Neighbor Discovery Protocol in 6LoWPAN

Fernando Solano Donado
fs@tele.pw.edu.pl
Protocol stack

IP Protocol Stack

HTTP | RTP
---|---
TCP  | UDP | ICMP
IP
Ethernet MAC
Ethernet PHY

6LoWPAN

Application protocols
UDP | ICMP
IP
LoWPAN
IEEE 802.15.4 MAC
IEEE 802.15.4 PHY
Introduction to ICMPv6

- **Internet Control Message Protocol**
  - Performs error reporting and diagnostic functions at the network layer
  - Base for:
    - Echo messages (ping)
    - Neighbor Discovery Protocol
      - Secure Neighbor Discovery Protocol
      - Multicast Router Discovery
    - Routing in Low Power and Lossy Networks
    - ....
  - Errors:
    - Destination Unreachable
    - Packet too big
    - Time Exceeded
    - Parameter Problem
    - ...
6LoWPAN NH = ICMPv6

Type: Echo Request

Src/Dst IPv6 addr are L2 derived

Identifier
Seq Number
Data

Checksum
Code

00000000 28 21 dc 02 00 00 02 00 00 ff fe 00 12 34 00 00
00000010 02 00 00 ff fe 00 f7 38 7a 33 3a 80 00 d4 4d 00
00000020 53 00 03 53 53 53 53 40 19

Ping6 to link-local address
What’s old?

- A link in a Low-power Wireless Personal Area Network (LoWPAN) is characterized as lossy, low-power, low-bit-rate, short-range; with many nodes saving energy with long sleep periods.
- Although a given radio range has broadcast capabilities, the aggregation of these is a complex Non-Broadcast Multiple Access (NBMA) [RFC2491] structure with generally no LoWPAN-wide multicast capabilities.
- Link-local scope is in reality defined by reachability and radio strength. Thus, we can consider a LoWPAN to be made up of links with undetermined connectivity properties as in [RFC5889], along with the corresponding address model assumptions defined therein.
What’s new?

- Host-initiated refresh of Router Advertisement information. This removes the need for periodic or unsolicited Router Advertisements from routers to hosts.
- No Duplicate Address Detection (DAD) is performed if EUI-64-based IPv6 addresses are used (as these addresses are assumed to be globally unique).
- DAD is optional if DHCPv6 is used to assign addresses.
- A new address registration mechanism using a new Address Registration Option between hosts and routers. This removes the need for routers to use multicast Neighbor Solicitations to find hosts and supports sleeping hosts. This also enables the same IPv6 address prefix(es) to be used across a route-over 6LoWPAN. It provides the host-to-router interface for Duplicate Address Detection.
- A new Router Advertisement option, the 6LoWPAN Context Option, for context information used by 6LoWPAN header compression.
- A new mechanism to perform Duplicate Address Detection across a route-over 6LoWPAN using the new Duplicate Address Request and Duplicate Address Confirmation messages.
- New mechanisms to distribute prefixes and context information across a route-over network that uses a new Authoritative Border Router Option to control the flooding of configuration changes.
- A few new default protocol constants are introduced, and some existing Neighbor Discovery protocol constants are tuned.
Network Autoconfiguration in 6LoWPAN
From IEEE802.15.4 long addresses to IPv6

<table>
<thead>
<tr>
<th>64 bits</th>
<th>64 bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prefix</td>
<td>Interface ID</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>EUI-64 (first 6-bits)</th>
<th>U</th>
<th>M</th>
<th>EUI-64 (last 56-bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>U: universal address</td>
<td></td>
<td></td>
<td>M: multicast address</td>
</tr>
</tbody>
</table>

Where is my Prefix?

2001:0DB8:0BAD:FADE:: (prefix)
ACDE:4812:3456:7890 (EUI-64)
2001:0DB8:0BAD:FADE:AEDE:4812:3456:7890 (IPv6)
Network Autoconfiguration

**IPv4**
- No self-assigned valid addresses
  - Self-assigned address for “unassigned”
- Dynamic Host Configuration Protocol (DHCP)

**IPv6**
- Self-assigned addresses using ICMPv6
  - Router Advertisement
  - Router Solicitation
- Dynamic Host Configuration for IPv6 (DHCPv6) for “unusual” cases.

**Router Advertisement:**
This is our global prefix!

**Router Solicitation:**
I need a prefix!
6loWPAN
Where are my Context Ids + Border Router info?

Router Advertisement:
This is our global prefix, Context Ids and Border Router!
NDP–Router Solicitation

Sent periodically while bootstrapping until a RA arrives.

Router solicitation

ICMPv6 checksum

S2LT (01): Source link-layer address – with 2x8 octets of data

Options
<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Router Advertisement</td>
<td>NDP – Router Advertisement</td>
</tr>
<tr>
<td>Cur Hop limit</td>
<td>MO flags</td>
</tr>
<tr>
<td>MO flags</td>
<td>Options: S2LT (01): Source link-layer address – with 2x8 octets of data</td>
</tr>
<tr>
<td>Router Advertisement</td>
<td>Options: S2LT (01): Source link-layer address – with 2x8 octets of data</td>
</tr>
<tr>
<td>Reachable Time</td>
<td>Retrans time (in ms) Time between retransmitted Neighbor Solicitation msgs</td>
</tr>
<tr>
<td>Time (in ms)</td>
<td>Time a node assumes a neighbor is reachable</td>
</tr>
<tr>
<td>00000000</td>
<td>00 38 41 dc 06 00 00 02 00 00 ff fe 00 5d 30 02</td>
</tr>
<tr>
<td>000010</td>
<td>00 00 ff fe 00 f7 38 7b 33 3a</td>
</tr>
<tr>
<td>000020</td>
<td>07 08 00 00 00 00 00 00 00 00 00 00 00 00 00 00</td>
</tr>
<tr>
<td>000030</td>
<td>fe 00 f7 38ff 00 00 00 00 00 00</td>
</tr>
<tr>
<td>NDP – Router Advertisement</td>
<td>86 00 29 ee 00 00</td>
</tr>
<tr>
<td>NDP – Router Advertisement</td>
<td>01 02 02 00 00 ff</td>
</tr>
</tbody>
</table>

Sent periodically after bootstrapping.
### How is NDP in IPv4 and IPv6?

<table>
<thead>
<tr>
<th>IPv4</th>
<th>IPv6</th>
</tr>
</thead>
<tbody>
<tr>
<td>Address Resolution Protocol (ARP)</td>
<td>Neighbor Discovery Protocol</td>
</tr>
<tr>
<td>· Request-Response:</td>
<td>· Part of ICMPv6 protocol</td>
</tr>
<tr>
<td>· What’s the L2 address of the host with IPv4 address X?</td>
<td>· Request-Response:</td>
</tr>
<tr>
<td>· Broadcast to all hosts</td>
<td>· What’s the L2 address of the host with IPv6 address X?</td>
</tr>
<tr>
<td>· Host with IPv4 address X replies.</td>
<td>· Multicast to all link-local hosts</td>
</tr>
<tr>
<td>· ARP announce (not mandatory)</td>
<td>· Host with IPv6 address X replies.</td>
</tr>
</tbody>
</table>

Any differences?
Address Resolution in 6loWPAN
NDP in WPAN: Broadcast problems

ETHERNET

Neighbor Solicitation: who has IPv6 address X?

Neighbor Advertisement: Me! (L2 address Y)

Who receives L2 broadcast?
NDP in WPAN: Sleeping problems

Neighbor Solicitation: who has IPv6 address X?

Neighbor Advertisement: Me! (L2 address Y)

Will a host always receive a NS?
Optimization of NDP for 6loWPAN

Neighbor Solicitation (to C): I'm node B. Please register me!

Neighbor Advertisement (to B): Sure!

Neighbor Solicitation: who has IPv6 address X?

Neighbor Advertisement: Me! (L2 address Y)

I think C (default router) should know who B is...
ICMPv6 – Neighbour Solicitation

Options:
- S2LT (01): Source link-layer address – with 2x8 octets of data

Options:
- Address Registration Option (21):
  - Status (00)
  - Registration lifetime (in minutes): The amount of time that the router should retain the NCE for the sender of the NS

NCE = Neighbour Cache Entry
ICMPv6 – Neighbour Advertisement

Options:
S2LT (01): Source link-layer address – with 2x8 octets of data

Address Registration Option (21):
00 – Status
00 of – Registration lifetime (in minutes). The amount of time that the router should retain the NCE for the sender of the NS
... EUI-64

NCE = Neighbour Cache Entry
Optimization of NDP for 6loWPAN

I think C (default router) should know who Z is...

- If I don’t know Z, either:
  - Z does not exist,
  - Z is attached to some other router,
  - Or Z is external to my 6loWPAN.
<table>
<thead>
<tr>
<th>IPv6 address</th>
<th>if L2 address</th>
<th>state</th>
<th>type</th>
</tr>
</thead>
<tbody>
<tr>
<td>fe80::ff:fe00:5d39</td>
<td>7 02:00:00:ff:fe:00:5d:39</td>
<td>REACHABLE</td>
<td>REG</td>
</tr>
</tbody>
</table>
### Neighbour Cache Entries - States

<table>
<thead>
<tr>
<th>Status</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>INCOMPLETE</td>
<td>NS was sent, waiting for NA</td>
</tr>
<tr>
<td>REACHABLE</td>
<td>Positive confirmation was received recently that the forward path works properly</td>
</tr>
<tr>
<td>STALE</td>
<td>Positive confirmation was received some time ago that the forward path works properly</td>
</tr>
<tr>
<td>DELAY</td>
<td>Positive confirmation was received long time ago that the forward path works properly</td>
</tr>
<tr>
<td>PROBE</td>
<td>A reachability confirmation is actively sought by Tx NS often until reachability confirmation is received</td>
</tr>
</tbody>
</table>

- (Tx ACK)
- (No Tx)
- (No Tx – warn higher layers)
NDP summary

Router Solicitation: I need a RA!

Router Advertisement: This is our global prefix, Context Ids and Border Router!

Neighbor Solicitation: Please register me!

Neighbor Advertisement: Ok!

I'm configured!
Routing in Low-Power and Lossy Networks
I think C (default router) should know who Z is...

- If I don’t know Z, either:
  - Z does not exist,
  - Z is attached to some other router,
  - Or Z is external to my 6loWPAN.
Traditionally - Two types of routing protocols

<table>
<thead>
<tr>
<th>Link-state Routing</th>
<th>Distance-vector Routing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Each node has a whole view of the topology graph</td>
<td>Neighbors offer information about reachability to other nodes</td>
</tr>
<tr>
<td></td>
<td>Pros:</td>
</tr>
<tr>
<td></td>
<td>• Robustness and convergence</td>
</tr>
<tr>
<td></td>
<td>Cons:</td>
</tr>
<tr>
<td></td>
<td>• Memory and amount of transmitted data</td>
</tr>
<tr>
<td></td>
<td>Cons:</td>
</tr>
<tr>
<td></td>
<td>• Convergence</td>
</tr>
</tbody>
</table>
- Destination-Oriented Directed Acyclic Graph, a graph that:
  - Has no cycles,
  - With directed arcs,
  - Where every node has a path (at least one) to a single node: the ROOT
RPL Modes of Operation

- **Upwards routes only**
- **Downwards, Non-storing (source/root routing)**
- **Downwards, Storing**
- **Downwards, Multicast**
How upward works?

DODAG Information Object (path cost to root=0)

DIO (1)

DIO (2)
How downward non-storing works?

Same as upwards, plus...

Destination Advertisement Object
(parent-child relationship)
Sent from * to root

Packets with full hop-by-hop path encoded in 6lowPAN header
How downward storing works?

Same as upwards, plus...

Destination Advertisement
Object
(parent-childs relationship)
Sent from * to parent

Aware of downward nodes (targets)

Packets with L2 address resolved at each node
How downward multicast works?
RPL Modes of Operation – Trade-offs

Upwards routes only
- root-dst
- No downward traffic!

Downwards, Non-storing (source/root routing)
- root
- Costly set-up + large packets downwards

Downwards, Storing
- root
- Memory and CPU at every node

Downwards, Multicast
- root
- ?
Two more RPL messages

DODAG Information Solicitation (DIS)

- may be used to solicit a DODAG Information Object from a RPL node
- Its use is analogous to that of a Router Solicitation as specified in IPv6 Neighbor Discovery
- a node may use DIS to probe its neighborhood for nearby DODAGs

DAO-ACK

- DAO messages can request an acknowledgement (DAO-ACK)
- A node receiving a unicast DAO message requesting acknowledgement SHOULD respond with a DAO-ACK.
- A node receiving a DAO message without requesting acknowledgement MAY respond with a DAO-ACK, especially to report an error condition.

Why shouldn’t we have an ACK message for DIO?
An important problem: How often should we send a DIO?
Trickle Algorithm
Timer!

- Defines how often should a node send updates
- Every so often, a node transmits data unless it hears a few other transmissions whose data suggest its won transmission is redundant:
  - If a node agrees, it slowly decreases the Tx periodicity (few packets per hour)
  - If a node disagrees, the node communicates quickly to resolve inconsistencies (in millis).
Ring!

Three possible scenarios:
- I heard consistent information all the time, and it was enough
- I heard consistent information all the time... but not enough
- I heard an inconsistency
If I don’t hear enough consistent information…

Is $c < k$?

- Yes: Transmit
- No: Nothing
If I hear inconsistent information...

But does not react/Tx immediately!
The algorithm

<table>
<thead>
<tr>
<th>Event</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>I expires</td>
<td>Double $I$ (up to $I_{\text{max}}$), reset $c$, pick a new $t$</td>
</tr>
<tr>
<td>$t$ expires</td>
<td>If $c &lt; k$, transmit</td>
</tr>
<tr>
<td>Receive consistent data</td>
<td>Increment $c$</td>
</tr>
<tr>
<td>Receive inconsistent data</td>
<td>Set $I$ to $I_{\text{min}}$, reset $c$, pick a new $t$</td>
</tr>
</tbody>
</table>
Issues to consider for your project

- If you opt for RPL,
  - Do you need upwards, downwards non-storing, storing, or multicast?
  - What is your metric? objective function?
Knowledge base

- About NDP:
  - Neighbor Unreachability Detection
- About RPL:
  - RPL multicast mode of operation
  - Source Header Extension for RPL/6loWPAN
  - Routing Metrics Used for Path Calculation in RPL
  - Objective Function Zero for RPL
  - The Minimum Rank with Hysteresis Objective Function
  - Reactive Discovery of P2P Routes in RPL
Open Research Problems

- Objective functions for specific purposes