          IPv6 Neighbor Discovery Optimizations for Wired and Wireless Networks
                 draft-chakrabarti-nordmark-6man-efficient-nd-07

Abstract

   IPv6 Neighbor Discovery (RFC 4861 going back to RFC 1970) was defined
   at a time when link-local multicast was as reliable and with the same
   network cost (send a packet on a yellow-coax Ethernet) as unicast and
   where the hosts were assumed to be always on and connected.

   Since 1996 we've seen a significant change with both an introduction
   of wireless networks and battery operated devices, which poses
   significant challenges for the old assumptions. We are also seeing
   datacenter networks where virtual machines are not always on and
   connected, and scaling of multicast can be challenging.

   This specification contains extensions to IPv6 Neighbor Discovery
   which remove most use of multicast and make sleeping hosts more
   efficient. The specification includes a default mixed mode where a
   link can have an arbitrary mix of hosts and/or routers - some
   implementing legacy Neighbor Discovery and some implementing the
   optimizations in this specification. The optimizations provide
   incremental benefits to hosts as soon as the first updated routers
   are deployed on a link.

time. It is inappropriate to use Internet-Drafts as reference
material or to cite them other than as "work in progress."

This Internet-Draft will expire on August 31, 2015.

Copyright Notice

Table of Contents

1. Introduction

   IPv6 Neighbor Discovery [RFC4861] was defined at a time when local
   area networks had different properties than today. A common link was
   the yellow-coax shared wire Ethernet, where a link-layer multicast
   and unicast worked the same - send the packet on the wire and the
   interested receivers will pick it up. Thus the network cost
   (ignoring any processing cost on the receivers that might not
   completely filter out Ethernet multicast addresses that they did not
   want) and the reliability of sending a link-layer unicast and
   multicast was the same. Furthermore, the hosts at the time was
   always on and connected. Powering on and off the workstation/PC
   hosts at the time was slow and disruptive process.

   Under the above assumptions it was quite efficient to maintain the
   shared state of the link such as the prefixes and their lifetimes
   using periodic multicast Router Advertisement messages. It was also
   efficient to use multicast Neighbor Solicitations for address
   resolution as a slight improvement over the broadcast use in ARP.
   And finally, checking for a potential duplicate IPv6 address using
   broadcast was efficient and believed to be likely to be robust.

   Since then we've seen a tremendous change with the wide-spread
   deployment of WiFi and other wireless network technologies. WiFi is
   a case in point in that it provides the same network service
   abstraction as Ethernet and is often bridged with Ethernets, meaning
   that the Neighbor Discovery software on hosts and routers might be
   unaware that WiFi is being used. Yet the performance and reliability
   of multicast is quite different than for unicast on WiFi (see for
   instance [I-D.vyncke-6man-mcast-not-efficient]). Similarly 3GPP and
   M2M networks and devices will benefit from a standard specification
   for optimized Neighbor discovery. Even wired networks have evolved
   substantially with modern switch fabrics using explicit packet
   replication logic to handle multicast packets.

   The assumptions about the reliability of a single multicast message
   for duplicate address detection has also shown to be not correct, due
   to a set of issues listed in [I-D.yourtchenko-6man-dad-issues].

   The hosts and usage patterns has undergone radical changes as well.
   Hosts go to sleep when not in use to save on battery power [RFC6574]
   or to be more energy efficient even with mains power. The
   expectation is that waking up doesn't take much time and power
   otherwise the benefits of sleeping are greatly reduced. Initially
   sleeping hosts were esoteric sensor nodes, but this sleeping hosts
   have become mainstream in smartphones.

   Some of the above trends were observed early (e.g., Ohta-san

commented on Neighbor Discovery being inefficient on WiFi a long time
back) but the issues were not broadly understood. The issues were
raised in the 6LowPAN context where the desire was to run IPv6 over
low-power radio networks and battery operated devices. That lead to
defining a set of optimizations [RFC6775] for that specific category
of links. However, the trends are not limited to such specific link
types.

This document applies what we have learned from 6LowPAN to all link
types. That includes reusing existing support from base Neighbor
Discovery (such as Redirect messages) and reusing from 6LowPAN-ND
(Address Registration Option). There are additions above and beyond
that to improve the robustness with redundant routers and to support
mixed mode.

The optimizations are done in a way to provide incremental benefits.
As soon as there is at least one router on a link which supports
these optimizations, then the updated hosts on the link can sleep
better, while co-existing on the same link with unmodified hosts.

2. Goals and Requirements

The goal is to remove as much Neighbor Discovery multicast traffic on
the link as possible, and handle Duplicate Address Detection (DAD)
without requiring the hosts to always be awake. While not an
explicit goal, it turned out that the issues in
[I-D.yourtchenko-6man-dad-issues] that are about robustness/
correctness are also addressed as a side effect of supporting sleepy
hosts.

The optimization will be highly effective for links and nodes which
do not support multicast and for multicast networks without MLD
snooping switches. Moreover, in the MLD-snooping networks the MLD
switches will deal with less number of multicasts.

The requirements handle are:

   Remove the use of multicast for DAD and Address Resolution (no
   multicast NS messages), and remove periodic multicast RAs. Some
   multicast RS and RA are needed to handle the arrival of new hosts
   and routers on the link since they need to bootstrap to find each
   other.

   Remove the need for hosts to always be awake to defend their
   addresses by responding to any DAD probes.

Ensure that the protocol is robust against single points of
failure and uses soft state which is automatically rebuilt after a
state loss.

A router which does not support legacy hosts will always maintain
a complete set of Neighbor Cache Entries (NCEs) for all hosts on
the link. Hence there is no need for it to send Neighbor
Solicitations. Thus it can avoid the problem specified in
[RFC6583].

The optimized solution SHOULD be independent of the addresses
allocation mechanism. In addition to supporting SLAAC [RFC4862]
and DHCPv6 [RFC3315] it SHOULD also work with hosts with 'Privacy
Extensions for Stateless Address Autoconfiguration in
IPv6'[RFC4941] and with stable IPv6 private addresses
[I-D.ietf-6man-stable-privacy-addresses] thus it handles the
recommendations in [I-D.ietf-6man-default-iids].

2.1. Mixed-Mode Operations

Mixed-Mode operation is the protocol behavior when the IPv6 subnet is
composed of legacy IPv6 Neighbor Discovery compliant nodes and
efficiency-aware IPv6 nodes implementing this specification.

The mixed-mode model SHOULD support arbitrary combinations of legacy
[RFC4861] hosts and/or routers with new hosts and/or routers on a
link. The introduction of one new router SHOULD provide improved
services to a new host, allowing the new host to avoid multicast and
not requiring the host to be awake to defend its IPv6 addresses using
DAD.

This document assumes that an implementation will have configuration
knobs to determine whether it is running in legacy IPv6 ND [RFC4861]
or Efficiency Aware only mode (no-legacy mode) or both (Mixed mode).

3. Changes to ND state management

These optimizations change some fundamental aspects of Neighbor
Discovery. Firstly, it moves the distributed address resolution
state (each node responding to a multicast NS for its own addresses)
to a set of routers which maintain a list of Address Registrations
for the hosts. Secondly, the information distributed in Router
Advertisements changes from being periodically flooded by the routers
to explicit requests from the hosts for refreshed information using
unicast Router Solicitations.

4. Definition Of Terms

   The keywords "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in [RFC2119].

   IPv6 ND-efficiency-aware Router (NEAR):
      A router that implements the optimizations specified in this
      document. This router should be able to handle both legacy IPv6
      nodes and nodes that sends registration request.

   Efficiency-Aware Host (EAH):
      The EAH is the host which implements the host functionality for
      optimized Neighbor Discovery mentioned in this document. At least
      initially implementations are likely to have a fallback to legacy
      Neighbor Discovery when no NEAR is on the link.

   Legacy IPv6 Host:
      A IPv6 host that implements [RFC4861] without the extensions in
      this document.

   Legacy IPv6 Router:
      A IPv6 router that implements [RFC4861] without the extensions in
      this document.

   Mixed mode
      A NEAR supports both legacy hosts and EAH, with a configuration
      knob to disable the support for legacy hosts. Some routers on the
      link can be legacy and some can be NEAR.

   No-legacy mode
      A mode configured on a NEAR to not support any legacy [RFC4861]
      hosts or routers. Opposite of mixed mode.

   IPv6 Address Registrar
      Normally the default router(s) on the link will act as IPv6
      Address Registrars tracking all the EAHs. But in some cases it is
      more efficient to use a different set of routers as Address
      Registrars. The hosts are informed of the address registrars
      using router advertisement messages, and register with the
      available registrars.

   EUI-64:
      It is the IEEE defined 64-bit extended unique identifier formed by
      concatenation of 24-bit or 36-bit company id value by IEEE
      Registration Authority and the extension identifier within that
      company-id assignment. The extension identifiers are 40-bit (for
      24-bit company-id) or 28-bit (for the 36-bit company-id)

respectively. The protocol supports EUI-64 for compatibility with
[RFC6775].

DUID
   It is a DHCP Unique ID of a device [RFC3315]. The DUID is assumed
   to be stable in a given IPv6 subnet. A device which does not have
   an EUI-64 can form and use a DUID in its address registrations.

NCE
   Neighbor Cache Entry. It is a conceptual data structure
   introduced in section 5.1 of [RFC4861] and further elaborated in
   [RFC6775].

TID
   The Transaction ID is a device generated sequence number used for
   registration. This number is used to allow a host to have
   concurrent registrations with different routers, while also being
   able to robustly replace a registration with a new one. It
   facilitates interoperability with protocols like RPL [RFC6550]
   which use a TID internally to handle host movement.

5. Protocol Overview

   In a nutshell, the following basic optimizations are made from the
   original IPv6 Neighbor Discovery protocol [RFC4861]:

   o Adds Node Registration with the default router(s).

   o Address Resolution and DAD uses the registered addresses instead
      of multicast Neighbor Solicitation messages for non-link-local
      IPv6 addresses.

   o Does not require unsolicited periodic Router Advertisements.

   o Supports mixed-mode operation where legacy IPv6 hosts and routers
      and NEARs and EAHs can co-exist on the same link. This support
      can be configured off.

   When a host powers on it behaves conforms to legacy ND [RFC4861] by
   multicasting up to MAX_RTR_SOLICITATIONS Router Solicitations and
   receives Router Advertisements. The additional information in the
   Router Advertisements by the NEARs is used by the EAH to build a list
   of IPv6 Address Registrars. As the host is forming its IPv6
   addresses (using any of the many stateless and stateful IPv6 address
   allocation mechanism) then, instead of using a multicast DAD message,
   it unicasts an Neighbor Solicitation with the new Address
   Registration Option (ARO) to the Registrars. Assuming an IPv6

addresses are not duplicate the EAH receives a Neighbor Advertisement with the ARO option from the NEARs. The EAH refreshes the registered addresses before they expire, thereby removing the need for the EAH to be awake to defend its addresses using DAD as specified in [RFC4862] as the NEARs will handle DAD.

The routers on the link advertise the prefixes without setting the L flag. Thus only the IPv6 link-local addresses are considered on-link. Thus the hosts will send packets to a default router, and the default routers maintain all the registrations. Hence a router will know the link-layer addresses of all the registered hosts. This enables the router to forward the packet (without needing any Address Resolution with the multicast Neighbor Solicitation), and also to send a Redirect to the originating host informing the host of the link-layer address.

Instead of relying on periodic multicast RAs to refresh the lifetimes of prefixes etc., the hosts ask for refreshed information by unicasting a Router Solicitation before the information expires. Note that [I-D.nordmark-6man-rs-refresh] make that behavior more explicit by having the routers advertise a timeout.

The periodic multicast RAs may be used to provide new information such as additional prefixes, radical reduction in the preferred and/or valid lifetime for a prefix. A host does not know to ask for such information. Thus a router that wishes to quickly disseminate such change can resort to a few multicast RAs, or wait for the hosts to request a refresh using a Router Solicitation.

The routers can crash and loose all their state, including the Address Registrations. On router initialization the router will multicast a few initial RAs. The protocol has a Router Epoch mechanism which is used by the hosts to detect that the router has lost state. In that case the hosts will immediately re-register allowing the router to quickly rebuild its Address Registration state.

Normally it is sufficient for the hosts to register with all the default routers on the link. However, in some cases such as simplistic VRRP deployment the hosts should register with all the VRRP routers even though they only know of one virtual router IPv6 address. And in other cases it would be more efficient to only register with one router even though there are multiple default routers. The RAs can contain a Registrar Address Option to explicitly tell the hosts where to register.

Sleepy hosts are supported by this Neighbor Discovery procedures as they are not woken up periodically by Router Advertisement multicast

messages or Neighbor Solicitation multicast messages. Sleepy nodes
may wake up in its own schedule and send unicast registration refresh
messages before the registration lifetime expiration. The
recommended procedure on wakeup is to unicast a Neighbor Solicitation
to the default router(s), which serves as DNA check [RFC6059] that
the host is on the same link, performs NUD against the router, and
includes the Address Registration Option to refresh the registration.

5.1. Proxying to handle Mixed mode

When there are one or more legacy routers on the link then the
recommendation is to configure those to advertise the prefixes with
L=0 just as the NEARs. That results in the hosts sending all packets
to a default router unless/until they receive a Redirect. However,
the legacy routers do not maintain the address registrations. Thus
even though all the hosts send the packets to the routers, the legacy
routers might in turn need to perform Address Resolution by
multicasting a Neighbor Solicitation per [RFC4861]. In addition,
legacy hosts and legacy routers will perform DAD as specified in
[RFC4862] that is, by sending a multicast NS and waiting for a NS or
NA which indicates a conflict. A EAH will not receive and respond to
such messages.

If the NEARs have been configured to operate in mixed-mode, then they
will respond to multicast NS messages from legacy nodes for both DAD
and Address Resolution. They will respond with the Target Link Layer
Address being that of the registered host, so that subsequent
communication will not go via the routers. (Implementations of
"Neighbor Discovery Proxies (ND Proxy)" [RFC4389] might proxy using
their own MAC address as TLLA, but that is outside of the scope of
this document.)

6. New Neighbor Discovery Options and Messages

This specification introduces a new flag in the RAs, reuses and
extends the ARO option from [RFC6775] and introduces a new Registrar
Address option.

6.1. Router Advertisement flag for NEARs

A new Router Advertisement flag is needed in order to distinguish a
router advertisement sent by a NEAR, which will trigger an EAH to
register with the NEAR. This flag is ignored by the legacy IPv6
hosts.

The current flags field in RA is reproduced here with the added 'E'
bit.

```
              0 1 2 3 4 5 6 7
              +-+-+-+-+-+-+-+-+
              |M|O|H|Prf|P|E|R|
              +-+-+-+-+-+-+-+-+
```

The 'E' bit is set to 1 in a RA sent by a NEAR. In all other cases
the E bit MUST be 0.

6.2. Address Registration Option (ARO)

The Address Registration Option was defined in [RFC6775] for the
purposes of 6LowPAN and this document extends it in a backwards
compatible way by using some of the reserved fields. The extensions
are to handle different unique identifiers than EUI-64 (this document
specifies how to use DHCP Unique Identifiers with the ability do use
other identifier namespaces in the future) and a Transaction Id.

The Unique Interface Identifier is used by the NEARs to distinguish
between a refresh of an existing registration and a different host
trying to register an IPv6 address which is already registered by
some other host. Thus the requirement is that the unique id is
unique per link, but due to the potential for host mobility across
links and subnets it should be globally unique. Both an EUI-64 and a
DUID satisfies that requirement.

The TID is used by the NEARs to handle the case when due to packet
loss one NEAR might have a old registration and another NEAR has a
newer registration. The TID allows them to determine which is more
recent. The TID also facilitates the interaction with RPL [RFC6550].

An Address Registration Option can be included in unicast Neighbor
Solicitation (NS) messages sent by hosts. Thus it can be included in
the unicast NS messages that a host sends as part of Neighbor
Unreachability Detection to determine that it can still reach the
default router(s). The ARO is used by the receiving router to
reliably maintain its Neighbor Cache. The same option is included in
corresponding Neighbor Advertisement (NA) messages with a Status
field indicating the success or failure of the registration.

When the ARO is sent by a host then, for links which have link-layer
addresses, a SLLA option MUST be included. The address that is
registered is the IPv6 source address of the Neighbor Solicitation
message. Typically a host would have several addresses to register,
with each one being registered using a separate NS containing an ARO.
(This approach facilitates applying SeND [RFC3971].)

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |     Length    |     Status    |   Reserved    |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Res | IDS |T|     TID       |    Registration Lifetime      |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~          Unique Interface Identifier (variable length)       ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
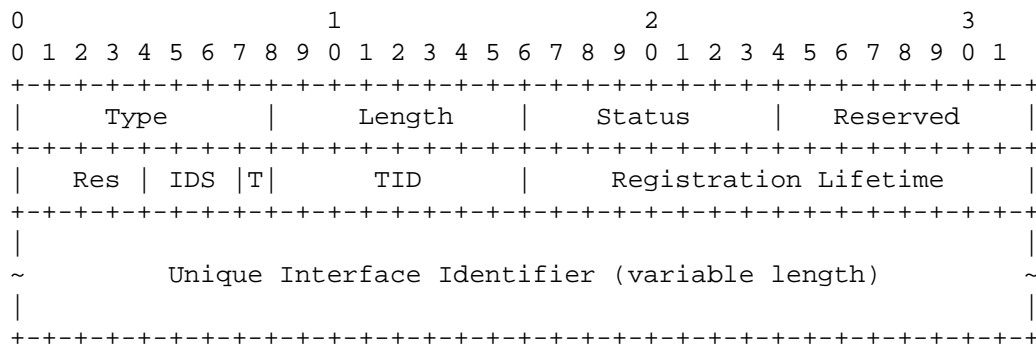
Fields:

Type: 33 [RFC6775]

Length: 8-bit unsigned integer. The length of the option
                (including the type and length fields) in units of 8
                bytes. The value 0 is invalid.

Status: 8-bit unsigned integer. Indicates the status of a
                registration in the NA response. MUST be set to 0 in
                NS messages. See [RFC6775].

Reserved: 8 bits. This field is unused. It MUST be initialized
                to zero by the sender and MUST be ignored by the
                receiver.

Res: 4 bits. This field is unused. It MUST be initialized
                to zero by the sender and MUST be ignored by the
                receiver.

IDS: 3 bits. Identifier name Space. Indicates whether the
                Unique Interface Identifier is a DUID or or a IEEE
                assigned EUI-64 with room to define additional name
                spaces.

T bit: One bit flag. Set if the TID octet is valid.

TID: 8-bit integer. It is a transaction id maintained by
                the host and used by the NEARs to determine the most
                recent registration.

Registration Lifetime: 16-bit unsigned integer. The amount of time
                in a unit of 60 seconds that the router should retain
                the Neighbor Cache entry for the sender of the NS that
                includes this option. A value of zero means to remove

the registration.

Unique Interface Identifier: Variable length in multiples of 8
             bytes. If the IDS=000, then it is an 8 byte (64 bit)
             unmodified EUI-64. If IDS=0011 then it is a variable
             length DUID. A DUID MUST be zero padded to a multiple
             of 8 bytes.

When a node includes ARO option in a Neighbor Solicitation it MUST,
on links that have link-layer addresses, also include a SLLA option.
That option is needed so that the registrar can record the link-layer
address on success and send back an error if the address is a
duplicate.

6.3. Registrar Address Option (RAO)

Normally the Registrars are the Default Routers. However, there are
cases (like some approaches to handle VRRP, or coordinated separate
routers) where there is a need to have different (and either more or
less) Registrars than Default Routers. Furthermore, to robustly
handle NEAR state state loss this option carries a Router Epoch which
triggers the EAHs to re-register on a router epoch change. The RAO
contains the information for both of those.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |    Length     |        Num Addresses          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Reserved             |        Router Epoch           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                   IPv6 registrar addresses                    ~
|                      (Num Addresses)                          |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                                                               |
~                 Reserved for future extensions                ~
|                                                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
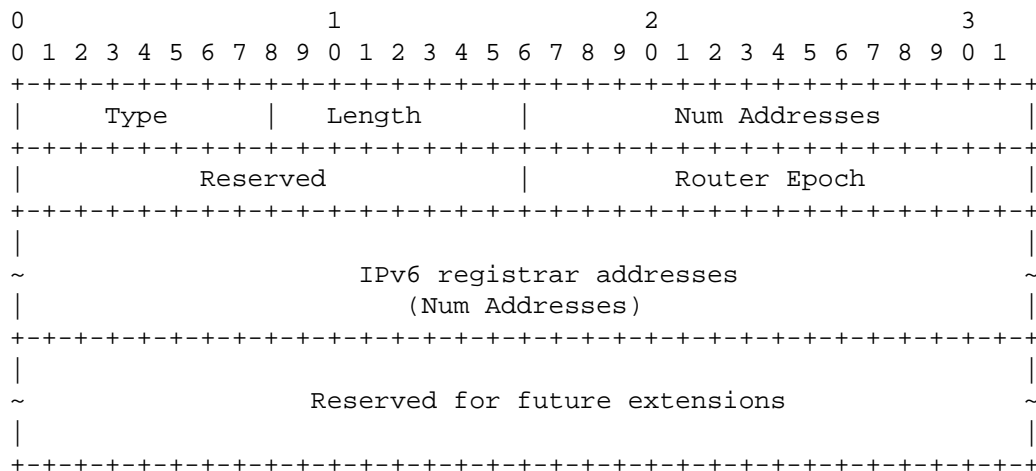
Fields:

Type: TBD (IANA)

Length: 8-bit unsigned integer. The length of the option
              (including the type and length fields) in units of 8
              bytes. The value 0 is invalid.

Num Addresses 16-bit unsigned integer. Set to zero if there are no
               addresses i.e., when the option is used to only carry
               the router epoch. NumAdressses*2 + 1 must not exceed
               the Length.

Reserved   16-bit unused field. It MUST be initialized to zero
               by the sender and MUST be ignored by the receiver.

Router epoch   16-bit integer. A router MUST pick a new epoch after
               a state loss, either by keeping the epoch in stable
               storage and incrementing it, or picking a good random
               number.

IPv6 registrar addresses Zero or more IPv6 addresses, typically of
               link-local scope.

The receiver MUST silently ignore any data after the IPv6 registrar
addresses field (such data is present when the Length is greater than
NumAdressses*2 + 1).

The Registrar Addresses are subject to the same lifetime as the
Default Router Lifetime (thus there is no explicit lifetime field in
the RAO).

7. Conceptual Data Structures

In addition to the Conceptual Data structures in [RFC4861] a EAH
needs to maintain the new Registrar List for each interface. The
Registrar List contains the set of IP addresses to which the host
needs to send Address Registrations. Each IP address has a Router
Epoch (used to determine when a router might have lost state). Also,
the host MAY use this data structure to track when it needs to
refresh its registrations with the registrar.

The use of explicit registrations with lifetimes plus the desire to
not multicast Neighbor Solicitation messages for hosts imply that we
manage the Neighbor Cache entries slightly differently than in
[RFC4861]. This results in two different types of NCEs and the types
specify how those entries can be removed:

Legacy: Entries that are subject to the normal rules in
                    [RFC4861] that allow for garbage collection
                    when low on memory. Legacy entries are created
                    only when there is no duplicate NCE. The
                    legacy entries are converted to the registered
                    entries upon successful processing of ARO.
                    Legacy type can be considered as union of

                              garbage-collectible and Tentative Type NCEs
                              described in [RFC6775].

   Registered: Entries that have an explicit registered
                              lifetime and are kept until this lifetime
                              expires or they are explicitly unregistered.

   Note that the type of the NCE is orthogonal to the states specified
   in [RFC4861]. There can only be one type of NCE for an IP address at
   a time.

8. Host Behavior

   A EAH follows [RFC4861] and applicable parts of [RFC6775] as
   specified in this section./

   A EAH implementation MAY be able to fall back to [RFC4861] host
   behavior if there is no NEAR on the link.

8.1. Host and/or Interface Initialization

   A host multicasts Router Solicitation at system startup or interface
   initialization as specified in [RFC4861] and its updates such as
   [I-D.ietf-6man-resilient-rs]. If the interface initialization is due
   to potential host movement or a wakeup from sleep then the host
   initially sends a unicast Neighbor Solicitation to the default
   router(s).

   Unlike RFC4861 the RS MUST (on link layers which have addresses)
   include a SLLA option, which is used by the router to unicast the RA.

   The host is not required to join the solicited-node multicast
   address(es) but it MUST join the all-nodes multicast address.

8.2. Host Receiving Router Advertisements

   The RA is validated and processed as specified in [RFC4861] with
   additional behavior for RAO and the Registrar List as follows.

   When a RA is received without a RAO (but with the E flag set), or the
   RAO contains no Registrar Addresses, then the IPv6 source address is
   added/updated in the Registrar List. When a RA is received with a
   RAO then the IPv6 Registrar Addresses in that option are added/
   updated in the data structure.

   If those Registrar List (or entries) already exist and the Router
   Epoch in the RAO differs from the Router Epoch in the Registrar List

entry, or if the entry does not exist, then the host will initiate
sending NS messages with ARO options to the new/updated Registration
List entries. Note that if the RA contains no RAO (but the E flag is
set) then for the purposes of the epoch comparison one should use a
zero Router Epoch.

However, if the Default Router Lifetime in the RA is zero, then any
matching Registration List entry (or entries) are instead deleted
from the Registration List. It is OPTIONAL for the host to de-
register when an entry is deleted from the Registration List.

The host can form its IPv6 address using any available mechanism -
SLAAC, DHCPv6, temporary addresses, etc - as the registration
mechanism is orthogonal and independent of the address allocation.
The Address Registration procedure replaces the DAD procedure in
[RFC4862].

8.3. Timing out Registrar List entries

The lifetime for the Registrar List entries are taken from the
Default Router Lifetime in the RA. When an entry is removed the host
MAY send AROs with a zero Registration Lifetime to the removed
Registrar Addresses.

8.4. Sending AROs

When a host has formed a new IPv6 address, or when the host learns of
a new NEAR and has existing IPv6 addresses, then it would register
the new things (could be new addresses to all the existing
Registrars, or the all the IPv6 addresses with the new Registrar.
IPv6 link-local addresses are registered as well as the global
addresses and ULAs.

If the EAH has a TID then it sets the T-bit and includes the TID in
the ARO. When the host registers its addresses with multiple
Registrars it uses the same TID. However, if the host has moved
(lost its network attachment and determines it is attached to a
different link using e.g., DNA [RFC6059]), then it will increment the
TID value and use the new value for subsequent registrations.

The host places its Unique Interface Identifier (whether it is a DUID
or EUI-64) in the ARO. This identifier is typically kept in stable
storage so that the host can use the same identifier over time. It
MUST use the same identifier when it re-registers its address, since
otherwise all those will be returned as duplicates.

The NS which includes the ARO option MUST include a SLLA option on
link layers that have link-layer addresses.

The EAH retransmits NS messages with ARO as specified in [RFC6775] until it receives a NA message from the Registrar containing an ARO. The number of such retransmissions SHOULD be configurable.

8.5. Receiving Neighbor Advertisements

The Neighbor Advertisement are validated and processed as specified in [RFC4861] for example to handle Neighbor Unreachability Detection (NUD). In addition, the host processes any received ARO as follows.

If the ARO has status code 0 (Success), then the host updates the information in the Registrar List to know when it next needs to refresh the registered address with this Registrar. No further processing is needed of the ARO.

If the ARO has status code 1 (Duplicate), then the host can not use the IPv6 address. The host follows the address allocation protocol it used to pick a new address - be that DHCPv6, temporary addresses, etc.

If the ARO has a status code 2 (Neighbor Cache Full) in response to its registration request from a Registrar, then the node SHOULD continue to register the address with other Registrars (when available).

TBD: What about other not yet defined status code values?

8.6. Host Management of the TID

It is RECOMMENDED that the EAH MAY maintain a sequence counter (TID) in stable storage according to section 7 of [RFC6550]. The TID is used to resolve conflicts between different registrations issues by the same host for the same IPv6 address. Conflicts can be due to different link-layer addresses, but it can also be due to registering with different NEARs/Registrars and those routers connect use protocols like RPL for routing, and RPL uses a TID to handle movement vs. multi-homing.

Thus the same TID should be used if the host is registering its addresses with multiple Registrars at the same time. But if the host might have moved to a different attachment point, then it should increment its TID for subsequent registrations.

8.7. Refreshing a Registration

A host SHOULD send a Registration message in order to renew its registration before its registration lifetime expires in order to continue its connectivity with the network.

This specification does not constrain the implementation. One
possible implementation strategy is to attempt re-register at 1/3rd
of the registration lifetime, and if no response try again at 2/3rd
of the lifetime, etc. Another possible strategy is to wait until the
end of the registration lifetime and then do the same relatively
rapid retransmissions as for the initial registration [RFC6775]. In
all cases the host SHOULD apply a random factor to its re-
registration timeout to avoid self-synchronizing behavior across lots
of hosts. Sleeping hosts would re-register when they are waking up
to do other work.

8.8. De-registering

If anytime, the node decides that it does not need a particular
default router's service anymore, the it SHOULD send a de-
registration message to that NEAR/Registrar. Similarly if the host
stops using a particular IPv6 address, then it SHOULD de-register
that address with all the Registrars it had registered with. This
applies even if the host was using the IPv6 address, then went to
sleep, and then picked a different set of IPv6 addresses. In such a
case it is preferred if the host de-registers before going to sleep.
A mobile host SHOULD first de-register its addresses before it moves
away from the subnet (if the mobile host can know that in advance of
moving.)

De-registration is performed by unicasting a Neighbor Solicitation
with an ARO where the Registration Lifetime is set to zero. Such an
ARO should have an incremented TID. De-registration AROs are
retransmitted just like other AROs as specified above.

8.9. Refreshing RA information

The EAH is responsible for asking the routers for updates to the
information contained in the Router Advertisements, since the NEARs
will not multicast such updates. That is done by sending unicast RSs
to the router(s) which will result in unicast RAs. However,
significant care is required in determining when the RSs should be
transmitted.

As part of normal operation the Default Routers, Prefixes, and other
RA information have lifetimes, and there are a few common cases:

1. The advertised lifetimes are constant i.e., the routers keep on
    advancing the time when the information will expire.

2. The routers decrement the advertised lifetimes in real time i.e.,
    the information is set to expire at a determined time and
    subsequent RAs have lower and lower lifetimes.

3. The routers forcibly expire some information by advertising it
    with a zero lifetime for a while, and then stop advertising it.

4. A router crashes or is silently removed from the network and as a
    result there are no more updates. For example, that default
    router will expire and there is little benefit in trying to
    refresh it by sending lots of RSs.

The host's logic for refreshing the information needs to be careful
to not send a large number of RSs, in particular if there is
information that is supposed to expire at a fixed time i.e., the
lifetime decrements in real time.

A host MUST NOT try to refresh information because its lifetime is
near zero, since that would cause unnecessary RSs. Instead the
refresh needs to be based on when the information was most recently
received from the router. A lifetime of 10 minutes that was heard
from the router 10 minutes ago might be normal as part of expiring
some information. But a remaining lifetime of 10 minutes for a
prefix that was last heard 24 hours ago with a lifetime of 24 hours
means that a refresh is in order.

It is RECOMMENDED that the host track the expiry time (the wall clock
time when some information will expire) and when it receives an RA
with that information it SHOULD check whether the expiry time is
moving forward, or appears to be frozen in time. That can tell the
difference between he first two cases above, and avoid unnecessary
RSs as some information naturally expires. Furthermore it is
RECOMMENDED that the host track which information was received from
which router, so that it can see when a router used to provide the
information no longer provides it. That helps to see if the third
case above might be in play. Finally, if a router has not responded
to a few (e.g., MAX_RTR_SOLICITATIONS) attempts to get a refresh,
then the router might be unreachable or dead, and there is little
benefit in sending further RSs to that router. When the router comes
back it will multicast a few RAs.

When the hosts determines that case 1 above is likely, then it should
pick a reasonable time to ask for refreshes. The exact refresh
behavior is not mandated by this specification, but the same
implementation strategies as for refreshing address registrations in
Section 8.7 can be considered.

A example simple implementation approach is to only base the
refreshing on the default router lifetime (thus ignore prefix and
other lifetime), and pick a refresh time which is 1/3 of the default
router lifetime. If no RA is received, a subsequent refresh can be
done at 2/3 of the default router lifetime. If that does not result

in a RA, then MAX_INITIAL_RTR_ADVERTISEMENTS can be sent as the
router lifetime is about to expire. Note that a default router
lifetime of zero MUST NOT result in sending a RS refresh based on a
timeout of zero.

If the host is unable to refresh the information and as a result ends
up with an empty default router list, or all the default routers are
marked as UNREACHABLE by NUD, then the host MAY switch to sending
initial multicast Router Solicitations as in Section 8.1.

Note that [I-D.nordmark-6man-rs-refresh] make that behavior more
explicit by having the routers advertise a timeout.

8.10. Sleep and Wakeup

The protocol allows the sleepy nodes to complete its sleep schedule
without waking up due to periodic Router Advertisement messages or
due to Multicast Neighbor Solicitation for address resolution. The
node registration lifetime SHOULD be related with its sleep interval
period in order to avoid waking up in the middle of sleep for
registration refresh. Depending on the application, the registration
lifetime SHOULD be equal to or integral multiple of a node's sleep
interval period.

When a host wakes up it can combine movement detecting (DNA), NUD,
and refreshing its Address Registration by sending a unicast NS with
an ARO to its existing default router(s).

8.11. Receiving Redirects

An EAH handles Redirect messages as specified in [RFC4861].

8.12. Movement Detection

When a host moves from one subnet to another its IPv6 prefix changes
and the movement detection is determined according to the existing
IPv6 movement detection described in [RFC6059].

9. Router Behavior

A NEAR follows [RFC4861] and applicable parts of [RFC6775] as follows
in this section.

A NEAR SHOULD support legacy hosts and mixed mode as specified in
this section by being able to proxy Address Resolution and DAD. The
NEAR SHOULD implement a knob to be able to disable this behavior.
That knob can either be set to "mixed-mode" or to "no-legacy-mode".

The RECOMMENDED default mode of operation for the NEAR is Mixed-mode.

A NEAR should be configured to advertise prefixes without the on-link (L-bit) unset. Furthermore, any legacy routers attached to the same link as a NEAR should be configured the same way. That reduces the cases in mixed mode when multicast NS messages are needed between legacy hosts and routers.

9.1. Router and/or Interface Initialization

A NEAR multicasts some initial Router Advertisements (MAX_INITIAL_RTR_ADVERTISEMENTS) at system startup or interface initialization as specified in [RFC4861] and its updates.

The NEAR MUST join the all-nodes and all-routers multicast addresses. In mixed mode it MUST also join the solicited-node multicast address(es) for its addresses and also for all the Registered NCEs.

A NEAR picks a new Router Epoch if it has lost the Registered NCEs, which is typically the case for router initialization. Either the Router Epoch can be stored in stable storage and incremented on each router initialization, or the NEAR can pick a good random number on router initialization. The effect of occasionally picking the same Router Epoch as before the state loss is that it will take longer for the router to build up its state of Registered NCEs.

9.2. Receiving Router Solicitations

Periodic RAs SHOULD be avoided. Only solicited RAs are RECOMMENDED. An RA MUST contain the Source Link-layer Address option containing Router's link-layer address (this is optional in [RFC4861]. An RA MUST carry any Prefix information option with L bit being unset, so that hosts do not multicast any NS messages as part of address resolution. A new flag (E-flag) is introduced in the RA which the hosts use to distinguish a NEAR from a legacy router.

Unlike [RFC4861] which suggests multicast Router Advertisements, this specification optimizes the exchange by always unicasting RAs in response to RSs. This is possible since the RS always includes a SLLA option, which is used by the router to unicast the RA.

If the NEAR has been configured to send an explicit set of IPv6 Registrar Addresses, or implements a Router Epoch, then the NEAR includes a RAO in all its RAs.

9.3. Periodic Multicast RA for legacy hosts

   The NEAR MUST NOT send periodic RA in no-legacy mode. In mixed mode
   the NEAR needs to send periodic multicast RAs as specified in
   [RFC4861] to support legacy hosts.

9.4. Multicast RA when new information

   When a router has new information to share (new prefixes, prefixes
   that should be immediately deprecated, etc) it MAY multicast up to
   MAX_INITIAL_RTR_ADVERTISEMENTS number of Router Advertisements. Note
   that such new information is not likely to reach sleeping hosts until
   those hosts refresh by sending a RS.

9.5. Receiving ARO

   The NEAR follows the logic in [RFC6775] for managing the NCEs and
   responding to NS messages with the ARO option. The slight
   modification is that instead of using an EUI-64 as the key to check
   for duplicates, the NEAR uses the IDS value plus the variable length
   Unique Interface Identifier value as the key. In addition the NEAR
   checks the new TID field as follows.

   The TID field is used together with age of a registration for
   arbitration between two routers to ensure freshness of the
   registration of a given target address. Same value of TID indicates
   that both states of registration are valid. In case of a mismatch
   between comparable TIDs, the most recent TID wins. The TIDs are
   compared as specified in section 7 of [RFC6550].

9.6. NCE Management in NEARs

   When a host interacts with a router by sending Router Solicitations
   that does not match with the existing NCE entry of any type, a Legacy
   NCE is first created. Once a node successfully registers with a
   Router the result is a Registered NCE. As Routers send RAs to legacy
   hosts, or receive multicast NS messages from other Routers the result
   is Legacy NCEs.

   A Router Solicitation might be received from a host that has not yet
   registered its address with the router or from a legacy [RFC4861]
   host in the Mixed-mode operation.

   The router MUST NOT modify an existing Registered Neighbor Cache
   entry based on the SLLA option from the Router Solicitation. Thus, a
   router SHOULD create a tentative Legacy Neighbor Cache entry based on
   SLLA option when there is no match with the existing NCE. Such a
   legacy Neighbor Cache entry SHOULD be timed out in

TENTATIVE_LEGACY_NCE_LIFETIME seconds unless a registration converts
it into a Registered NCE.

However, in 'Mixed-mode' operation, the router does not require to
keep track of TENTATIVE_LEGACY_NCE_LIFETIME as it does not know if
the RS request is from a legacy host or from a EAH. However, it
creates the legacy type of NCE and updates it to a registered NCE if
the ARO NS request arrives corresponding to the Legacy NCE.
Successful processing of ARO will complete the NCE creation phase.

If ARO did not result in a duplicate address being detected, and the
registration life-time is non-zero, the router creates or updates the
Registered NCE for the IPv6 address. If the Neighbor Cache is full
and new entries need to be created, then the router SHOULD respond
with a NA with status field set to 2. For successful creation of
NCE, the router SHOULD include a copy of ARO and send NA to the
requester with the status field 0. A TLLA (Target Link Layer) Option
is not required with this NA.

Typically for efficiency-aware routers (NEAR), the Registration
Lifetime and IDS plus Unique Interface Identifier are recorded in the
Neighbor Cache Entry along with the existing information described in
[RFC4861]. The registered NCE are meant to be ready and reachable
for communication and no address resolution is required in the link.
An EAH will renew its registration to Registered NCE at the router.
However the router may perform NUD towards the EAH hosts as per
[RFC4861]. Should NUD fail the NEAR MUST NOT remove the Registered
NCE. Instead it marks it as UNREACHABLE.

9.7. Sending non-zero status in ARO

If the Registration fails (due to it being a duplicate or the
Neighbor Cache being full), then the NEAR will send an NA with ARO
having a non-zero status. However, it needs to send that back to the
originator of the failing ARO, and that host and link-layer address
will not be present in the Neighbor Cache.

The NEAR forms a NA with ARO, and then forms the link-layer address
by using the content of the SLLA option in the NS, bypassing the
Neighbor Cache to send this error.

9.8. Timing out Registered NCEs

The router SHOULD NOT garbage collect Registered Neighbor Cache
entries since they need to retain them until the Registration
Lifetime expires. If a NEAR receives a NS message from the same host
one with ARO and another without ARO then the NS message with ARO
gets the precedence and the NS without ARO is ignored.

   Similarly, if Neighbor Unreachability Detection on the router
   determines that the host is UNREACHABLE (based on the logic in
   [RFC4861]), the Neighbor Cache entry SHOULD NOT be deleted but be
   retained until the Registration Lifetime expires. If an ARO arrives
   for an NCE that is in UNREACHABLE state, that NCE should be marked as
   STALE.

   The NEAR router SHOULD deny registration to a new registration
   request with the status code 2 when it reaches the maximum capacity
   for handling the neighbor cache.

9.9. Router Advertisement Consistency

   The NEAR follows section 6.2.7 in [RFC4861] by receiving RAs from
   other routers (NEAR and legacy) on the link. In addition to the
   checks in that section it verifies that the prefixes to not have the
   L flag set, and that the Registrar Address options are consistent.
   Two RAOs are inconsistent if they contain the have a different Router
   Epoch and have some IPv6 Registration Addresses in common.

9.10. Sending Redirects

   A NEAR sends redirects (with target=destination) to inform the hosts
   of the link-layer address of the nodes on the link.

   This can be disabled on specific link types for instance, radio link
   technologies with hidden terminal issues.

9.11. Providing Information to Routing Protocols

   If there is a routing protocols like RPL which wants visibility into
   the location of each IPv6 address, then this can be retrieved from
   the Registered NCEs on the NEAR.

9.12. Creating Legacy NCEs

   In mixed-mode a NEAR will create Legacy NCEs when receiving RA, RS,
   and NS messages based on the source of those packets. When not it
   mixed-mode it needs to create Legacy NCEs for other routers by
   looking at those packets.

9.13. Proxy Address Resolution and DAD for Legacy Hosts

   This section applies in mixed mode. It does not apply in no-legacy
   mode.

   A NEAR in mixed mode MUST join all solicited-node for all Registered
   NCEs.

The NEAR SHOULD continue to support the legacy IPv6 Neighbor
Solicitation requests in the mixed mode. The NEAR router SHOULD act
as the ND proxy on behalf of EAH hosts for the legacy nodes' NS
requests for EAH. This form of proxying is to respond with a NA that
has a TLLAO taken from the Registered NCE for the target. Thus it is
unlike ND Proxy as specified in [RFC4389].(Implementations of
"Neighbor Discovery Proxies (ND Proxy)" [RFC4389] might proxy using
their own MAC address as TLLA, but that is outside of the scope of
this document.)

In the context of this specification, proxy means:

o Responding to DAD probes for a registered NCE. A DAD probe from a
   legacy host would not contain any ARO, hence the NEAR will assume
   it is always a duplicate if the IPv6 target address has a
   Registered NCE.

o Defending a registered address using NA messages with and ARO
   option and the Override bit set if the ARO option in the NS
   indicates either a different node (different IDS+Unique Interface
   Id) or a older registration (when comparing the TID).

o Advertising a registered address using NA messages, asynchronously
   or as a response to a Neighbor Solicitation messages.

o Looking up a destination on the link using Neighbor Solicitation
   messages in order to deliver packets arriving for the EAH.

The NEAR SHOULD also support DAD from a EAH for IPv6 address that
might be in use by a legacy node. Thus when a NEAR in mixed-mode
received an ARO for a new address it SHOULD perform DAD as specified
in [RFC4862] by sending a multicast DAD message. Once that times out
the NEAR can respond to the ARO. If a legacy host responds to the
DAD probe, then the NEAR will respond to the ARO with Status=1
(Duplicate Address).

10. Handling ND DoS Attack

   IETF community has discussed possible issues with /64 DoS attacks on
   the ND networks when an attacker host can send thousands of packets
   to the router with an on-link destination address or sending RS
   messages to initiate a Neighbor Solicitation from the neighboring
   router which will create a number of INCOMPLETE NCE entries for non-
   existent nodes in the network resulting in table overflow and denial
   of service of the existing communications.

   The efficiency-aware behavior documented in this specification avoids

   the ND DoS attacks by:

   o Having the hosts register with the default router(s).

   o Having the hosts send their packets via the default router(s).

   o Not resolving addresses for the routing solicitor by mandating
      SLLA option along with RS

   o Checking for duplicates in NCE before the registration

   o On-link IPv6-destinations on a particular link must be registered
      else these packets are not resolved and extra NCEs are not created

   In order to get maximal benefits from the ND-DoS protection from
   Address Registrations, the hosts and routers on the link need to be
   upgraded to NEARs and EAHs, respectively. With some legacy hosts the
   routers will still need to create INCOMPLETE NCEs and send NSs, which
   keeps the DoS opportunity open. However, with fewer legacy hosts the
   lower rate limits can be applied on creation of INCOMPLETE NCEs.

11. Bootstrapping

   The bootstrapping mechanism described here is applicable for the
   efficiency-aware hosts and routers. At the start, the host uses its
   link-local address to send Router Solicitation and then it sends the
   Address Registration Option as described in this document in order to
   verify the IPv6 address. Note that on wakeup from sleep and after
   potential movement to a different link the host initially sends a
   unicast Neighbor Solicitation to the default router(s).

   The following step 3 and 4 SHOULD be repeated for all the IPv6
   addresses that are used for communications.

```
      Node                                            Router

  0.  |[Forms LL IPv6 addr]                               |

  1.  |         ---------- Router Solicitation -------->  |

      |                       [SLLAO]                     |

  2.  |         <-------- Router Advertisement ---------  |

      |                     [PIO + SLLAO]                 |
      |                                                   |

  3.  |         ----- Address Registration (NS) -------->  |

      |                     [ARO + SLLAO]                 |

  4.  |         <-------- Neighbor Advertisement -------  |

      |                 [ARO with Status code]            |

  5. ====> Address Assignment Complete
```
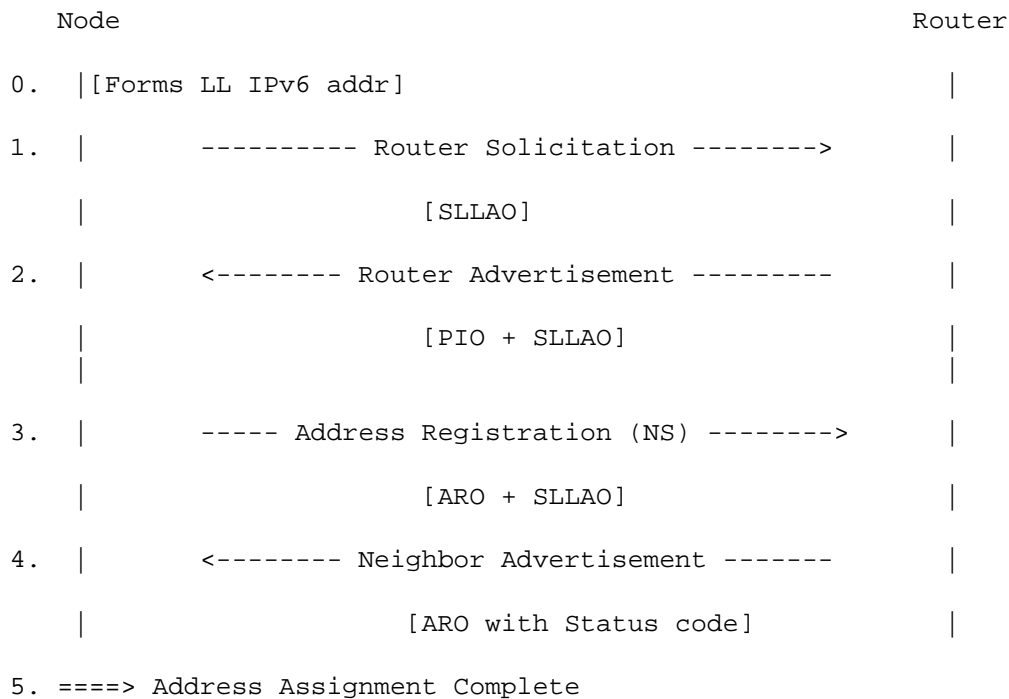
      Figure 1: Neighbor Discovery Address Registration and bootstrapping

12. Interaction with other protocols

12.1. Detecting Network Attachment (DNA)

   IPv6 DNA [RFC6059] uses unicast NS probes and link-layer indications
   to detect movement of its network attachments. That is consistent
   with the mechanism in this specification to unicast a NS when a host
   wakes up - this document recommends adding the ARO to that NS
   message.

   Thus the ND optimization solution will work seamlessly with DNA
   implementations and no change is required in DNA solution because of
   Neighbor Discovery updates. It is a deployment and configuration
   consideration as to run the network in mixed mode or efficient-mode.

12.2. DHCPv6 Interaction

   The protocol mechanisms in this document are orthogonal to the
   address assignment mechanism in use. If DHCPv6 is used for address
   assignment by an EAH then, if there are one or more NEARs on the
   subnet, the EAH will replace the DAD check specified in [RFC3315]

   with Address Registration as specified in this document.

12.3. Other use of Multicast

   Although the solution described in this document prevents unnecessary
   multicast messages in the IPv6 ND procedure, it does not affect
   normal IPv6 multicast packets nor the ability of nodes to join and
   leave the multicast group or forwarding multicast traffic or
   responding to multicast queries.

12.4. VRRP Interaction

   A VRRP set of routers can operate with efficient-nd in two different
   ways:

   o Provide the illusion to hosts that they are a single router for
      the purposes of registrations. No RAO is needed in that case.
      But the pair needs some mechanism to synchronize their neighbor
      caches. Such a mechanism is out of scope of this document.

   o Have the hosts register with each router independently. In that
      case the VRRP router includes the RAO with the individual IP
      addresses of the routers in the pair. No synchronization of the
      neighbor caches are needed in that case.

13. Updated Neighbor Discovery Constants

   This section discusses the updated default values of ND constants
   based on [RFC4861] section 10. New and changed constants are listed
   only for efficiency-aware-nd implementation. These values SHOULD be
   configurable and tunable to fit implementations and deployment.

   Router Constants:

   MAX_RTR_ADVERTISEMENTS(NEW)   3 transmissions

   MIN_DELAY_BETWEEN_RAS(CHANGED)   1 second

   TENTATIVE_LEGACY_NCE_LIFETIME(NEW)   30 seconds

   Host Constants:

   MAX_RTR_SOLICITATION_INTERVAL(NEW)   60 seconds

   Also refer to [RFC6583] , [RFC7048] and [RFC6775] for further tuning
   of ND constants.

14. Security Considerations

   These optimizations are not known to introduce any new threats
   against Neighbor Discovery beyond what is already documented for IPv6
   [RFC3756].

   Section 11.2 of [RFC4861] applies to this document as well.

   This mechanism minimizes the possibility of ND /64 DoS attacks in
   efficiency-aware mode. See Section 10.

   The mechanisms in this document work with SeND [RFC3971] in the no-
   legacy mode. In the mixed mode operation when a NEAR needs to
   respond to a legacy host sending a NS for a EAH, then SeND would not
   automatically fit. Potentially proxy SeND [RFC6496] could be used,
   but that would require explicit awareness and setup between the proxy
   and the proxied EAHs which seems impractical.

   The mechanisms in this specification are orthogonal to the address
   allocation thus works as well with SLAAC and DHCPv6 as the various
   privacy-enhanced address allocation specifications. In particular,
   using an EUI-64 for the Unique Interface Identifier in this protocol
   does not require or assume that the IPv6 addresses will be formed
   using the modified EUI-64 format.

   The mechanism uses a Unique Interface Identifier for the purposes of
   telling apart a re-registration from the same host and a duplicate/
   conflicting registration from a different host. That unique ID is
   not globally visible. Currently the protocol supports DHCPv6 DUID
   and EUI-64 format for this I-D, but other formats which facilitate
   non-linkability (such as strong random numbers large enough to be
   unlikely to cause collisions) can be defined.

15. IANA Considerations

   A new flag (E-bit) in RA has been introduced. IANA assignment of the
   E-bit flag is required upon approval of this document.

   This document needs a new Neighbor Discovery option type for the RAO.

16. Changelog

   Changes from draft-chakrabarti-nordmark-energy-aware-nd-06:

   o Added references to dad-issues and rs-refresh.

Changes from draft-chakrabarti-nordmark-energy-aware-nd-05:

o Fixed typos.

o Clarified that on interface initialization after sleep or
   potential movement the host unicasts a NS to the default
   router(s).

o Simplified the example timer handling for refreshing RA
   information.

o Added handling of DAD from EAH to legacy node that was included in
   -04 and lost in the -05 edits.

Changes from draft-chakrabarti-nordmark-energy-aware-nd-04:

o Significantly simplified the description of the protocol.

o Added clarification on problem statement

o Clarified that privacy and temporary addresses will be supported

o Added an IDS field in the ARO to allow a DHCP Unique ID (DUID) as
   an alternative to EUI-64, with room to define other (pseudo)
   unique identifiers.

o Allowed router redirects for NEAR.

o Addressed some of comments made in the 6man list.

o Added RAO to handle VRRP and similar cases when the default router
   list and registrar list needs to be different.

o Added Router Epoch to cause re-registration on NEAR state loss.

o Specified considerations for when to refresh address
   registrations.

o Specified considerations for when to refresh RA information.

17. Acknowledgements

   The primary idea of this document are from 6LoWPAN Neighbor Discovery
   document [RFC6775] and the discussions from the 6lowpan working group
   members, chairs Carsten Bormann and Geoff Mulligan and through our
   discussions with Zach Shelby, editor of the [RFC6775].

The inspiration of such a IPv6 generic document came from Margaret Wasserman who saw a need for such a document at the IOT workshop at Prague IETF.

The authors acknowledge the ND denial of service issues and key causes mentioned in the draft-halpern-6man-nddos-mitigation document by Joel Halpern. Thanks to Joel Halpern for pinpointing the problems that are now addressed in the NCE management discussion in this document.

The authors like to thank Dave Thaler, Stuart Cheshire, Jari Arkko, Ylva Jading, Niklas J. Johnsson, Reda Nedjar, Purvi Shah, Jaume Rius Riu, Fredrik Garneij, Andrew Yourtchenko, Jouni Korhonen, Suresh Krishnan, Brian Haberman, Anders Brandt, Mark Smith, Lorenzo Colitti, David Miles, Eric Vyncke, Mark ZZZ Smith, Mikael Abrahammsson, Eric Levy-Abignoli, and Carsten Bormann for their useful comments and suggestions on this work.

18. Open Issues

The known open issues are:

o IPv6 link-local addresses are always on-link and in this version of the document that results in multicast NS messages. The technique used in 6LowPAN-ND is too restrictive (extract the link-layer address from the IID). Should we send link-locals to routers and depend on Redirect?

o If the Router Epoch is critical then we will see a RAO in all the RAs sent by NEARs. In such a case we don't need the E-bit in the RA.

o Editorial: Add Comparison with 6lowpan-nd and 4861?

o Editorial: Verify and update the description in this document to make it complete removing the need to read 6LowPAN-ND.

o When a router has new information for the RA, currently it takes a while to disseminate that to sleeping nodes as this depends on when the hosts send a RS. We could potentially improve this is we could have an "information epoch number" in the ARO sent in the NA. But that only helps if the registrations are refreshed more frequently that the RA information.

o Future? Currently if a router changes its information, a sleeping host would not find out when it wakes up and sends the NS with ARO. That could be improved if we fit the Router Epoch in NA/ARO.

But there is no room for 16 bits.

o A separate but related problem is with unused NCEs due to frequent
   IPv6 address change e.g., hosts which pick a different set of
   addresses each time they wake up. This document recommends that
   they be de-registered by the host. That could be made simpler by
   introducing some Host Epoch counter in the NS/ARO.

19. References

19.1. Normative References

   [I-D.ietf-6man-resilient-rs]
             Krishnan, S., Anipko, D., and D. Thaler, "Packet loss
             resiliency for Router Solicitations",
             draft-ietf-6man-resilient-rs-04 (work in progress),
             October 2014.

   [I-D.nordmark-6man-rs-refresh]
             Nordmark, E., Yourtchenko, A., and S. Krishnan, "IPv6
             Neighbor Discovery Optional Unicast RS/RA Refresh",
             draft-nordmark-6man-rs-refresh-01 (work in progress),
             October 2014.

   [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate
             Requirement Levels", BCP 14, RFC 2119, March 1997.

   [RFC4861] Narten, T., Nordmark, E., Simpson, W., and H. Soliman,
             "Neighbor Discovery for IP version 6 (IPv6)", RFC 4861,
             September 2007.

   [RFC6775] Shelby, Z., Chakrabarti, S., Nordmark, E., and C. Bormann,
             "Neighbor Discovery Optimization for IPv6 over Low-Power
             Wireless Personal Area Networks (6LoWPANs)", RFC 6775,
             November 2012.

19.2. Informative References

   [I-D.ietf-6man-default-iids]
             Gont, F., Cooper, A., Thaler, D., and W. Will,
             "Recommendation on Stable IPv6 Interface Identifiers",
             draft-ietf-6man-default-iids-02 (work in progress),
             January 2015.

   [I-D.ietf-6man-stable-privacy-addresses]
             Gont, F., "A Method for Generating Semantically Opaque
             Interface Identifiers with IPv6 Stateless Address

                   Autoconfiguration (SLAAC)",
                   draft-ietf-6man-stable-privacy-addresses-17 (work in
                   progress), January 2014.

   [I-D.vyncke-6man-mcast-not-efficient]
                   Vyncke, E., Thubert, P., Levy-Abegnoli, E., and A.
                   Yourtchenko, "Why Network-Layer Multicast is Not Always
                   Efficient At Datalink Layer",
                   draft-vyncke-6man-mcast-not-efficient-01 (work in
                   progress), February 2014.

   [I-D.yourtchenko-6man-dad-issues]
                   Yourtchenko, A., "A survey of issues related to IPv6
                   Duplicate Address Detection",
                   draft-yourtchenko-6man-dad-issues-00 (work in progress),
                   October 2014.

   [RFC3315]   Droms, R., Bound, J., Volz, B., Lemon, T., Perkins, C.,
                   and M. Carney, "Dynamic Host Configuration Protocol for
                   IPv6 (DHCPv6)", RFC 3315, July 2003.

   [RFC3756]   Nikander, P., Kempf, J., and E. Nordmark, "IPv6 Neighbor
                   Discovery (ND) Trust Models and Threats", RFC 3756,
                   May 2004.

   [RFC3971]   Arkko, J., Kempf, J., Zill, B., and P. Nikander, "SEcure
                   Neighbor Discovery (SEND)", RFC 3971, March 2005.

   [RFC4389]   Thaler, D., Talwar, M., and C. Patel, "Neighbor Discovery
                   Proxies (ND Proxy)", RFC 4389, April 2006.

   [RFC4862]   Thomson, S., Narten, T., and T. Jinmei, "IPv6 Stateless
                   Address Autoconfiguration", RFC 4862, September 2007.

   [RFC4941]   Narten, T., Draves, R., and S. Krishnan, "Privacy
                   Extensions for Stateless Address Autoconfiguration in
                   IPv6", RFC 4941, September 2007.

   [RFC6059]   Krishnan, S. and G. Daley, "Simple Procedures for
                   Detecting Network Attachment in IPv6", RFC 6059,
                   November 2010.

   [RFC6496]   Krishnan, S., Laganier, J., Bonola, M., and A. Garcia-
                   Martinez, "Secure Proxy ND Support for SEcure Neighbor
                   Discovery (SEND)", RFC 6496, February 2012.

   [RFC6550]   Winter, T., Thubert, P., Brandt, A., Hui, J., Kelsey, R.,
                   Levis, P., Pister, K., Struik, R., Vasseur, JP., and R.

          Alexander, "RPL: IPv6 Routing Protocol for Low-Power and
          Lossy Networks", RFC 6550, March 2012.

   [RFC6574] Tschofenig, H. and J. Arkko, "Report from the Smart Object
              Workshop", RFC 6574, April 2012.

   [RFC6583] Gashinsky, I., Jaeggli, J., and W. Kumari, "Operational
              Neighbor Discovery Problems", RFC 6583, March 2012.

   [RFC7048] Nordmark, E. and I. Gashinsky, "Neighbor Unreachability
              Detection Is Too Impatient", RFC 7048, January 2014.

Authors' Addresses

   Samita Chakrabarti
   Ericsson
   San Jose, CA
   USA

   Email: samita.chakrabarti@ericsson.com

   Erik Nordmark
   Arista Networks
   Santa Clara, CA
   USA

   Email: nordmark@arista.com

   Pascal Thubert
   Cisco Systems

   Email: pthubert@cisco.com

   Margaret Wasserman
   Painless Security

   Email: mrw@painless-security.com