Network Working Group
Internet Draft
expires in six months

W A Simpson
Daydreamer
February 1995

IPv6 Neighbor Discovery -- Processing
draft-simpson-ipv6-discov-process-02.txt |

Status of this Memo

This document is a submission to the IPng Working Group of the Internet Engineering Task Force (IETF). Comments should be submitted to the ipng@sunroof.eng.sun.com mailing list.

Distribution of this memo is unlimited.

This document is an Internet-Draft. Internet Drafts are working documents of the Internet Engineering Task Force (IETF), its Areas, and its Working Groups. Note that other groups may also distribute working documents as Internet Drafts.

Internet Drafts are draft documents valid for a maximum of six months, and may be updated, replaced, or obsoleted by other documents at any time. It is not appropriate to use Internet Drafts as reference material, or to cite them other than as a ``working draft'' or ``work in progress.''

To learn the current status of any Internet-Draft, please check the ``1id-abstracts.txt'' listing contained in the internet-drafts Shadow | Directories on: |

ds.internic.net (US East Coast)
nic.nordu.net (Europe)
ftp.isi.edu (US West Coast)
munnari.oz.au (Pacific Rim)

Abstract

This document discusses the implementation techniques for identification of and forwarding to adjacent IPv6 nodes, including Next Hop Determination and Router Discovery.

1. Introduction

This document describes how to

- determine the availability of other IPv6 nodes as demand for communication occurs;

- detect the presence of available IPv6 routers;

- learn the appropriate link address for sending to its neighbors;

- and redirect traffic where appropriate.

The design requirements are more completely described in [D-Sign]. The ICMP packet formats are described in [D-Form].

This document contains only that information which is particularly relevant to IPv6 Neighbor Discovery, or that differs from IPv4 techniques. Other IPv6 documents should be consulted for further details.

2. Link-Layers

This document anticipates that link-layer material will be covered in a separate Link Layer Requirements document. Specific link-layer protocol implementation details are beyond the scope of this document.

2.1. Addresses

For multi-access links, every node requires a unique address on the link.

When multi-homed hosts are present, every link address MUST be unique over all such attached links.

The same link address MAY be used by the same node on multiple links.

2.2. Address Resolution Protocol (ARP)

ARP [RFC-826] is no longer used for IPv6.

2.3. Trailers

Because ARP is no longer used for IPv6, trailer encapsulation [RFC-893] MUST NOT be used.

2.4. Maximum Transmission Unit (MTU)

The MTU is a internetwork-layer indication of the maximum datagram size which can be sent on an interface. This does not include link-layer encapsulation and framing, but includes padding which can be added by the link-layer for small frames. |

Many link-layer protocols define a maximum frame size that may be sent. In such cases, a node MUST NOT allow a datagram to be sent which would require frames larger than those allowed by the link-layer protocol.

The MTU of each logical interface MUST be configurable, as limited by the link-layer frame size. However, a node MUST be able to receive a packet as large as the maximum frame size, even if that is larger than the currently configured MTU. |

2.5. Maximum Receive Unit (MRU)

The MRU is a internetwork-layer indication of maximum datagram size that can be received by a peer. This does not include link-layer encapsulation and framing, but includes padding which can be added by the link-layer for small frames. |

Some link-layer protocols [RFC-1661] define a mechanism for adjusting the maximum frame size. In addition, this protocol provides the advertisement of an MRU independently of the link-layer.

When the advertised MRU of a peer node is less than the configured link MTU, that MRU MUST be maintained on a per peer basis.

2.6. Incoming Interface

On receipt of a link-layer unicast, broadcast or multicast packet, the node MAY check against a list of valid link addresses. If the packet passes the link-layer, then internetwork-layer will ensure the validity of the datagram. | | |

For each received packet, the link-layer MUST pass the following information to the internetwork-layer:

(1) the datagram itself.

(2) the length of the data portion of the link-layer frame, including padding (not including encapsulation and framing). |

(3) the identity of the physical interface from which the datagram was received.

(4) the classification of the received destination link-layer address as unicast or multicast (including broadcast). |

In addition, the link-layer SHOULD provide:

(5) the source link-layer address, if any.

RATIONALE:
This section is included because other parts of this document require specific information to be passed across this layer boundary.

Although every different medium typically has a different address format, the broadcast and multicast addresses are an important special case.

Some link adapters check only a coarse grained hash or suffix of the link destination. It is the responsiblity of the interface implementation to prevent leaking. See also "Processing Datagrams".

The source link address might be required in some implementations to map an essentially transient link-layer address (such as, a Frame-Relay DLCI) to the more stable family link address (that is, E.164). |

2.7. Outgoing Interface

For each transmitted packet, the internetwork-layer MUST pass the following information to the link-layer:

(1) the datagram itself.

(2) the length of the datagram.

(3) the destination physical interface.

(4) the destination link-layer address, if any.

In addition, the internetwork-layer SHOULD provide:

(5) the link-layer priority value.

The link-layer MUST notify the internetwork-layer if the packet to be transmitted causes a link-layer precedence-related error.

2.8. Unreachable

The link-layer MUST NOT report a Destination Unreachable error to the internetwork-layer solely because there is no Hop Cache entry for a particular Destination.

3. Sending Datagrams

For outgoing datagrams, the internetwork-layer:

(1) sets any fields not set by the transport-layer.

(2) chooses the interface and next hop.

(3) fragments the datagram if necessary, when intentional fragmentation is configured.

(4) passes the packet(s) to the appropriate link-layer interface.

3.1. Choosing a Source Address

When a node sends a datagram, the IPv6 Source MUST be one of its own IPv6 Unicast Addresses (not an IPv6 Cluster or Multicast Address).

NOTE:
This process is essentially the same as choosing the next hop (see "Next Hop Decision"). Implementors might be able to combine the two functions.

If the datagram is sent in response to a received datagram, the Destination from that datagram SHOULD be used as the Source for the response (unless it was an IPv6 Unspecified, Cluster or Multicast Address).

An application MUST be able to explicitly specify the Source for initiating a connection or a request.

If the Source to be used is unknown, a Source MUST be selected by the internetwork-layer.

(a) When no Router Advertisements have been heard, the Destination is assumed to be on an attached link.

The Source is chosen from an IPv6 Unicast Address which is bound to any interface.

(b) When one or more Router Advertisements have been heard, the Router List is examined.

If the Destination exactly matches the Primary Identifier of a router, a Prefix-Information entension, or an Known-Identifier extension, then the Source is chosen from the interface on

which the advertisement was received.

(c) The Destination is compared against the routing-prefixes configured for each interface, or learned from Prefix-Information and Known-Identifier extensions.

If the Destination exactly matches one of the routing-prefixes, then the Source is chosen from the indicated interface.

(d) If the Destination does not match any routing-prefixes, then the Source is chosen from the interface of the most preferred router, as described in the "Router Selection" section which follows.

When more than one IPv6 Unicast Address meets the criteria, the Source chosen SHOULD have the longest bit-wise prefix match with the Destination.

Other selection preferences are implementation dependent.

3.2. Hop Cache

To efficiently send a series of datagrams to the same Destination, each node MUST keep a cache of prior decisions, indexed by Destination.

The cache entry MUST include the link address (if any) to be used to send the datagram. This entry might point directly to the Destination, or to a router which forwards to the Destination. It MAY contain other information which records previous experience related to the Destination.

If the cache contains no information for a particular Destination, a determination is made where to send the datagram. This is described in the "Next Hop Decision" section which follows.

3.3. Next Hop Decision

The next hop is chosen based on the IPv6 Destination. If the Destination can be readily determined to be on an attached link, the datagram is sent directly to the Destination.

To determine the next hop, the following algorithm is used:

(a) If the Destination is the IPv6 Loopback Address, or an IPv6 Multicast Address with scope intra-node, process as an incoming datagram.

(b) If the Destination is another scope of IPv6 Multicast Address, simply pass the datagram to the link-layer for any indicated interface(s).

This requires that multicast datagrams are registered on a per interface basis. Other aspects of multicast routing are beyond the scope of this document.

(c) When no Router Advertisements have been heard, the Destination is assumed to be on an attached link.

The datagram is duplicated for each interface. Each interface separately solicits the Destination location, as described in "Sending General Solicitations". The correct interface will be learned when the Hop Cache is updated through the future advertisements of the target node.

(d) When one or more Router Advertisements have been heard, the Router List is examined.

If the Destination exactly matches the Primary Identifier of a router, a Prefix-Information entension, or an Known-Identifier extension, then the datagram is sent directly to the indicated router (using the Media-Access extension provided, if any).

(e) The Destination is compared against the routing-prefixes configured for each interface, or learned from Prefix-Information and Known-Identifier extensions. There is a Contiguous-bit which indicates whether the routing-prefix is confined to a single link [D-Form].

If the Destination exactly matches one of the routing-prefixes, and the Contiguous-bit is set, then the Destination is assumed to be on that specific link. For that interface, the Destination is located as described in "Sending General Solicitations".

(f) If the Destination exactly matches one of the routing-prefixes, but the Contiguous-bit is not set, then the datagram is sent directly to the indicated router (using the Media-Access extension provided, if any).

(g) If the Destination does not match any routing-prefixes, then the datagram is sent to a single preferred router, as described

in the "Router Selection" section which follows.

For a node with multiple interfaces, when one or more Router Advertisements have been heard on some interfaces, but no Router Advertisements have been heard on other interfaces, the datagram is duplicated as described above. It is sent to the most preferred router, and also to all those interfaces without routers for which a peer entity is unknown. This allows a node to continue operation in the presence of private or partitioned links.

Every host MUST operate correctly in a minimal environment. For example, if the host insists on finding at least one router to initialize, the host will be unable to operate on an isolated link.

3.4. Router Selection

The router is chosen based on the IPv6 Source. To decide which router to send a datagram, the following procedure is used:

(a) routing-prefixes are learned from the Prefix-Information extension of Router Advertisements. The Prefix_Size is the number of valid bits in the routing-prefix.

(b) The Source is compared to the list of routing-prefixes in the Router List.

(c) If a routing-prefix exactly matches the Source prefix extracted by the same Prefix_Size, then that router is one of the preferred routers for that Source. The node selects the highest preference value among those matching routers.

(d) If there are no matching routing-prefixes, or the Source is the Unknown Address, then there is no preferred router for the Source. The node selects the highest preference value among all routers found on all interfaces.

(e) If that router is not the best next-hop to the Destination, that router will forward the datagram to the best next-hop, and return a Local Redirect message to the sending node. See "Sending Local Redirects".

(f) When the sending node receives a Local Redirect, it updates the next-hop in the appropriate Hop Cache entry, so that later datagrams to the same Destination will go directly to the best next-hop. See "Processing Local Redirects".

When the Destination is determined to be accessible through a router, a separate entry is created in the Hop Cache for that Destination, and the cache entry for the router is used to send the datagram. The Router List entry might be duplicated in the Hop Cache, or a system of pointers could be used. In any case, the Hop Cache entry for the Destination MUST have the same LifeTime as the cache entry for the router.

Once a Hop Cache entry has been added for a Destination, it is not affected by future Router Advertisements from new routers, or changes in Prefix-Information extensions. Only a Local Redirect changes the Hop Cache.

RATIONALE:
In a multiple router environment, this permits a saturated router to dynamically lower its preferences or reduce its number of advertised Prefix-Information extentions, in order to better share the load, without dumping its entire load on another router.

3.5. Static Routes

A static route is typically a particular preset mapping for a Destination IPv6 Unicast or Cluster Address. Static routes would be installed by administrators to override the normal automatic routing mechanism, and to handle exceptional situations.

However, any static routing information is a potential source of failure as configurations change or equipment fails.

Each such static route MUST be overridden by Local Redirects for the LifeTime indicated. Otherwise, every datagram sent will result in a repeated Redirect message.

3.6. Dead Node Detection

Routers periodically advertise their availability. The advertisement frequency is always greater than the LifeTime of the advertisement. When the LifeTime of a Router Advertisement expires, the router is presumed dead, and that Router List entry is immediately removed. All dependent Hop Cache Destinations are also removed.

RATIONALE:
The exact constraints on the timeliness of "black hole" detection may vary somewhat depending on the nature of the node's mission,

but a node generally needs to detect a failed first-hop router quickly enough that transport-layer connections will not break before an alternate router can be selected.

Hosts do not periodically advertise. Therefore, when the LifeTime of a General Advertisement expires, it is retained for an implementation dependent period of time. This retention time MAY be a reasonably large value, to avoid excessive multicast probes for that Destination when a node only occasionally communicates with a given peer node. The retention time SHOULD NOT be arbitrarily large or infinite, to avoid losses and delays after an extended idle period when the link address of the Destination has changed.

NOTE:
This does not preclude purging of a cache entry prior to the expiration of its LifeTime or retention time, such as when the implementation has insufficient storage.

When a datagram is to be sent, but the Hop Cache entry has expired and is retained, the datagram is sent using the last known link address. It is immediately followed by a General Solicitation, which is unicast to the Destination. The Hop Cache LifeTime is updated as described in "Sending General Solicitations".

When an entry is purged, the routing availability of the Destination MUST be redetermined as if no prior entry had existed.

Negative "advice" from other layers, such as excessive retransmissions by a transport-layer protocol, or a down indication from a link-layer interface, SHOULD be used to purge a cache entry.

Positive "advice" from other layers, such as returning acknowledgements from a transport-layer protocol, MUST NOT extend the LifeTime of a cache entry.

Promiscuous observation of link-layer or internetwork-layer Source fields MUST NOT extend the LifeTime of a cache entry.

ICMP Echo "pings" by the internetowrk-layer MUST NOT be used to actively check a cache entry.

RATIONALE:
If the cache has timed out, the node does not re-solicit until it has a datagram to send.

Queuing is avoided when any former link address is known.

Passing advice from other layers of the protocol stack complicates

the interfaces between the layers, but it is the preferred approach.

The detection mechanism must not cause unacceptable load on the node, on congested links, or on first-hop router(s).

Using other layer information to shorten, but not to lengthen, the cache LifeTime ensures that failed nodes are found quickly. Assuming that the configured LifeTime results in a light load on the network, then there is not much to be gained by lengthening the period.

Packets arriving from a particular link-layer address are evidence that the system at this address is alive. However, turning this information into advice requires mapping the link-layer address into an internetwork-layer address, and then checking that address against the entries in the Hop Cache. This is probably prohibitively inefficient.

Positive advice that is given for every datagram received could cause unacceptable overhead in the implementation.

Ping scales poorly.

4. Processing Datagrams

For incoming datagrams, the internetwork-layer:

(1) verifies that the datagram is correctly formatted for the IP version.

(2) verifies that it is destined to the local host.

(3) processes subsequent headers and options.

(4) reassembles the datagram as necessary.

(5) passes the final payload to the appropriate transport-layer protocol module.

4.1. Address List

Each interface requires at least one IPv6 Address.

Each IPv6 Address is bound to at most one interface.

Each interface contains (at least) the following configurable variables:

Address

The IPv6 Unicast Address which is presently in use for the interface.

Default: None

Prefix_Size

Each Address entry bound to a link interface has an associated Prefix_Size. The value ranges from 0 to 127, and indicates the number of bits in the Address which define the routing-prefix for the link. |

When the value is not zero, the Address may be used to discern routing-prefix mapping. |

If all associated Prefix_Size values are zero, then prefix routing is not in use on that link.

Default: 0

LifeTime

The value for the time that the Address is associated with an interface.

Default: infinity

The routing-prefix(es) for a host interface SHOULD NOT be configured manually. |

The routing-prefix(es) for a router interface SHOULD be configured manually, until such time in the future that an automatic algorithm is developed. |

4.2. Details

An incoming datagram is destined for the node when the IPv6 Destination is:

(1) (one of) the IPv6 Unicast Address(es) on any interface of the node.

A host MUST NOT discard an incoming datagram whose Destination does not correspond to the logical interface through which it is received.

(2) an IPv6 Cluster Address which corresponds to the incoming interface, and which matches the Prefix_Size. |

(3) an IPv6 Multicast group of which the host is a member on the incoming interface.

A host MUST silently discard any IPv6 datagram which is not destined for itself.

A router MUST silently discard any IPv6 unicast datagram which is not destined for itself, and that has arrived with a link-layer broadcast or multicast indication. Other unicast, cluster and multicast routing details are beyond the scope of this document.

All nodes MUST silently discard any IPv6 datagram containing an invalid IPv6 Source, such as an IPv6 Cluster or Multicast Address. This validation could be done in either the internetwork-layer, or by each protocol in the transport-layer.

RATIONALE:

A mis-addressed datagram might be caused by a link-layer broadcast of a unicast datagram, or by any node that is confused or mis-configured.

All nodes are required to check for a link-layer broadcast or multicast, as well as an internetwork-layer address. This is necessary to prevent propagation of mis-addressed datagrams, which can result in broadcast storms.

An architectural goal for hosts is to allow addresses to be featureless numbers, avoiding algorithms that required a knowledge of the format. Otherwise, any future change in the format or interpretation of addresses will require host software changes.

However, validation of IPv6 Cluster and Multicast Addresses violates this goal. This is mitigated by the need to explicitly learn or join these groups.

5. Sending General Solicitations

Every IPv6 node MUST implement General Solicitations.

The General Solicitation is used by any node to determine both the reachability and the link-layer information of a neighboring node. |

5.1. Configuration *

A node SHOULD allow the following variables to be configured by system management. Default values are specified which make it unnecessary to re-configure these variables in most cases.

For each interface:

General_Solicitation_Interval

The value to be placed in the Lifetime field of the Hop Cache entry for General Solicitations sent from the interface. MUST NOT be less than 1 second, and SHOULD NOT be greater than 10 seconds. |

Default: 3 seconds

5.2. Details

A node is required to transmit a single General Solicitation, at the times specified in "Next Hop Decision" and "Dead Node Detection". |

Whenever a solicitation is sent, a Hop Cache entry is added or updated with a LifeTime of General_Solicitation_Interval. No further solicitations are sent until this Hop Cache entry expires. | |

RATIONALE:
This mechanism prevents flooding (repeating a solicitation at a high rate).

The General_Solicitation_Interval is chosen to allow sufficient round trip time for low bandwidth or congested links, and response time for heavily loaded nodes.

A LifeTime this short could create noticeable overhead traffic on a link with large number of nodes. Therefore, it may be necessary to configure busy routers or active servers with a longer

General_Solicitation_Interval.

The following method is used to send the solicitation:

(a)If the interface and link address are known from an expired Hop Cache entry, the General Solicitation is sent using the retained link address.

(b) If the interface has no broadcast capability (a point-to-point link), and the peer entity is unknown (no advertisements received), the General Solicitation is sent on that interface. No link address is needed.

(c) If a virtual interface has no broadcast capability (a Frame-Relay or X.25 link), the General Solicitation is duplicated on each virtual circuit for which there is no known peer entity, as if they were each a separate point-to-point interface on a node with multiple physical interfaces. The link address used is determined by the virtual circuit setup.

(d) If an interface has no multicast capability, the General Solicitation is sent as a link-layer broadcast. The IPv6 Destination is unchanged.

(e) For an interface with multicast capability, the General Solicitation is sent as a link-layer multicast. The IPv6 Destination is used to calculate the appropriate multicast.

The solicitation is not delayed.

Upon receiving a valid advertisement (of any kind) from the target Destination, the node MUST NOT send any solicitation on that interface (even if no solicitation has been sent yet) until the advertisement LifeTime expires.

RATIONALE:
This serves to alleviate congestion when many nodes start up on a link at the same time, such as might happen after recovery from a power failure, or the periodic Hop Cache refresh of a large number of clients sharing a server.

When the link address is unknown, the original datagram SHOULD be held (rather than discarded) until a valid advertisement is received. When additional datagrams for the same Destination are received, the most recent are saved, and earlier datagrams MAY be discarded.

When the Hop Cache entry expires while waiting for an advertisement, any held datagrams MUST be discarded, and the entry is purged.

RATIONALE:

Failure to follow this recommendation causes the first packet of every exchange to be lost. Although transport-layer protocols can generally cope with packet loss by retransmission, packet loss does impact performance.

For example, loss of a TCP open request causes the initial round-trip time estimate to be inflated. UDP applications, such as the Domain Name System, are more seriously affected. |

Purging the link address after failure to receive a response ensures that changes in link address are detected. |

Repeating solicitations after failure to receive a response, without waiting for renewed transport-layer stimulus, can cause congestive collapse. |

6. Processing General Solicitations

Every IPv6 node MUST process General Solicitations.

All IPv6 nodes MUST accept the calculated Solicited-Nodes IPv6 Multicast Address for every address bound to every interface.

This is calculated by starting with the exclusive-or of each byte of the target IPv6 Unicast Address, then adding the result to the base Solicited-Nodes multicast (FFxx::0700).

For example, to calculate the destination value for target A::B:C, the exclusive-or is D. The calculated destination would be FFxx::070D.

On receipt of a valid General Solicitation, the target node sends a General Advertisement, using the extension information provided.

6.1. Validity

All nodes MUST silently discard any received General Solicitation messages that do not satisfy the following validity checks:

- IPv6 Version is correct.
- IPv6 Source is a Unicast Address, is not the Unspecified Address, and is not a Cluster or Multicast Address.
- When an Authentication Header is present, it is correct.
- ICMP Checksum is correct.
- ICMP length (derived from the payload length) is 8 or more octets.
- The Known-Identifier extension indicates one of the IPv6 Unicast Addresses which is bound to any interface of the node.
- For interfaces which are not point-to-point links, the Media-Access extension is present.

6.2. Details

The solicitation has no LifeTime. The extension information is used only for returning the advertisement, and then discarded.

No new Hop Cache entry is added, and any existing entry is not updated.

RATIONALE:
At the time of solicitation, the extension information might not be complete. For example, the initial solicitation will not contain the Node-Heard for the target, and the target will not be assured that the path to the sender is complete.

This also helps stagger solicitations.

To process a General Solicitation, the node scans the list of extensions in it.

6.2.1. Media-Access

If a Media-Access extension is present, the information MAY be used to return the General Advertisement directly to the solicitor. The Media-Access extension MAY appear anywhere in the list of extensions, but is most likely at the beginning or end.

6.2.2. Node-Heard

The absence of the Node-Heard extension serves as an indication that the solicitor has not yet heard any Router Advertisement. The General Advertisement MUST be sent directly to the solicitor.

If a Node-Heard extension is present which indicates that the solicitor has previously heard the node, it is confirmation of contact in those cases where the routing-prefix is not entirely confined to the link. The General Advertisement MAY be sent directly to the solicitor.

7. Sending General Advertisements

Every IPv6 node MUST implement General Advertisements.

A General Advertisement is sent in response to a General Solicitation, or to expire a previous Advertisement.

7.1. Constants

GENERAL_ADVERTISEMENT_COUNT 4 entries

7.2. Configuration

A node SHOULD allow the following variables to be configured by system management. Default values are specified which make it unnecessary to re-configure these variables in most cases.

For each interface:

Advertisement_LifeTime

The value to be placed in the Lifetime field of General Advertisements sent from the interface. The value MAY be reduced on a case by case basis for demand critical applications operating in a low delay environment.

Default: 600 seconds

7.3. Details

The IPv6 Source specified in the solicitation is used as the IPv6 Destination in the advertisement, except under the following conditions:

(a) When the number of current Hop Cache entries (exclusive of static routes and router list entries) is GENERAL_ADVERTISEMENT_COUNT or more.

(b) When one or more Router Advertisements have been heard, and the Source does not match any routing-prefixes configured for an interface, or learned from Prefix-Information extensions.

(c) When the Source exactly matches one of the routing-prefixes, but the Contiguous-bit is not set. |

In these cases, the advertisement is sent to the All-Nodes IPv6 Multicast Address (FF02::1). The scope is intra-link.

RATIONALE:

The decision to multicast an advertisement to all nodes, instead of repeating a unicast to each successive soliciting node, is a balance between disturbing a large number of nodes at the internetwork-layer against a greater amount of traffic.

Assuming that each node communicates with every neighbor, the discovery traffic increases at the rate of 2n*n nodes. When most nodes communicate only with a server, a multicast advertisement reduces the traffic to 2n.

The multicast advertisements could be particularly disruptive when they interrupt the sleep mode of a battery powered device. However, the device might already have been disrupted by the solicitation when the link has broadcast and not multicast capability. Also, it is precisely those devices which are most likely to be deployed on bandwidth-limited links, where a reduction of traffic is most important.

In addition, roaming nodes which experience multipath and half-link conditions use the multicast advertisement to learn whether a direct contact is possible.

8. Processing General Advertisements

Every IPv6 node MUST process General Advertisements.

All IPv6 nodes MUST accept the All-Nodes IPv6 Multicast Address (FFxx::1) on every interface.

On receipt of a valid General Advertisement, all nodes which have a Hop Cache entry for the Source update the cache entry with the current LifeTime and link address, and any other pertinent field values implemented.

8.1. Validity

All nodes MUST silently discard any received General Advertisement messages that do not satisfy the following validity checks:

- IPv6 Version is correct.
- IPv6 Source is a Unicast Address, is not the Unspecified Address, and is not a Cluster or Multicast Address.
- When an Authentication Header is present, it is correct.
- ICMP Checksum is correct.
- ICMP length (derived from the payload length) is 16 or more octets.
- For interfaces which are not point-to-point links, the Media-Access extension is present.

8.2. Details

To process a General Advertisement, the node scans the list of extensions in it.

8.2.1. Media-Access

If a Media-Access extension is present, the Hop Cache is updated with the information. The Media-Access extension MAY appear anywhere in the list of extensions, but is most likely at the beginning or end.

When an unsolicited advertisement is received, this extension MUST NOT be compared with the current contents of the Hop Cache, since the advertisement might have been sent from another interface.

8.2.2. Known-Identifier

The Known-Identifier extension is used to indicate IPv6 Addresses bound to other interfaces of the node, or other IPv6 addresses on the same interface which are not subsumed by the same routing-prefix.

Each Known-Identifier MAY be used to add or update another Hop Cache entry.

8.2.3. Node-Heard

If a Node-Heard extension is present which indicates that the advertiser has previously heard the node, it is confirmation of contact in those cases where the routing-prefix is not entirely confined to the link.

If the Quality specified is not zero, but is less than the Quality for some other router Node-Heard extension, the Hop Cache entry MAY be updated to point to that router instead. A Routing Header can be used to direct datagrams along the more reliable path.

9. Sending Router Solicitations

Every IPv6 node MUST implement Router Solicitations.

When any node initializes, it MUST send the Router Solicitation to prompt the advertisement of neighboring routers.

If (and only if) no advertisements from neighboring routers are forthcoming, the node MAY retransmit the Router Solicitation a small number of times, but then MUST desist from sending more solicitations.

Any routers that subsequently start up, or that were not discovered because of packet loss or temporary link partitioning, are eventually discovered by reception of their periodic (unsolicited) advertisements. Links that suffer high packet loss rates or frequent partitioning are accommodated by increasing the rate of Router Advertisements, rather than increasing the number of solicitations that nodes are permitted to send.

9.1. Constants

MAX_SOLICITATIONS	3 transmissions
MAX_SOLICITATION_DELAY	2 seconds

9.2. Configuration

A node SHOULD allow the following variables to be configured by system management. Default values are specified which make it unnecessary to re-configure these variables in most cases.

For each interface:

Router_Solicitation_Interval

The value to be used for repeated Router Solicitations sent from the interface. MUST NOT be less than 2 * MAX_SOLICITATION_DELAY, and SHOULD NOT be greater than 10 seconds.

Default: 6 seconds

9.3. Details

A node is required to transmit up to MAX_SOLICITATIONS messages from any of its interfaces 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.

- A router has its forwarding capability for that interface turned off by system management.

If a node chooses to send a solicitation after one of the above events, it MUST delay transmission for a random amount of time between 0 and MAX_SOLICITATION_DELAY.

It is recommended that nodes include some unique value (such as one of their interface or link-layer identifiers or addresses) in the seed used to initialize their pseudo-random number generators. Although the randomization range is specified in units of seconds, the actual randomly-chosen value SHOULD NOT be in units of whole seconds, but rather in units of the highest available timer resolution.

The small number of retransmissions of a solicitation, which are permitted if no advertisement is received, should be sent at intervals of Router_Solicitation_Interval without further randomization.

Upon receiving a valid Router Advertisement subsequent to one of the above events, the node MUST NOT send any solicitation on that interface (even if no solicitation has been sent yet) until the next time one of the above events occurs.

RATIONALE:
This serves to alleviate congestion when many nodes start up on a link at the same time, such as might happen after recovery from a power failure.

10. Processing Router Solicitations

Every IPv6 router MUST process Router Solicitations.

All IPv6 routers MUST accept the All-Routers IPv6 Multicast Address (FFxx::2) on every interface for which forwarding is enabled.

On receipt of a valid Router Solicitation, the target router sends a Router Advertisement.

If the IPv6 Source does not match one of the router's own IPv6 Cluster Addresses on the arrival interface, by matching the associated routing-prefix, the sender is considered a Mobile Node. The location of every reachable Mobile Node is maintained separately within the router.

10.1. Validity

A non-router MUST silently discard any received Router Solicitation messages.

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

- IPv6 Version is correct.
- IPv6 Source is the Unspecified Address, or an IPv6 Unicast Address, but is not a Cluster or Multicast Address.
- When an Authentication Header is present, it is correct.
- ICMP Checksum is correct.
- ICMP length (derived from the payload length) is 8 or more octets.
- For interfaces which are not point-to-point links, the Media-Access extension is present.

10.2. Details

To process a Router Solicitation, the node scans the list of extensions in it.

10.2.1. Media-Access

If a Media-Access extension is present, the information MAY be used to return the Router Advertisement directly to the solicitor. The Media-Access extension MAY appear anywhere in the list of extensions, but is most likely at the beginning or end.

10.2.2. Node-Heard

The absence of the Node-Heard extension serves as an indication that the solicitor has not yet heard any Router Advertisement. The Router Advertisement MUST be sent promptly, at the accelerated initial rate.

If a Node-Heard extension is present which indicates that the solicitor has previously heard the node, it is confirmation of contact in those cases where the routing-prefix is not entirely confined to the link.

11. Sending Router Advertisements

Every IPv6 router MUST implement Router Advertisements.

A Router Advertisement is sent periodically, and also in response to a Router Solicitation.

11.1. Constants

MAX_INITIAL_ADVERTISEMENTS	3 transmissions
MAX_INITIAL_ADVERT_INTERVAL	10 seconds
MAX_ADVERTISEMENT_DELAY	1 second

11.2. Configuration

A router MUST allow the following variables to be configured by system management. Default values are specified which make it unnecessary to re-configure these variables in most cases.

For each interface:

Minimum_Advertisement_Interval

The minimum time allowed between sending unsolicited Router Advertisements from the interface. MUST NOT be less than MAX_ADVERTISEMENT_DELAY.

Default: 30 seconds

Maximum_Advertisement_Interval

The maximum time allowed between sending Router Advertisements from the interface. MUST NOT be less than Minimum_Advertisement_Interval.

Default: 1.50 * Minimum_Advertisement_Interval

Advertisement_LifeTime

The value to be placed in the Lifetime field of Router Advertisements sent from the interface. MUST NOT be less than Maximum_Advertisement_Interval, and SHOULD NOT be greater than

600 seconds (10 minutes).

Default: 3 * Maximum_Advertisement_Interval

For each of the IPv6 Unicast or Cluster Addresses of each interface:

Advertise

A flag indicating whether or not the IPv6 Address is to be advertised.

Default: TRUE

Preference

The preferability of the interface as a default router choice, relative to other router interfaces serving the same routing-prefix on the same link.

Values are in the range 0 to 255. Higher values mean more preferable. The minimum value zero is reserved to indicate that the IPv6 Address, even though it may be advertised, is not to be used by neighboring hosts as a default Router Address. The maximum value 255 is reserved to indicate that the preference was locally configured, and not learned through advertisments.

Default: 128

It is useful to configure an IPv6 Address with a preference level of zero (rather than simply setting its Advertise flag to FALSE) when advertisements are being used for "black hole" detection. In particular, a router that is to be used to reach only specific destinations could advertise a preference level of zero (so that neighboring hosts will not use it as a default router for reaching arbitrary destinations) and a non-zero lifetime (so that neighboring hosts that have been redirected or configured to use it can detect its failure by timing out the reception of its advertisements).

DISCUSSION:

It has been suggested that, when the preference level of an IPv6 Address has not been explicitly configured, a router could set it according to the metric of the router's "default route" (if it has one), rather than defaulting as suggested above. Thus, a router with a better metric for its default route would advertise a higher preference level for its IPv6 Address.

(Note that routing metrics that are encoded such that "lower is better" would have to be inverted before being used as preference levels in Router Advertisement messages.) Such a strategy might reduce the amount of redirect traffic on some links by making it more likely that the host's first choice for reaching an arbitrary destination is also the best choice.

On the other hand, redirect traffic is rarely a significant load on a link, and there are some cases where such a strategy would result in more redirect traffic (on links from which the most frequently chosen destinations are best reached via routers other than the one with the best default route). Also, since the routing algorithms learn of neighboring routers from the advertisements, and the default routes are learned from the routing algorithms, the calculated preference may be unstable from time to time. This document makes no recommendation concerning this issue, and implementors are free to try such a strategy, as long as they also support static configuration of preference levels as specified above.

11.3. Details

The term "advertising interface" refers to any functioning and enabled interface that has at least one IPv6 Address whose configured Advertise flag is TRUE.

From each advertising interface, the router MUST transmit Router Advertisements.

The 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. This is done by maintaining a separate transmission interval timer for each advertising interface. Each time an advertisement is sent from an interface, that interface's timer is reset to a uniformly-distributed random value between the configured Minimum_Advertisement_Interval and Maximum_Advertisement_Interval. Expiration of the timer causes the next advertisement to be sent, and a new random value to be chosen.

For the first few advertisements sent from an interface (up to MAX_INITIAL_ADVERTISEMENTS), if the randomly chosen interval is greater than MAX_INITIAL_ADVERT_INTERVAL, the timer should be set to MAX_INITIAL_ADVERT_INTERVAL instead. Using this 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.

An interface may become an advertising interface at times other than system startup, as a result of recovery from an interface failure or through actions of system management such as:

- enabling the interface, if it had been administratively disabled and it has one or more IPv6 Addresses whose Advertise flag is TRUE,

- enabling IPv6 forwarding capability (changing the node from a host to a router), when the interface has one or more IPv6 Addresses whose Advertise flag is TRUE,

- setting the Advertise flag of one or more of the interface's IPv6 Addresses to TRUE (or adding a new IPv6 Address with a TRUE Advertise flag), when previously the interface had no IPv6 Address whose Advertise flag was TRUE.

In such cases, the router MUST commence transmission of periodic advertisements on the new advertising interface, limiting the first few advertisements to intervals no greater than MAX_INITIAL_ADVERT_INTERVAL. In the case of a host becoming a router, the node MUST also accept the All-Routers IPv6 Multicast Address on all interfaces on which the router supports multicast (whether or not they are advertising interfaces).

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

- shutting down the node,

- administratively disabling the interface,

- disabling IPv6 forwarding capability (changing the node from a router to a host),

- setting the Advertise flags of all of the interface's IPv6 Addresses to FALSE.

In such cases, the router MUST transmit a final multicast advertisement on the interface, identical to its previous transmission, but with a Lifetime of zero. In the case of a router becoming a host, the node MUST also drop the All-Routers IPv6 Multicast Address on all interfaces on which the router supports multicast (whether or not they had been advertising interfaces).

When the Advertise flag of one or more of an interface's IPv6

Addresses are set to FALSE by system management, but there remain other IPv6 Addresses on that interface whose Advertise flags are TRUE, the router SHOULD send a single multicast advertisement containing only those IPv6 Addresses whose Advertise flags were set to FALSE, with a Lifetime of zero.

In addition to the periodic unsolicited advertisements, a router MUST send advertisements in response to valid Router Advertisements or Router Solicitations received on any of its advertising interfaces. If the received advertisement or solicitation does not contain any Node-Heard extension, and the time since the previous advertisement is greater than MAX_INITIAL_ADVERT_INTERVAL, the router MUST multicast an advertisement from that interface.

Whenever these response advertisements are sent, the node MUST delay transmission for a random amount of time between 0 and MAX_ADVERTISEMENT_DELAY, in order to prevent synchronization with other responding routers, and to allow multiple closely-spaced solicitations to be answered with a single advertisement. The interface's interval timer is reset to a new random value, as with unsolicited advertisements.

It is recommended that routers include some unique value (such as one of their interface or link-layer addresses) in the seed used to initialize their pseudo-random number generators. Although the randomization range is configured in units of seconds, the actual randomly-chosen values SHOULD NOT be in units of whole seconds, but rather in units of the highest available timer resolution.

12. Processing Router Advertisements

Every IPv6 node MUST process Router Advertisements.

All IPv6 nodes MUST accept the All-Nodes IPv6 Multicast Address (FFxx::1) on every interface.

Each router saves the information contained in the advertisements, in order to respond to future requests. Any other action on receipt of such messages by a router (for example, as part of a "peer discovery" process) is beyond the scope of this document.

Each host saves the information contained in the advertisements, in order to determine the next-hop when sending datagrams. Hop determination is elaborated in "Sending Datagrams".

12.1. Validity

All nodes MUST silently discard any received Router Advertisement messages that do not satisfy the following validity checks:

- IPv6 Version is correct.
- IPv6 Source is a Unicast Address, is not the Unspecified Address, and is not a Cluster or Multicast Address.
- When an Authentication Header is present, it is correct. |
- ICMP Checksum is correct.
- ICMP length (derived from the payload length) is 16 or more octets.
- At least one Prefix-Information extension is present. |
- For interfaces which are not point-to-point links, the Media-Access extension is present.

12.2. Router List

Host Requirements -- Communication Layers [1], Section 3.3.1.6, specifies that each host (must) support a configurable list of default routers. The purpose of the Router Advertisement messages is to eliminate the need to configure that list.

Each entry in the list contains (at least) the following configurable variables:

Router_Identifier

An IPv6 Unicast Address of a default router.

Default: (none)

Prefix_Size

Each router entry has an associated Prefix_Size. The value ranges from 0 to 127, and indicates the number of bits in the IPv6 Address which define the routing-prefix for the link. A value of zero indicates an end-point IPv6 Address. When the value is not zero, the IPv6 Address may be used to discern routing-prefix mapping.

If all associated Prefix_Size values are zero, then prefix routing is not in use on that link.

Default: 0

Preference

The preferability of the Router_Identifier as a default router choice, relative to other router interfaces serving the same routing-prefix on the same link. Host Requirements does not specify how this value is to be encoded.

The values used here are defined as in Router Advertisements. Values are in the range 0 to 255. Higher values mean more preferable. The minimum value zero is reserved to indicate that the IPv6 Address, even though it may be advertised, is not to be used by neighboring hosts as a default Router Address. The maximum value 255 is reserved to indicate that the preference was locally configured, and not learned through advertisments.

Default: 255

Default routers and preference levels SHOULD NOT be configured manually. On links for which router discovery is administratively disabled, it MAY continue to be necessary to configure the default Router List in each host.

NOTES: Any router IPv4 Address acquired from the "Gateway" subfield of the vendor extensions field of a BOOTP packet [RFC-BOOTP?11] are considered to be configured; they are assigned the

default preference level of 255, and they do not have an associated LifeTime.

Any IPv4 Address found in the "giaddr" field of a BOOTP packet [RFC-BOOTP?3] identifies a BOOTP forwarder which is not necessarily a router; an entry SHOULD NOT be installed in the default Router List.

12.3. Details

To process a Router Advertisement, the node scans the list of extensions in it.

12.3.1. Media-Access

If a Media-Access extension is present, the Router List is updated with the information. The Media-Access extension MAY appear anywhere in the list of extensions, but is most likely at the beginning or end.

12.3.2. Change-Identifier

This extension gives advance indication that an address or prefix will no longer be routable. Applications SHOULD cease to accept new connections with the old value. Existing connections SHOULD issue a Remote Redirect.

Change-Identifier extensions MUST preceed Prefix-Information extensions. |

- If the Prefix_Size is zero, the IPv6 Address indicates the change of a single node, without affecting other nodes on that link.

- If the Prefix_Size is not zero, the IPv6 Address indicates the change of routing-prefix for all nodes on that link. |

- The IPv6 Address and Prefix_Size are compared against any IPv6 Addresses defined for the node. If there is a match, a Remote Redirect MAY be sent to correspondents to inform them of the change.

The node MUST continue to accept datagrams destined for the old IPv6

Address(es), until such time as all stimulus for maintaining the entry has expired. This implies that the node will maintain a LifeTime for most sources of IPv6 Addresses, such as DNS records and dynamic configuration.

12.3.3. Prefix-Information

Prefix-Information extensions MUST preceed Known-Identifier extensions.

- If the Prefix_Size is not zero, the IPv6 Address and Prefix_Size are compared against any IPv6 Addresses defined for the node. If there is a match, the IPv6 Address is associated with the interface on which the message was received, and the Prefix_Size is set to the advertised Prefix_Size.

- If the IPv6 Address is not already present in the Router List, a new entry is added to the list, containing the IPv6 Address along with its accompanying preference level, and the Lifetime value from the advertisement.

- If the IPv6 Address is already present in the Router List as a result of a previously-received advertisement, its preference level is updated and its LifeTime is reset to the value in the newly-received advertisement.

- If the IPv6 Address is already present in the Router List as a result of static configuration, no change is made to its preference level. There is no LifeTime associated with a configured IPv6 Address. To limit the storage needed for the default Router List, the host MAY choose not to store all of the router IPv6 Addresses discovered via advertisements. The host SHOULD discard those IPv6 Addresses with lower preference levels in favor of those with higher levels.

It is desirable to retain more than one default router in the list; if the current choice of default router is discovered to be down, the host may immediately choose another default router without having to wait for the next advertisement to arrive.

Any router IPv6 Address advertised with a preference level of zero is not to be used by the host as default router IPv6 Address. Such an IPv6 Address may be omitted from the default Router List, unless its LifeTime is being used as a "black-hole" detection mechanism.

12.3.4. Known-Identifier

The Known-Identifier extension is used to indicate IPv6 Addresses bound to other interfaces of the router, or other IPv6 Addresses on the same interface which are not subsumed by the same routing-prefix.

- If the IPv6 Address is not already present in the Router List, a new entry MAY be added, containing the IPv6 Address, the preference level set to zero, and the Lifetime value from the advertisement.

- If the IPv6 Address is already present in the Router List as a result of a previously-received advertisement, and its preference level is zero, its LifeTime is reset to the value in the newly-received advertisement.

- If the IPv6 Address is already present in the Router List as a result of static configuration, no change is made to its preference level. There is no LifeTime associated with a configured IPv6 Address.

  To limit the storage needed for the default Router List, the host MAY choose not to store all of the other IPv6 Addresses discovered via advertisements. The most preferred router is used for unknown Destinations, and it will send a redirect when appropriate.

12.3.5. Node-Heard

As in a General or Router Solicitation, the absence of the Node-Heard extension serves as an indication that the router has not yet heard any other Router Advertisement.

If the Quality specified is not zero, but is less than the Quality for some other router Node-Heard extension, the Hop Cache entry MAY be updated to point to that router instead. A Routing Header can be used to direct datagrams along the more reliable path.

13. Sending Local Redirects

Every IPv6 router MUST implement Local Redirect.

A router sends a Local Redirect when it receives datagrams for which that router is not the best next-hop to the Destination. The router will forward the datagram to the best next-hop, and return a Local Redirect message to the sending node.

A host SHOULD NOT send a Local Redirect.

14. Processing Local Redirects

Every IPv6 host MUST process Local Redirects. |

On receipt of a valid Local Redirect, a host MUST update its Hop Cache. |

Every IPv6 router which is participating in a routing protocol MUST ignore Local Redirects. |

14.1. Validity

All nodes MUST silently discard any received Local Redirect messages that do not satisfy the following validity checks:

- IPv6 Version is correct.

- IPv6 Source is a Unicast Address, is not the Unspecified Address, is not a Cluster or Multicast Address, and is the current next hop router for the target specified in the Known-Identifier extension(s). |

- When an Authentication Header is present, it is correct. |

- ICMP Checksum is correct.

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

- The Known-Identifier extension indicates one of the IPv6 Unicast Addresses in the Hop Cache. |

- For interfaces which are not point-to-point links, the Media-

Access extension is present.

14.2. Details

Since the routing-prefixes bound to an interface are not required to be relevant for all Destinations, the next hop specified is always presumed to be accessible via the same interface through which the Redirect arrived. The Redirect MUST NOT be discarded simply because it arrives on an interface which has no matching advertised prefix.

RATIONALE:
A Mobile Node will likely not have a prefix which matches any router advertised prefixes. When a local host (such as a printer or DNS) responds to a message from the Mobile Node, it will initially send to its preferred router. That router will send a Redirect to the Mobile Node.

When a Redirect is received with a non-zero Prefix_Size, it is treated as if it has a zero Prefix_Size. That is, the cache entry for the Destination (only) would be updated (or created when an entry for that Destination did not exist), with a Prefix_Size of zero.

RATIONALE:
This recommendation is to protect against routers that erroneously send Redirects for an entire routing prefix.

15. Sending Remote Redirects

Every IPv6 node MUST implement Remote Redirects.

A node sends a Remote Redirect when it receives a Router Advertisement containing Change-Identifier extensions. The Hop Cache is examined for Destinations accessed through that router. Those remote nodes are sent the Remote Redirect with an indication of a Care-Of-Address to use in order to reach the expiring identification of the node.

A Mobile Node MAY also send a Remote Redirect when it receives a datagram which does not have a Routing Header containing its current Care-Of-Address(es). See [Mobility] for details. |

The Remote Redirect is only sent to those remote nodes with which the node maintains a Security Association.

16. Processing Remote Redirects

Every IPv6 node MUST process Remote Redirects.

On receipt of a valid Remote Redirect, the node uses a Routing Header to reach the sender.

16.1. Validity

All nodes MUST silently discard any received Remote Redirect messages that do not satisfy the following validity checks:

- IPv6 Version is correct.

- IPv6 Source is a Unicast Address, is not the Unspecified Address, is not a Cluster or Multicast Address, and indicates one of the IPv6 Unicast Addresses in the Hop Cache.

- An Authentication Header is present, and it is correct. |

- ICMP Checksum is correct.

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

16.2. Details

To process a Remote Redirect, the node scans the list of extensions in it.

16.2.1. Known-Identifier |

The Known-Identifier extension is used to indicate the IPv6 Unicast or Cluster Address which is used as a Care-Of-Address to reach the Source. |

A. Configuration Summary

A.1. Router Configuration

A router requires at least one IPv6 Address to be configured.

For each interface, a Prefix_Size is assigned to each IPv6 Address, unless automatic prefix discovery is in place.

> Note that this procedure minimizes the number of items to be configured, and possible configuration errors.

Optionally, other values MAY be altered from their defaults, such as preference and advertisement lifetime.

Optionally, routing protocols MAY require additional values to be configured, such as metric and priority. Such functions are beyond the scope of this document.

A.2. Host Configuration

Most hosts need no prior configuration.

A node attached to a multi-access link creates a local-use unicast address from the link address. |

A node attached to a point-to-point link (using the Point-to-Point Protocol [RFC-1661]) can be dynamically assigned either a global or local unicast address. |

Other nodes require configuration of an IPv6 Address, as described in "Address List".

B. Hop Cache Implementation

Each Hop Cache entry needs to include the following items:

(1) LifeTime
(2) Next-hop interface (when a node is multi-homed)
(3) Next-hop link address |
(4) Destination IPv6 Address
(5) Destination Prefix_Size
(6) Source IPv6 Address
(7) Flow Label
(8) Path Maximum Transmission Unit
(9) Path Round Trip Time

Field (4) MAY be the full IPv6 Address of the Destination, or the Cluster which includes the Destination. This is determined by the routing-prefix size in (5). |

Field (7) SHOULD be included, as it is related to the Source in (6).

DISCUSSION:
Each Hop Cache entry defines the end-points of an internetwork path. Although the connecting path may change dynamically in an arbitrary way, the transmission characteristics of the path tend to remain approximately constant over a time period longer than a single typical host-host transport connection. Therefore, a Hop Cache entry is a natural place to cache data on the properties of the path.

Examples of such properties might be the maximum unfragmented datagram size, or the average round-trip delay measured by a transport protocol. This data will generally be both gathered and used by a higher layer protocol (that is, by TCP or by an application using UDP). Experiments are currently in progress on caching path properties in this manner.

There is no consensus on whether the Hop Cache should be keyed on destinations alone, or allow both Unicast and Cluster addresses. Those who favor the use of only node identifiers argue that:

(1) Redirect messages will generally result in entries keyed on nodes. The simplest and most general scheme would be to only use node identifiers.

(2) The internetwork layer may not always know the Prefix_Size for a remote link. |

(3) The use of only node identifiers may allow the Internet

architecture to be more easily extended in the future without any change to the hosts.

The opposing view is that allowing a mixture of destination nodes and routing-prefixes in the Hop Cache: |

(1) Saves memory space.

(2) Leads to a simpler data structure, easily combining the cache with the tables of default and static routes.

(3) Provides a more useful place to cache path properties.

The cache needs to be large enough to include entries for the maximum number of destinations that may be in use at one time.

Advertisement updates could indefinitely continue to refresh otherwise unused entries. A Hop Cache entry SHOULD also include control information used to choose an entry for replacement. For example, this might take the form of a "recently used" bit, a use count, or a last-used timestamp. It is also recommended that it include the time of last modification of the entry, for diagnostic purposes.

An implementation may wish to reduce the overhead of scanning the Hop Cache for every datagram to be transmitted. This may be accomplished with a hash table to speed the lookup, or by giving a connection-oriented transport protocol a "hint", or temporary handle on the appropriate cache entry, to be passed to the internetwork-layer with each subsequent datagram.

Although the Hop Cache, the Router List, and a table of static routes are described as conceptually distinct, in practice they may be combined into a single "routing table" data structure.

C. Proxy Advertisements

A router MAY proxy for the identifiers of other nodes, using the Known-Identifier extension. |

This SHOULD only be used when the router is translating to another internetwork protocol format.

D. Examples

D.1. Simple Solicitation

Assume host A has address 1 and host B has addresses 2 and 3. There is only one interface on A, and only one interface on B.

A is trying to talk to B, so it sends a General Solicitation to B, fills in portions of a Hop Cache entry, and waits for the General Advertisement from B.

The Known-Identifier extension in the solicitation is 2. |

B sends a General Advertisement with the Media-Access for its interface. It also puts 3 in an Known-Identifier extension. |

A MUST update its cache entry for 2.

A SHOULD also add another cache entry for 3, using the same link address. This is not required in a minimal memory implementation. |

D.2. Complex Solicitation

Assume that host B had a different interface for 3 on the same link.

If host A already had a Hop Cache entry for 3 (using the original link address), but the advertisement (above) contains a different link address for 3, A MUST add another cache entry for 3, pointing to 2. | |

This is required in order that the purpose of redundant interfaces on the same link be fulfilled, and 3 is accessible through 2 (and vice versa) when its interface fails.

The pairing of IPv6 address and link address can be considered a tuple consisting of {IPv6 address, interface, link address}. |

This allows annealing of partitioned links with no effort by hosts.

Security Considerations

There are a lot of Security issues which are not discussed in this memo.

Acknowledgements

The document was initially composed of quotations from the RFC-1122 "Requirements for Internet Hosts -- Communication Layers" (Robert Braden, Editor), and RFC-1256 "ICMP Router Discovery Messages" (Steve Deering, Editor), and "Requirements for IP Routers" (Almquist and Kastenholz, Editors).

Thanks also for suggestions and contributions from the Simple-IP Working Group.

The Dead Node detection method was clarified by Robert Elz.

Special thanks for implementation review by Ran Atkinson (Naval Research Laboratory), Alex Conta (Digital Equipment Corporation), Dan McDonald (Naval Research Laboratory), Fred Rabouw (Network Systems Netherlands), and Brad Stone (Hewlett-Packard).

Author's Address

Questions about this memo can also be directed to:

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

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

Table of Contents