighbor (i.e, send out a multicast Neighbor Solicitation).

### 7.2.5. Receipt of Neighbor Advertisements

When a valid Neighbor Advertisement is received (either solicited or unsolicited), the Neighbor Cache is searched for the target's entry. If no entry exists, the advertisement SHOULD be silently discarded. There is no need to create an entry if none exists, since the recipient has apparently not initiated any communication with the target.

Once the appropriate Neighbor Cache entry has been located, the specific actions taken depend on the state of the Neighbor Cache entry, the flags in the advertisement and the actual link-layer address supplied.

If the target's Neighbor Cache entry is in the INCOMPLETE state when the advertisement is received, one of two things happens. If the link layer has addresses and no Target Link-Layer address option is included, the receiving node SHOULD silently discard the received advertisement. Otherwise, the receiving node performs the following steps:

- It records the link-layer address in the Neighbor Cache entry.

- If the advertisement's Solicited flag is set, the state of the entry is set to REACHABLE, otherwise it is set to STALE.

- It sets the IsRouter flag in the cache entry based on the Router flag in the received advertisement.

- It sends any packets queued for the neighbor awaiting address resolution.

Note that the Override flag is ignored if the entry is in the INCOMPLETE state.

If the target's Neighbor Cache entry is in any state other than INCOMPLETE when the advertisement is received, processing becomes quite a bit more complex. If the Override flag is clear and the supplied link-layer address differs from that in the cache, then one of two actions takes place: if the state of the entry is REACHABLE,

set it to STALE, but do not update the entry in any other way; otherwise, the received advertisement should be ignored and MUST NOT update the cache. If the Override flag is set, both the Override flag is clear and the supplied link-layer address is the same as that in the cache, or no Target Link-layer address option was supplied, the received advertisement MUST update the Neighbor Cache entry as follows:

- The link-layer address in the Target Link-Layer Address option MUST be inserted in the cache (if one is supplied and is different than the already recorded address).

- If the Solicited flag is set, the state of the entry MUST be set to REACHABLE. If the Solicited flag is zero and the link-layer address was updated with a different address the state MUST be set to STALE. Otherwise, the entry's state remains unchanged.

  An advertisement's Solicited flag should only be set if the advertisement is a response to a Neighbor Solicitation. Because Neighbor Unreachability Detection Solicitations are sent to the cached link-layer address, receipt of a solicited advertisement indicates that the forward path is working. Receipt of an unsolicited advertisement, however, suggests that a neighbor has urgent information to announce (e.g., a changed link-layer address). If the urgent information indicates a change from what a node is currently using, the node should verify the reachability of the (new) path when it sends the next packet. There is no need to update the state for unsolicited advertisements that do not change the contents of the cache.

- The IsRouter flag in the cache entry MUST be set based on the Router flag in the received advertisement. In those cases where the IsRouter flag changes from TRUE to FALSE as a result of this update, the node MUST remove that router from the Default Router List and update the Destination Cache entries for all destinations using that neighbor as a router as specified in Section 7.3.3. This is needed to detect when a node that is used as a router stops forwarding packets due to being configured as a host.

The above rules ensure that the cache is updated either when the Neighbor Advertisement takes precedence (i.e., the Override flag is set) or when the Neighbor Advertisement refers to the same link-layer address that is currently recorded in the cache. If none of the above apply, the advertisement prompts future Neighbor Unreachability Detection (if it is not already in progress) by changing the state in the cache entry.

### 7.2.6. Sending Unsolicited Neighbor Advertisements

In some cases a node may be able to determine that its link-layer address has changed (e.g., hot-swap of an interface card) and may wish to inform its neighbors of the new link-layer address quickly. In such cases a node MAY send up to MAX_NEIGHBOR_ADVERTISEMENT unsolicited Neighbor Advertisement messages to the all-nodes multicast address. These advertisements MUST be separated by at least RetransTimer seconds.

The Target Address field in the unsolicited advertisement is set to an IP address of the interface, and the Target Link-Layer Address option is filled with the new link-layer address. The Solicited flag MUST be set to zero, in order to avoid confusing the Neighbor Unreachability Detection algorithm. If the node is a router, it MUST set the Router flag to one; otherwise it MUST set it to zero. The Override flag MAY be set to either zero or one. In either case, neighboring nodes will immediately change the state of their Neighbor Cache entries for the Target Address to STALE, prompting them to verify the path for reachability. If the Override flag is set to one, neighboring nodes will install the new link-layer address in their caches. Otherwise, they will ignore the new link-layer address, choosing instead to probe the cached address.

A node that has multiple IP addresses assigned to an interface MAY multicast a separate Neighbor Advertisement for each address. In such a case the node SHOULD introduce a small delay between the sending of each advertisement to reduce the probability of the advertisements being lost due to congestion.

A proxy MAY multicast Neighbor Advertisements when its link-layer address changes or when it is configured (by system management or other mechanisms) to proxy for an address. If there are multiple nodes that are providing proxy services for the same set of addresses the proxies SHOULD provide a mechanism that prevents multiple proxies from multicasting advertisements for any one address, in order to reduce the risk of excessive multicast traffic.

Also, a node belonging to an anycast address MAY multicast unsolicited Neighbor Advertisements for the anycast address when the node’s link-layer address changes.

Note that because unsolicited Neighbor Advertisements do not reliably update caches in all nodes (the advertisements might not be received by all nodes), they should only be viewed as a performance optimization to quickly update the caches in most neighbors. The Neighbor Unreachability Detection algorithm ensures that all nodes obtain a reachable link-layer address, though the delay may be

slightly longer.

### 7.2.7. Anycast Neighbor Advertisements

From the perspective of Neighbor Discovery, anycast addresses are treated just like unicast addresses in most cases. Because an anycast address is syntactically the same as a unicast address, nodes performing address resolution or Neighbor Unreachability Detection on an anycast address treat it as if it were a unicast address. No special processing takes place.

Nodes that have an anycast address assigned to an interface treat them exactly the same as if they were unicast addresses with two exceptions. First, Neighbor Advertisements sent in response to a Neighbor Solicitation SHOULD be delayed by a random time between 0 and MAX_ANYCAST_DELAY_TIME to reduce the probability of network congestion. Second, the Override flag in Neighbor Advertisements SHOULD be set to 0, so that when multiple advertisements are received, the first received advertisement is used rather than the most recently received advertisement.

As with unicast addresses, Neighbor Unreachability Detection ensures that a node quickly detects when the current binding for an anycast address becomes invalid.

### 7.2.8. Proxy Neighbor Advertisements

Under limited circumstances, a router MAY proxy for one or more other nodes, that is, through Neighbor Advertisements indicate that it is willing to accept packets not explicitly addressed to itself. For example, a router might accept packets on behalf of a mobile node that has moved off-link. The mechanisms used by proxy are identical to the mechanisms used with anycast addresses.

A proxy MUST join the solicited-node multicast address(es) that correspond to the IP address(es) assigned to the node for which it is proxying.

All solicited proxy Neighbor Advertisement messages MUST have the Override flag set to zero. This ensures that if the node itself is present on the link its Neighbor Advertisement (with the Override flag set to one) will take precedence of any advertisement received from a proxy. A proxy MAY send unsolicited advertisements with the Override flag set to one as specified in Section 7.2.6, but doing so may cause the proxy advertisement to override a valid entry created by the node itself.

Finally, when sending a proxy advertisement in response to a Neighbor Solicitation, the sender should delay its response by a random time between 0 and MAX_ANYCAST_DELAY_TIME seconds.

### 7.3. Neighbor Unreachability Detection

Communication to or through a neighbor may fail for numerous reasons at any time, including hardware failure, hot-swap of an interface card, etc. If the destination has failed, no recovery is possible and communication fails. On the other hand, if it is the path that has failed, recovery may be possible. Thus, a node actively tracks the reachability "state" for the neighbors to which it is sending packets.

Neighbor Unreachability Detection is used for all paths between hosts and neighboring nodes, including host-to-host, host-to-router, and router-to-host communication. Neighbor Unreachability Detection may also be used between routers, but is not required if an equivalent mechanism is available, for example, as part of the routing protocols.

When a path to a neighbor appears to be failing, the specific recovery procedure depends on how the neighbor is being used. If the neighbor is the ultimate destination, for example, address resolution should be performed again. If the neighbor is a router, however, attempting to switch to another router would be appropriate. The specific recovery that takes place is covered under next-hop determination; Neighbor Unreachability Detection signals the need for next-hop determination by deleting a Neighbor Cache entry.

Neighbor Unreachability Detection is performed only for neighbors to which unicast packets are sent; it is not used when sending to multicast addresses.

#### 7.3.1. Reachability Confirmation

A neighbor is considered reachable if the node has recently received a confirmation that packets sent recently to the neighbor were received by its IP layer. Positive confirmation can be gathered in two ways: hints from upper layer protocols that indicate a connection is making "forward progress", or receipt of a Neighbor Advertisement message that is a response to a Neighbor Solicitation message.

A connection makes "forward progress" if the packets received from a remote peer can only be arriving if recent packets sent to that peer are actually reaching it. In TCP, for example, receipt of a (new) acknowledgement indicates that previously sent data reached the peer. Likewise, the arrival of new (non-duplicate) data indicates that

earlier acknowledgements are being delivered to the remote peer. If packets are reaching the peer, they must also be reaching the sender's next-hop neighbor; thus "forward progress" is a confirmation that the next-hop neighbor is reachable. For off-link destinations, forward progress implies that the first-hop router is reachable. When available, this upper-layer information SHOULD be used.

In some cases (e.g., UDP-based protocols and routers forwarding packets to hosts) such reachability information may not be readily available from upper-layer protocols. When no hints are available and a node is sending packets to a neighbor, the node actively probes the neighbor using unicast Neighbor Solicitation messages to verify that the forward path is still working.

The receipt of a solicited Neighbor Advertisement serves as reachability confirmation, since advertisements with the Solicited flag set to one are sent only in response to a Neighbor Solicitation. Receipt of other Neighbor Discovery messages such as Router Advertisements and Neighbor Advertisement with the Solicited flag set to zero MUST NOT be treated as a reachability confirmation. Receipt of unsolicited messages only confirm the one-way path from the sender to the recipient node. In contrast, Neighbor Unreachability Detection requires that a node keep track of the reachability of the forward path to a neighbor from the its perspective, not the neighbor's perspective. Note that receipt of a solicited advertisement indicates that a path is working in both directions. The solicitation must have reached the neighbor, prompting it to generate an advertisement. Likewise, receipt of an advertisement indicates that the path from the sender to the recipient is working. However, the latter fact is known only to the recipient; the advertisement's sender has no direct way of knowing that the advertisement it sent actually reached a neighbor. From the perspective of Neighbor Unreachability Detection, only the reachability of the forward path is of interest.

### 7.3.2. Neighbor Cache Entry States

A Neighbor Cache entry can be in one of five states:

INCOMPLETE Address resolution is being performed on the entry. Specifically, a Neighbor Solicitation has been sent to the solicited-node multicast address of the target, but the corresponding Neighbor Advertisement has not yet been received.

REACHABLE Positive confirmation was received within the last ReachableTime milliseconds that the forward path to the neighbor was functioning properly. While

REACHABLE, no special action takes place as packets are sent.

STALE More than ReachableTime milliseconds have elapsed since the last positive confirmation was received that the forward path was functioning properly. While stale, no action takes place until a packet is sent.

The STALE state is entered upon receiving an unsolicited Neighbor Discovery message that updates the cached link-layer address. Receipt of such a message does not confirm reachability, and entering the STALE state insures reachability is verified quickly if the entry is actually being used. However, reachability is not actually verified until the entry is actually used.

DELAY More than ReachableTime milliseconds have elapsed since the last positive confirmation was received that the forward path was functioning properly, and a packet was sent within the last DELAY_FIRST_PROBE_TIME seconds. If no reachability confirmation is received within DELAY_FIRST_PROBE_TIME seconds of entering the DELAY state, send a Neighbor Solicitation and change the state to PROBE.

The DELAY state is an optimization that gives upper-layer protocols additional time to provide reachability confirmation in those cases where ReachableTime milliseconds have passed since the last confirmation due to lack of recent traffic. Without this optimization the opening of a TCP connection after a traffic lull would initiate probes even though the subsequent three-way handshake would provide a reachability confirmation almost immediately.

PROBE A reachability confirmation is actively sought by retransmitting Neighbor Solicitations every RetransTimer milliseconds until a reachability confirmation is received.

### 7.3.3. Node Behavior

Neighbor Unreachability Detection operates in parallel with the sending of packets to a neighbor. While reasserting a neighbor's reachability, a node continues sending packets to that neighbor using the cached link-layer address. If no traffic is sent to a neighbor, no probes are sent.

When a node needs to perform address resolution on a neighboring address, it creates an entry in the INCOMPLETE state and initiates address resolution as specified in Section 7.2. If address resolution fails, the entry SHOULD be deleted, so that subsequent traffic to that neighbor invokes the next-hop determination procedure again. Invoking next-hop determination at this point insures that alternate default routers are tried.

When a reachability confirmation is received (either through upper-layer advice or a solicited Neighbor Advertisement) an entry's state changes to REACHABLE. The one exception is that upper-layer advice has no effect on entries in the INCOMPLETE state (e.g., for which no link-layer address is cached).

When ReachableTime milliseconds have passed since receipt of the last reachability confirmation for a neighbor, the Neighbor Cache entry's state changes from REACHABLE to STALE.

> Note: An implementation may actually defer changing the state from REACHABLE to STALE until a packet is sent to the neighbor, i.e., there need not be an explicit timeout event associated with the expiration of ReachableTime.

The first time a node sends a packet to a neighbor whose entry is STALE, the sender changes the state to DELAY and a sets a timer to expire in DELAY_FIRST_PROBE_TIME seconds. If the entry is still in the DELAY state when the timer expires, the entry's state changes to PROBE. If reachability confirmation is received, the entry's state changes to REACHABLE.

Upon entering the PROBE state, a node sends a unicast Neighbor Solicitation message to the neighbor using the cached link-layer address. While in the PROBE state, a node retransmits Neighbor Solicitation messages every RetransTimer milliseconds until reachability confirmation is obtained. Probes are retransmitted even if no additional packets are sent to the neighbor. If no response is received after waiting RetransTimer milliseconds after sending the MAX_UNICAST_SOLICIT solicitations, retransmissions cease and the entry SHOULD be deleted. Subsequent traffic to that neighbor will recreate the entry and performs address resolution again.

Note that all Neighbor Solicitations are rate-limited on a per-neighbor basis. A node MUST NOT send Neighbor Solicitations to the same neighbor more frequently than once every RetransTimer milliseconds.

A Neighbor Cache entry enters the STALE state when created as a result of receiving packets other than solicited Neighbor Advertisements (i.e., Router Solicitations, Router Advertisements, Redirects, and Neighbor Solicitations). These packets contain the link-layer address of either the sender or, in the case of Redirect, the redirection target. However, receipt of these link-layer addresses does not confirm reachability of the forward-direction path to that node. Placing a newly created Neighbor Cache entry for which the link-layer address is known in the STALE state provides assurance that path failures are detected quickly. In addition, should a cached link-layer address be modified due to receiving one of the above messages the state SHOULD also be set to STALE to provide prompt verification that the path to the new link-layer address is working.

To properly detect the case where a router switches from being a router to being a host (e.g., if its IP forwarding capability is turned off by system management), a node MUST compare the Router flag field in all received Neighbor Advertisement messages with the IsRouter flag recorded in the Neighbor Cache entry. When a node detects that a neighbor has changed from being a router to being a host, the node MUST remove that router from the Default Router List and update the Destination Cache as described in Section 6.3.5. Note that a router may not be listed in the Default Router List, even though a Destination Cache entry is using it (e.g., a host was redirected to it). In such cases, all Destination Cache entries that reference the (former) router must perform next-hop determination again before using the entry.

In some cases, link-specific information may indicate that a path to a neighbor has failed (e.g., the resetting of a virtual circuit). In such cases, link-specific information may be used to purge Neighbor Cache entries before the Neighbor Unreachability Detection would do so. However, link-specific information MUST NOT be used to confirm the reachability of a neighbor; such information does not provide end-to-end confirmation between neighboring IP layers.

## 8. REDIRECT FUNCTION

This section describes the functions related to the sending and processing of Redirect messages.

Redirect messages are sent by routers to redirect a host to a better first-hop router for a specific destination or to inform hosts that a destination is in fact a neighbor (i.e., on-link). The latter is accomplished by having the ICMP Target Address be equal to the ICMP Destination Address.

A router MUST be able to determine the link-local address for each of its neighboring routers in order to ensure that the target address in a Redirect message identifies the neighbor router by its link-local address. For static routing this requirement implies that the next-hop router's address should be specified using the link-local address of the router. For dynamic routing this requirement implies that all IPv6 routing protocols must somehow exchange the link-local addresses of neighboring routers.

### 8.1. Validation of Redirect Messages

A host MUST silently discard any received Redirect message that does not satisfy all of the following validity checks:

- IP Source Address is a link-local address. Routers must use their link-local address as the source for Router Advertisement and Redirect messages so that hosts can uniquely identify routers.

- The IP Hop Limit field has a value of 255, i.e., the packet could not possibly have been forwarded by a router.

- If the message includes an IP Authentication Header, the message authenticates correctly.

- ICMP Checksum is valid.

- ICMP Code is 0.

- ICMP length (derived from the IP length) is 40 or more octets.

- The IP source address of the Redirect is the same as the current first-hop router for the specified ICMP Destination Address.

- The ICMP Destination Address field in the redirect message does not contain a multicast address.

- The ICMP Target Address is either a link-local address (when redirected to a router) or the same as the ICMP Destination Address (when redirected to the on-link destination).

- All included options have a length that is greater than zero.

The contents of the Reserved field, and of any unrecognized options MUST be ignored. Future, backward-compatible changes to the protocol may specify the contents of the Reserved field or add new options; backward-incompatible changes may use different Code values.

The contents of any defined options that are not specified to be used with Redirect messages MUST be ignored and the packet processed as normal. The only defined options that may appear are the Target Link-Layer Address option and the Redirected Header option.

A host MUST NOT consider a redirect invalid just because the Target Address of the redirect is not covered under one of the link's prefixes. Part of the semantics of the Redirect message is that the Target Address is on-link.

A redirect that passes the validity checks is called a "valid redirect".

## 8.2. Router Specification

A router SHOULD send a redirect message, subject to rate limiting, whenever it forwards a packet that is not explicitly addressed to itself (i.e. a packet that is not source routed through the router) in which:

- the Source Address field of the packet identifies a neighbor, and
- the router determines that a better first-hop node resides on the same link as the sending node for the Destination Address of the packet being forwarded, and
- the Destination Address of the packet is not a multicast address, and

The transmitted redirect packet contains, consistent with the message format given in Section 4.5:

- In the Target Address field: the address to which subsequent packets for the destination SHOULD be sent. If the target is a router, that router's link-local address MUST be used. If the target is a host the target address field MUST be set to the same value as the Destination Address field.
- In the Destination Address field: the destination address of the invoking IP packet.
- In the options:
    - o Target Link-Layer Address option: link-layer address of the target, if known.

- o Redirected Header: as much of the forwarded packet as can fit without the redirect packet exceeding 1280 octets in size.

A router MUST limit the rate at which Redirect messages are sent, in order to limit the bandwidth and processing costs incurred by the Redirect messages when the source does not correctly respond to the Redirects, or the source chooses to ignore unauthenticated Redirect messages. More details on the rate-limiting of ICMP error messages can be found in [ICMPv6].

A router MUST NOT update its routing tables upon receipt of a Redirect.

## 8.3. Host Specification

A host receiving a valid redirect SHOULD update its Destination Cache accordingly so that subsequent traffic goes to the specified target. If no Destination Cache entry exists for the destination, an implementation SHOULD create such an entry.

If the redirect contains a Target Link-Layer Address option the host either creates or updates the Neighbor Cache entry for the target. In both cases the cached link-layer address is copied from the Target Link-Layer Address option. If a Neighbor Cache entry is created for the target its reachability state MUST be set to STALE as specified in Section 7.3.3. If a cache entry already existed and it is updated with a different link-layer address, its reachability state MUST also be set to STALE. If the link-layer address is the same as that already in the cache, the cache entry's state remains unchanged.

If the Target and Destination Addresses are the same, the host MUST treat the Target as on-link. If the Target Address is not the same as the Destination Address, the host MUST set IsRouter to TRUE for the target. If the Target and Destination Addresses are the same, however, one cannot reliably determine whether the Target Address is a router. Consequently, newly created Neighbor Cache entries should set the IsRouter flag to FALSE, while existing cache entries should leave the flag unchanged. If the Target is a router, subsequent Neighbor Advertisement or Router Advertisement messages will update IsRouter accordingly.

Redirect messages apply to all flows that are being sent to a given destination. That is, upon receipt of a Redirect for a Destination Address, all Destination Cache entries to that address should be updated to use the specified next-hop, regardless of the contents of the Flow Label field that appears in the Redirected Header option.

A host MAY have a configuration switch that can be set to make it ignore a Redirect message that does not have an IP Authentication header.

A host MUST NOT send Redirect messages.

## 9. EXTENSIBILITY - OPTION PROCESSING

Options provide a mechanism for encoding variable length fields, fields that may appear multiple times in the same packet, or information that may not appear in all packets. Options can also be used to add additional functionality to future versions of ND.

In order to ensure that future extensions properly coexist with current implementations, all nodes MUST silently ignore any options they do not recognize in received ND packets and continue processing the packet. All options specified in this document MUST be recognized. A node MUST NOT ignore valid options just because the ND message contains unrecognized ones.

The current set of options is defined in such a way that receivers can process multiple options in the same packet independently of each other. In order to maintain these properties future options SHOULD follow the simple rule:

> The option MUST NOT depend on the presence or absence of any other options. The semantics of an option should depend only on the information in the fixed part of the ND packet and on the information contained in the option itself.

Adhering to the above rule has the following benefits:

1) Receivers can process options independently of one another. For example, an implementation can choose to process the Prefix Information option contained in a Router Advertisement message in a user-space process while the link-layer address option in the same message is processed by routines in the kernel.

2) Should the number of options cause a packet to exceed a link's MTU, multiple packets can carry subsets of the options without any change in semantics.

3) Senders MAY send a subset of options in different packets. For instance, if a prefix's Valid and Preferred Lifetime are high enough, it might not be necessary to include the Prefix Information option in every Router Advertisement. In addition, different routers might send different sets of options. Thus, a receiver MUST NOT associate any action with the absence of an

option in a particular packet. This protocol specifies that receivers should only act on the expiration of timers and on the information that is received in the packets.

Options in Neighbor Discovery packets can appear in any order; receivers MUST be prepared to process them independently of their order. There can also be multiple instances of the same option in a message (e.g., Prefix Information options).

If the number of included options in a Router Advertisement causes the advertisement's size to exceed the link MTU, the router can send multiple separate advertisements each containing a subset of the options.

The amount of data to include in the Redirected Header option MUST be limited so that the entire redirect packet does not exceed 1280 octets.

All options are a multiple of 8 octets of length, ensuring appropriate alignment without any "pad" options. The fields in the options (as well as the fields in ND packets) are defined to align on their natural boundaries (e.g., a 16-bit field is aligned on a 16-bit boundary) with the exception of the 128-bit IP addresses/prefixes, which are aligned on a 64-bit boundary. The link-layer address field contains an uninterpreted octet string; it is aligned on an 8-bit boundary.

The size of an ND packet including the IP header is limited to the link MTU (which is at least 1280 octets). When adding options to an ND packet a node MUST NOT exceed the link MTU.

Future versions of this protocol may define new option types. Receivers MUST silently ignore any options they do not recognize and continue processing the message.

## 10. PROTOCOL CONSTANTS

Router constants:

| | |
|---|---|
| MAX_INITIAL_RTR_ADVERT_INTERVAL | 16 seconds |
| MAX_INITIAL_RTR_ADVERTISEMENTS | 3 transmissions |
| MAX_FINAL_RTR_ADVERTISEMENTS | 3 transmissions |
| MIN_DELAY_BETWEEN_RAS | 3 seconds |
| MAX_RA_DELAY_TIME | .5 seconds |

Host constants:

| | |
|---|---|
| MAX_RTR_SOLICITATION_DELAY | 1 second |
| RTR_SOLICITATION_INTERVAL | 4 seconds |
| MAX_RTR_SOLICITATIONS | 3 transmissions |

Node constants:

| | |
|---|---|
| MAX_MULTICAST_SOLICIT | 3 transmissions |
| MAX_UNICAST_SOLICIT | 3 transmissions |
| MAX_ANYCAST_DELAY_TIME | 1 second |
| MAX_NEIGHBOR_ADVERTISEMENT | 3 transmissions |
| REACHABLE_TIME | 30,000 milliseconds |
| RETRANS_TIMER | 1,000 milliseconds |
| DELAY_FIRST_PROBE_TIME | 5 seconds |
| MIN_RANDOM_FACTOR | .5 |
| MAX_RANDOM_FACTOR | 1.5 |

Additional protocol constants are defined with the message formats in Section 4.

All protocol constants are subject to change in future revisions of the protocol.

The constants in this specification may be overridden by specific documents that describe how IPv6 operates over different link layers. This rule allows Neighbor Discovery to operate over links with widely varying performance characteristics.

## 11. SECURITY CONSIDERATIONS

Neighbor Discovery is subject to attacks that cause IP packets to flow to unexpected places. Such attacks can be used to cause denial of service but also allow nodes to intercept and optionally modify packets destined for other nodes.

The protocol reduces the exposure to such threats in the absence of authentication by ignoring ND packets received from off-link senders. The Hop Limit field of all received packets is verified to contain 255, the maximum legal value. Because routers decrement the Hop Limit on all packets they forward, received packets containing a Hop Limit of 255 must have originated from a neighbor.

An example of denial of service attacks is that a node on the link that can send packets with an arbitrary IP source address can both advertise itself as a default router and also send "forged" Router Advertisement messages that immediately time out all other default routers as well as all on-link prefixes. An intruder can achieve this by sending out multiple Router Advertisements, one for each legitimate router, with the source address set to the address of another router, the Router Lifetime field set to zero, and the Preferred and Valid lifetimes set to zero for all the prefixes. Such an attack would cause all packets, for both on-link and off-link destinations, to go to the rogue router. That router can then selectively examine, modify or drop all packets sent on the link. The Neighbor Unreachability Detection will not detect such a black hole as long as the rogue router politely answers the NUD probes with a Neighbor Advertisement with the R-bit set.

Many link layers are also subject to different denial of service attacks such as continuously occupying the link in CSMA/CD networks (e.g., by sending packets closely back-to-back or asserting the collision signal on the link), or originating packets with somebody else’s source MAC address to confuse, e.g., Ethernet switches.

The trust model for redirects is the same as in IPv4. A redirect is accepted only if received from the same router that is currently being used for that destination. It is natural to trust the routers on the link. If a host has been redirected to another node (i.e., the destination is on-link) there is no way to prevent the target from issuing another redirect to some other destination. However, this exposure is no worse than it was; the target host, once subverted, could always act as a hidden router to forward traffic elsewhere.

The protocol contains no mechanism to determine which neighbors are authorized to send a particular type of message (e.g., Router Advertisements); any neighbor, presumably even in the presence of authentication, can send Router Advertisement messages thereby being able to cause denial of service. Furthermore, any neighbor can send proxy Neighbor Advertisements as well as unsolicited Neighbor Advertisements as a potential denial of service attack.

Neighbor Discovery protocol packet exchanges can be authenticated using the IP Authentication Header [IPv6-AUTH]. A node SHOULD include an Authentication Header when sending Neighbor Discovery packets if a security association for use with the IP Authentication Header exists for the destination address. The security associations may have been created through manual configuration or through the operation of some key management protocol.

Received Authentication Headers in Neighbor Discovery packets MUST be verified for correctness and packets with incorrect authentication MUST be ignored.

It SHOULD be possible for the system administrator to configure a node to ignore any Neighbor Discovery messages that are not authenticated using either the Authentication Header or Encapsulating Security Payload. The configuration technique for this MUST be documented. Such a switch SHOULD default to allowing unauthenticated messages.

Confidentiality issues are addressed by the IP Security Architecture and the IP Encapsulating Security Payload documents [IPv6-SA, IPv6-ESP].

## 12. RENUMBERING CONSIDERATIONS

The Neighbor Discovery protocol together with IPv6 Address Autoconfiguration [ADDRCONF] provides mechanisms to aid in renumbering - new prefixes and addresses can be introduced and old ones can be deprecated and removed.

The robustness of these mechanisms is based on all the nodes on the link receiving the Router Advertisement messages in a timely manner. However, a host might be turned off or be unreachable for an extended period of time (i.e., a machine is powered down for months after a project terminates). It is possible to preserve robust renumbering in such cases but it does place some constraints on how long prefixes must be advertised.

Consider the following example in which a prefix is initially advertised with a lifetime of 2 months, but on August 1st it is determined that the prefix needs to be deprecated and removed due to renumbering by September 1st. This can be done by reducing the advertised lifetime to 1 week starting on August 1st and as the cutoff gets closer the lifetimes can be made shorter until by September 1st the prefix is advertised with a zero lifetime. The point is that, if one or more nodes were unplugged from the link prior to September 1st they might still think that the prefix is valid since the last lifetime they received was 2 months. Thus if a

node was unplugged on July 31st it thinks the prefix is valid until September 30th. If that node is plugged back in prior to September 30th it may continue to use the old prefix. The only way to force a node to stop using a prefix that was previously advertised with a long Lifetime is to have that node receive an advertisement for that prefix that changes the lifetime downward. The solution in this example is simple: continue advertising the prefix with a lifetime of 0 from September 1st until October 1st.

In general, in order to be robust against nodes that might be unplugged from the link it is important to track the furthest into the future a particular prefix can be viewed valid by any node on the link. The prefix must then be advertised with a 0 Lifetime until that point in future. This "furthest into the future" time is simply the maximum, over all Router Advertisements, of the time the advertisement was sent plus the prefix's Lifetime contained in the advertisement.

The above has an important implication on using infinite lifetimes. If a prefix is advertised with an infinite lifetime, and that prefix later needs to be renumbered, it is undesirable to continue advertising that prefix with a zero lifetime forever. Thus either infinite lifetimes should be avoided or there must be a limit on how long time a node can be unplugged from the link before it is plugged back in again. However, it is unclear how the network administrator can enforce a limit on how long time hosts such as laptops can be unplugged from the link.

Network administrators should give serious consideration to using relatively short lifetimes (i.e., no more than a few weeks). While it might appear that using long lifetimes would help insure robustness, in reality a host will be unable to communicate in the absence of properly functioning routers. Such routers will be sending Router Advertisements that contain appropriate (and current) prefixes. A host connected to a network that has no functioning routers is likely to have more serious problems than just a lack of a valid prefix and address.

The above discussion does not distinguish between the preferred and valid lifetimes. For all practical purposes it is probably sufficient to track the valid lifetime since the preferred lifetime will not exceed the valid lifetime.

REFERENCES

[ADDRCONF] Thomson, S. and T. Narten, "IPv6 Address Autoconfiguration", RFC 2462, December 1998.

[ADDR-ARCH] Hinden, R. and S. Deering, "IP Version 6 Addressing Architecture", RFC 2373, July 1998.

[ANYCST] Partridge, C., Mendez, T. and W. Milliken, "Host Anycasting Service", RFC 1546, November 1993.

[ARP] Plummer, D., "An Ethernet Address Resolution Protocol", STD 37, RFC 826, November 1982.

[HR-CL] Braden, R., Editor, "Requirements for Internet Hosts -- Communication Layers", STD 3, RFC 1122, October 1989.

[ICMPv4] Postel, J., "Internet Control Message Protocol", STD 5, RFC 792, September 1981.

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

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

[IPv6-ETHER] Crawford, M., "Transmission of IPv6 Packets over Ethernet Networks", RFC 2464, December 1998.

[IPv6-SA] Kent, S. and R. Atkinson, "Security Architecture for the Internet Protocol", RFC 2401, November 1998.

[IPv6-AUTH] Kent, S. and R. Atkinson, "IP Authentication Header", RFC 2402, November 1998.

[IPv6-ESP] Kent, S. and R. Atkinson, "IP Encapsulating Security Payload (ESP)", RFC 2406, November 1998.

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

[RDISC] Deering, S., "ICMP Router Discovery Messages", RFC 1256, September 1991.

[SH-MEDIA] Braden, R., Postel, J. and Y. Rekhter, "Internet Architecture Extensions for Shared Media", RFC 1620, May 1994.

[ASSIGNED] Reynolds, J. and J. Postel, "ASSIGNED NUMBERS", STD 2, RFC 1700, October 1994. See also: http://www.iana.org/numbers.html

[SYNC] S. Floyd, V. Jacobson, "The Synchronization of Periodic Routing Messages", IEEE/ACM Transactions on Networking, April 1994. ftp://ftp.ee.lbl.gov/papers/sync_94.ps.Z

## Authors' Addresses

Thomas Narten
IBM Corporation
P.O. Box 12195
Research Triangle Park, NC 27709-2195
USA

Phone: +1 919 254 7798
EMail: narten@raleigh.ibm.com

Erik Nordmark
Sun Microsystems, Inc.
901 San Antonio Road
Palo Alto, CA 94303
USA

Phone: +1 650 786 5166
Fax: +1 650 786 5896
EMail: nordmark@sun.com

William Allen Simpson
Daydreamer
Computer Systems Consulting Services
1384 Fontaine
Madison Heights, Michigan 48071
USA

EMail: Bill.Simpson@um.cc.umich.edu
bsimpson@MorningStar.com

## APPENDIX A: MULTIHOMED HOSTS

There are a number of complicating issues that arise when Neighbor Discovery is used by hosts that have multiple interfaces. This section does not attempt to define the proper operation of multihomed hosts with regard to Neighbor Discovery. Rather, it identifies issues that require further study. Implementors are encouraged to experiment with various approaches to making Neighbor Discovery work on multihomed hosts and to report their experiences.

If a multihomed host receives Router Advertisements on all of its interfaces, it will (probably) have learned on-link prefixes for the addresses residing on each link. When a packet must be sent through a router, however, selecting the "wrong" router can result in a suboptimal or non-functioning path. There are number of issues to consider:

1) In order for a router to send a redirect, it must determine that the packet it is forwarding originates from a neighbor. The standard test for this case is to compare the source address of the packet to the list of on-link prefixes associated with the interface on which the packet was received. If the originating host is multihomed, however, the source address it uses may belong to an interface other than the interface from which it was sent. In such cases, a router will not send redirects, and suboptimal routing is likely. In order to be redirected, the sending host must always send packets out the interface corresponding to the outgoing packet's source address. Note that this issue never arises with non-multihomed hosts; they only have one interface.

2) If the selected first-hop router does not have a route at all for the destination, it will be unable to deliver the packet. However, the destination may be reachable through a router on one of the other interfaces. Neighbor Discovery does not address this scenario; it does not arise in the non-multihomed case.

3) Even if the first-hop router does have a route for a destination, there may be a better route via another interface. No mechanism exists for the multihomed host to detect this situation.

If a multihomed host fails to receive Router Advertisements on one or more of its interfaces, it will not know (in the absence of configured information) which destinations are on-link on the affected interface(s). This leads to a number of problems:

1) If no Router Advertisement is received on any interfaces, a multihomed host will have no way of knowing which interface to send packets out on, even for on-link destinations. Under similar conditions in the non-multihomed host case, a node treats all destinations as residing on-link, and communication proceeds. In the multihomed case, however, additional information is needed to select the proper outgoing interface. Alternatively, a node could attempt to perform address resolution on all interfaces, a step involving significant complexity that is not present in the non-multihomed host case.

2) If Router Advertisements are received on some, but not all interfaces, a multihomed host could choose to only send packets out on the interfaces on which it has received Router Advertisements. A key assumption made here, however, is that routers on those other interfaces will be able to route packets to the ultimate destination, even when those destinations reside on the subnet to which the sender connects, but has no on-link prefix information. Should the assumption be FALSE, communication would fail. Even if the assumption holds, packets will traverse a sub-optimal path.

## APPENDIX B: FUTURE EXTENSIONS

Possible extensions for future study are:

- Using dynamic timers to be able to adapt to links with widely varying delay. Measuring round trip times, however, requires acknowledgments and sequence numbers in order to match received Neighbor Advertisements with the actual Neighbor Solicitation that triggered the advertisement. Implementors wishing to experiment with such a facility could do so in a backwards-compatible way by defining a new option carrying the necessary information. Nodes not understanding the option would simply ignore it.

- Adding capabilities to facilitate the operation over links that currently require hosts to register with an address resolution server. This could for instance enable routers to ask hosts to send them periodic unsolicited advertisements. Once again this can be added using a new option sent in the Router Advertisements.

- Adding additional procedures for links where asymmetric and non-transitive reachability is part of normal operations. Such procedures might allow hosts and routers to find usable paths on, e.g., radio links.

## APPENDIX C: STATE MACHINE FOR THE REACHABILITY STATE

This appendix contains a summary of the rules specified in Sections 7.2 and 7.3. This document does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this document.

When performing address resolution and Neighbor Unreachability Detection the following state transitions apply using the conceptual model:

| State | Event | Action | New state |
|---|---|---|---|
| - | Packet to send. | Create entry.<br>Send multicast NS.<br>Start retransmit timer | INCOMPLETE |
| INCOMPLETE | Retransmit timeout, less than N retransmissions. | Retransmit NS<br>Start retransmit timer | INCOMPLETE |
| INCOMPLETE | Retransmit timeout, N or more retransmissions. | Discard entry<br>Send ICMP error | - |
| INCOMPLETE | NA, Solicited=0, Override=any | Record link-layer address. Send queued packets. | STALE |
| INCOMPLETE | NA, Solicited=1, Override=any | Record link-layer address. Send queued packets. | REACHABLE |
| !INCOMPLETE | NA, Solicited=1, Override=0<br>Same link-layer address as cached. | - | REACHABLE |
| REACHABLE | NA, Solicited=1, Override=0<br>Different link-layer address than cached. | - | STALE |
| STALE or PROBE | NA, Solicited=1, Override=0<br>Different link-layer address than cached. | - | unchanged |

| | | | |
|---|---|---|---|
| !INCOMPLETE | NA, Solicited=1, Override=1 | Record link-layer address (if different). | REACHABLE |
| !INCOMPLETE | NA, Solicited=0, Override=0 | - | unchanged |
| !INCOMPLETE | NA, Solicited=0, Override=1 Same link-layer address as cached. | - | unchanged |
| !INCOMPLETE | NA, Solicited=0, Override=1 Different link-layer address than cached. | Record link-layer address. | STALE |
| !INCOMPLETE | upper-layer reachability confirmation | - | REACHABLE |
| REACHABLE | timeout, more than N seconds since reachability confirm. | - | STALE |
| STALE | Sending packet | Start delay timer | DELAY |
| DELAY | Delay timeout | Send unicast NS probe Start retransmit timer | PROBE |
| PROBE | Retransmit timeout, less than N retransmissions. | Retransmit NS | PROBE |
| PROBE | Retransmit timeout, N or more retransmissions. | Discard entry | - |

The state transitions for receiving unsolicited information other than Neighbor Advertisement messages apply to either the source of the packet (for Neighbor Solicitation, Router Solicitation, and Router Advertisement messages) or the target address (for Redirect messages) as follows:

| State | Event | Action | New state |
|---|---|---|---|
| - | NS, RS, RA, Redirect | Create entry. | STALE |
| INCOMPLETE | NS, RS, RA, Redirect | Record link-layer address. Send queued packets. | STALE |
| !INCOMPLETE | NS, RS, RA, Redirect Different link-layer address than cached. | Update link-layer address | STALE |
| !INCOMPLETE | NS, RS, RA, Redirect Same link-layer address as cached. | - | unchanged |

## APPENDIX D: SUMMARY OF ISROUTER RULES

This appendix presents a summary of the rules for maintaining the IsRouter flag as specified in this document.

The background for these rules is that the ND messages contain, either implicitly or explicitly, information that indicates whether or not the sender (or Target Address) is a host or a router. The following assumptions are used:

- The sender of a Router Solicitation is implicitly assumed to be a host since there is no need for routers to send such messages.

- The sender of a Router Advertisement is implicitly assumed to be a router.

- Neighbor Solicitation messages do not contain either an implicit or explicit indication about the sender. Both hosts and routers send such messages.

  - Neighbor Advertisement messages contain an explicit "IsRouter flag", the R-bit.

- The target of the redirect, when the target differs from the destination address in the packet being redirected, is implicitly assumed to be a router. This is a natural assumption since that node is expected to be able to forward the packets towards the destination.

- The target of the redirect, when the target is the same as the destination, does not carry any host vs. router information. All that is known is that the destination (i.e. target) is on-link but it could be either a host or a router.

The rules for setting the IsRouter flag are based on the information content above. If an ND message contains explicit or implicit information the receipt of the message will cause the IsRouter flag to be updated. But when there is no host vs. router information in the ND message the receipt of the message MUST NOT cause a change to the IsRouter state. When the receipt of such a message causes a Neighbor Cache entry to be created this document specifies that the IsRouter flag be set to FALSE. There is greater potential for mischief when a node incorrectly thinks a host is a router, than the other way around. In these cases a subsequent Neighbor Advertisement or Router Advertisement message will set the correct IsRouter value.

## APPENDIX E: IMPLEMENTATION ISSUES

### Appendix E.1: Reachability confirmations

Neighbor Unreachability Detection requires explicit confirmation that a forward-path is functioning properly. To avoid the need for Neighbor Solicitation probe messages, upper layer protocols should provide such an indication when the cost of doing so is small. Reliable connection-oriented protocols such as TCP are generally aware when the forward-path is working. When TCP sends (or receives) data, for instance, it updates its window sequence numbers, sets and cancels retransmit timers, etc. Specific scenarios that usually indicate a properly functioning forward-path include:

- Receipt of an acknowledgement that covers a sequence number (e.g., data) not previously acknowledged indicates that the forward path was working at the time the data was sent.

- Completion of the initial three-way handshake is a special case of the previous rule; although no data is sent during the handshake, the SYN flags are counted as data from the sequence number perspective. This applies to both the SYN+ACK for the active open the ACK of that packet on the passively opening peer.

- Receipt of new data (i.e., data not previously received) indicates that the forward-path was working at the time an acknowledgement was sent that advanced the peer’s send window that allowed the new data to be sent.

To minimize the cost of communicating reachability information between the TCP and IP layers, an implementation may wish to rate-limit the reachability confirmations its sends IP. One possibility is to process reachability only every few packets. For example, one might update reachability information once per round trip time, if an implementation only has one round trip timer per connection. For those implementations that cache Destination Cache entries within control blocks, it may be possible to update the Neighbor Cache entry directly (i.e., without an expensive lookup) once the TCP packet has been demultiplexed to its corresponding control block. For other implementation it may be possible to piggyback the reachability confirmation on the next packet submitted to IP assuming that the implementation guards against the piggybacked confirmation becoming stale when no packets are sent to IP for an extended period of time.

TCP must also guard against thinking "stale" information indicates current reachability. For example, new data received 30 minutes after a window has opened up does not constitute a confirmation that the path is currently working. In merely indicates that 30 minutes

ago the window update reached the peer i.e. the path was working at that point in time. An implementation must also take into account TCP zero-window probes that are sent even if the path is broken and the window update did not reach the peer.

For UDP based applications (RPC, DNS) it is relatively simple to make the client send reachability confirmations when the response packet is received. It is more difficult and in some cases impossible for the server to generate such confirmations since there is no flow control, i.e., the server can not determine whether a received request indicates that a previous response reached the client.

Note that an implementation can not use negative upper-layer advise as a replacement for the Neighbor Unreachability Detection algorithm. Negative advise (e.g. from TCP when there are excessive retransmissions) could serve as a hint that the forward path from the sender of the data might not be working. But it would fail to detect when the path from the receiver of the data is not functioning causing, none of the acknowledgement packets to reach the sender.

## APPENDIX F: CHANGES SINCE RFC 1970

- Removed all references to the IPv6 priority field.
- Replaced definition of solicited node multicast address with a reference to the [ADDR-ARCH] specification. That specification says that "the solicited-node multicast address is formed by taking the low-order 24 bits of the address (unicast or anycast) and appending those bits to the prefix FF02:0:0:0:0:1:FF00::/104".
- Updated the references section to list (new) RFC numbers.
- Updated the text in section 7.2.5 and the tables in appendix C to have the receipt of an NS message update the state of an existing neighbor cache entry only if the link-layer address is different than the recorded link-layer address.
- Added an explicit check in section 7.1.1 so that received NS messages from an unsolicited address must be sent the solicited-node multicast address; if sent to unicast destination, silently discard.
- Added a requirement in section 6.2.1 that Lifetimes be configurable in either of two ways: as a fixed value that doesn't change over time, or one that decrements in real time.
- Added text in section 6.2.7 to relax the consistency checks on prefix lifetimes when the lifetimes are configured to decrement in real time. This is needed to avoid false alarms due to link propagation delay and lack of synchronized clocks.
- Added text to section 6.3.4 to point out that [ADDRCONF] might ignore short lifetimes but that Neighbor Discovery does not ignore short prefix lifetimes.
- Clarified the rules for RS and NS packets with an unspecified source address. Such packets MUST NOT include source link-layer address option; verified by receivers.
- Clarified in section 7.2.3 that addresses for which the node proxies are acceptable in NS messages. Previously the text only mentioned unicast and anycast addresses assigned to the interface (i.e., wasn't clear that proxy addresses were allowed).
- Tightened up ambiguities an inconsistencies regarding when to set the IsRouter flag in Neighbor Cache entries. Added an appendix to summarize these rules.

- Added a section on renumbering considerations to clarify how long prefixes have to be advertised when the lifetime(s) are reduced.

- Added additional text to the rules in section 7 for the NS/NA packets used for NUD probes so that the Link-Layer Address options can be omitted from these packets in certain cases without causing an infinite NS "recursion". Specifically, added text that permits the Link-Layer address to be omitted in unicast solicitations (i.e., MAY language).

- Changed the default AdvValidLifetime from infinity to 30 days.

- Changed the constant "576" to "1280" in places where its context was that of the minimum sized IP packet that all links must be able to carry.

Full Copyright Statement

Copyright (C) The Internet Society (1998). All Rights Reserved.

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.