Monday 2020/05/25

Virtual-interim 20200525

Routing over Low-Power And Lossy Networks

Chairs:

Dominique Barthel

Ines Robles

Secretary:

Michael Richardson

Note Well

This is a reminder of IETF policies in effect on various topics such as patents or code of conduct. It is only meant to point you in the right direction. Exceptions may apply. The IETF's patent policy and the definition of an IETF "contribution" and "participation" are set forth in BCP 79; please read it carefully.

As a reminder:

- By participating in the IETF, you agree to follow IETF processes and policies.
- If you are aware that any IETF contribution is covered by patents or patent applications that are owned or controlled by you or your sponsor, you must disclose that fact, or not participate in the discussion.
- As a participant in or attendee to any IETF activity you acknowledge that written, audio, video, and photographic records of meetings may be made public.
- Personal information that you provide to IETF will be handled in accordance with the IETF Privacy Statement.
- As a participant or attendee, you agree to work respectfully with other participants; please contact the ombudsteam (https://www.ietf.org/contact/ombudsteam/) if you have questions or concerns about this.

Definitive information is in the documents listed below and other IETF BCPs. For advice, please talk to WG chairs or ADs:

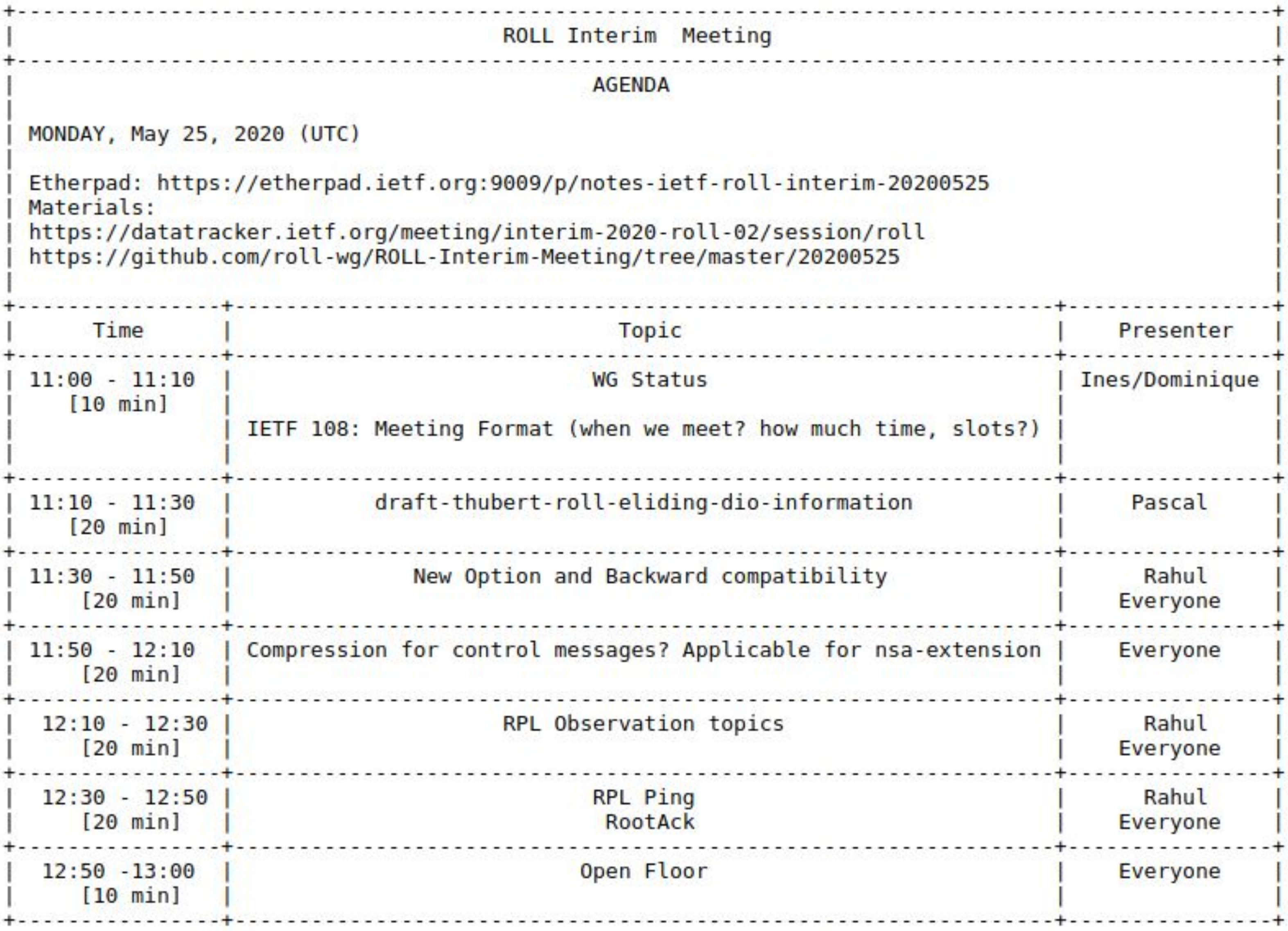
BCP 9 (Internet Standards Process)
BCP 25 (Working Group processes)
BCP 25 (Anti-Harassment Procedures)
BCP 54 (Code of Conduct)
BCP 78 (Copyright)
BCP 79 (Patents, Participation)
https://www.ietf.org/privacy-policy/ (Privacy Policy)

Source: https://www.ietf.org/about/note-well/

Meeting Materials

- Session: Monday 2020/05/25
- Remote Participation
  - Etherpad: https://etherpad.ietf.org:9009/p/notes-ietf-roll-interim-20200525
  - Slides: https://datatracker.ietf.org/meeting/interim-2020-roll-02/session/roll
  - Jabber room: roll@jabber.ietf.org
  - Minutes taker: **Please volunteer, thank you :)**
- **Please sign blue sheets = add your name into the etherpad please :-)**

Agenda



ROLL Interim Meeting

AGENDA

MONDAY, May 25, 2020 (UTC)

Etherpad: https://etherpad.ietf.org:9009/p/notes-ietf-roll-interim-20200525
Materials:
https://datatracker.ietf.org/meeting/interim-2020-roll-02/session/roll
https://github.com/roll-wg/ROLL-Interim-Meeting/tree/master/20200525

Time	Topic	Presenter
11:00 - 11:10 [10 min]	WG Status IETF 108: Meeting Format (when we meet? how much time, slots?)	Ines/Dominique
11:10 - 11:30 [20 min]	draft-thubert-roll-eliding-dio-information	Pascal
11:30 - 11:50 [20 min]	New Option and Backward compatibility	Rahul Everyone
11:50 - 12:10 [20 min]	Compression for control messages? Applicable for nsa-extension	Everyone
12:10 - 12:30 [20 min]	RPL Observation topics	Rahul Everyone
12:30 - 12:50 [20 min]	RPL Ping RootAck	Rahul Everyone
12:50 -13:00 [10 min]	Open Floor	Everyone

IETF 108

- IETF 108: Meeting format
  - **5 days**: starts Monday, 27 July and runs through Friday, 31 July
  - Approximately **5 hours of sessions** per day running from **11:00 to 16:00 UTC each day**.
  - Sessions slots are currently anticipated to be **50 or 100 minutes** in length, shorter than normal for an in-person meeting, with 20-minute breaks in between.
  - Plan to run up to **8 tracks in parallel** as we normally would at an in-person meeting.
  - **When we meet?**
    - During IETF 108?
    - How much time needed?
    - How many slots needed?
- Or we **schedule an interim meeting after IETF 108**?

Eliding and Querying RPL Information

draft-thubert-roll-eliding-dio-information

P. Thubert, D. Barthel, R.A. Jadhav

Pascal Thubert

May 25th, 2020

ROLL Virtual Interim

Changes Highlights

- No Change Since IETF 106
- Needs WG attention to progress
- So far we were really busy
  - What with NP-DAO, RUL, turnon-RFC8138, UseOfRPLInfo drafts!
- Now a good time to reboot this?
- Next To Do’s
  - Adapt to new MOPEXT/ CAPABILITIES split

What is this draft?

- The draft presents a method to safely elide a group of RPL options in a DIO message by synchronizing the state associated with each of these options between parent and child
- This is achieved using a new sequence counter in DIO messages called RPL Configuration State Sequence (RCSS)
- A child that missed a DIO message with an update of any of those protected options detects it by the change of RCSS and queries the update with a DIS Message.
- The draft also provides a method to fully elide the options in a DAO message.

Proposed method

- New RPL Configuration State Sequence (RCSS)
- Updates base objects
  - DIO to add RCSS
  - DAO to indicate it is abbreviated
  - DIS base objects to query missing options
- New “Abbreviated Option” Option (AOO)
  - Replacement for a full option, indicates last RCSS

Protected Options

The protected options are:

1. The Route Information Option (RIO) defined in section 6.7.5 of [RPL]

2. The DODAG Configuration Option (DCO) defined in section 6.7.6 of [RPL]

3. The Prefix Information Option (PIO) defined in section 6.7.10 of [RPL]

4. The Extended MOP Option (MOPex) defined in [MOPEX-CAP]

5. The Global Capabilities Option (GCO) defined in [MOPEX-CAP]

New Abbreviated Option Option

- Used as replacement of the full option
- Indicates the RCSS of the last change for this option

```
 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
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|  Option Type  | Option Length | Abbrev. opt.  | Last Mod RCSS |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 3: Abbreviated Option Option Format

Updated DIS object

- New bits to indicated requested options
- Last RCSS to which this node is synchronized

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|R|D|P[M|O| Flg | LastSync RCSS |   Option(s)...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Figure 2: Updated DIS Base Object

RCSS operation

- The RCSS applies to a DIO Message and a same value of the RCSS can be used in DIO messages that are sent consecutively with no change in the protected options.
- The RCSS is incremented by the Root using a lollipop technique
- A reboot of the Root is detected when the RCSS moves from the circular to the straight part of the lollipop.
- During the straight part of the lollipop, a second reboot of the Root might not be recognized. For that reason the protected options MUST be provided in full with each increment on the RCSS during the straight part of the lollipop.
- When a field is modified in one of the protected options, the Root MUST send a DIO with an incremented RCSS and the modified protected option(s) in full.

Resync operation

A child can resynchronize any of the protected options to the latest RCSS by sending a DIS Message to a candidate parent that advertises that RCSS in DIO messages.

The child MUST set the desired combination of 'R', 'D', 'P', 'M’ and 'O' flags to indicate the option(s) that it needs updated.

The child MUST signal in the Last Synchronized RCSS field of the DIS the freshest value of RCSS for which it was fully synchronized

The DIO message that is sent in response MUST contain in full all the options that are requested and that were updated since the Last Synchronized RCSS in the DIS Message. The other options MUST be added in the abbreviated form.

The options MAY be spread over more than one DIO message sent in a quick sequence.

Backward Compatible New Options

- ROLL Interim (25th May 2020)

What is the problem currently?

- (Almost every) New RPL extension requires sending new RPL Control Option
- Node which does not understand this new option may strip it off
  - Node may or may not strip, RFC 6550 does not explicitly state the handling
- Example
  - Enrollment Priority
    - Min priority field needs to be copied downstream even if not understood by node
  - Eliding-RPL-Info, AOO Option
    - Does not depend on copying the option downstream
    - However needs to explicitly strip off the option if not understood

Expectation

- There should be a way for the node to
  - Copy the option forward even though it doesn’t understand
  - Or Explicitly Drop the option (not message) if it doesn’t understand
    - This is what “may” happen currently
- There are more possibilities (just like cap flags)
  - Drop/Discard the message if option not understood
  - Join as 6LN if option not understood:w

Proposition-1

- Use second higher order bit of Option Type to indicate ‘C’ copy-if-not-understood flag
  - First high order bit is already used as ‘Secure’ flag
- Easy to handle, no change in control overhead
  - However cannot handle anymore flags (such as Join/Discard)

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+- - - - - - -
|S|C| Option Type | Option Length | Option Data
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+- - - - - - -
```

Proposition-2

- Use second higher order bit of Option Type to indicate ‘X’ extended Option flag
    - Option data starts after flags
- Allows to have more flags
    - Can satisfy requirements for other flags (Join and Discard flag)
- Option Length includes Option Flags and above
    - Thus making it backward compatible such that Option can be skipped while traversing

```
 0                   1                   2
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-  - - - - - -
|S|X| Option Type | Option Length |Option Flags | Option Data
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-  - - - - - -
```

Next Step

- WG to decide
- Add this to mopex draft (not cap draft)
  - Make the unknown-option-handling explicit
  - Make it mandatory for RPLv2 nodes to understand the new flag(s)

Compression RPL Control Messages

- Related work:
  - A Compression Format for RPL Control Messages
    - https://tools.ietf.org/html/draft-goyal-roll-rpl-compression-00
  - RFC7400: 6LoWPAN-GHC: Generic Header Compression for IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs)

- Would the Control-Message-Compression mechanism affect nsa-extension draft or this latter can proceed further?

RPL Observations

- ROLL Interim (25th May 2020)

Current Status

Sect	Point	Status	Possible Next step?
3	Handling DTSN increment in storing MOP	No-progress Problem stmt discussed few times in last two years	BCP? How to handle sub-dodag route update in storing MOP? When to update DTSN in storing MOP?
4	DAO-ACK Handling	Draft submitted in IETF107	
5	Interpreting Trickle Reset	Clarification in existing BCP?	May not warrant a new draft.
6	Handling resource unavailability	Enrollment priority handles this	
7	Handling aggregated targets	IETF107 discussion for compressing address vectors	
8	TIO handling	BCP?	
9	Upgrades to RPL	Backward compatibility options handles it partly	

Current Status - contd

Sect	Point	Status	Possible Next step?
10	Path Control Bits handling	Problem stmt discussed in IETF107	Errata?
11	Adjacency probing with RPL	DIS-use-cases	WG-adoption
12	Control Options eliding mechanism	Eliding-RPL-Info	WG-adoption
13	Eliminating need of persistent storage	No-progress. Problem stmt discussed several times	Need solution. Possibly standards track.
14	Capabilities	cap-draft	In progress
15	RPL underspecification	Few points where 6550 is not clear	Errata?

RootACK

- Roll Interim (25th May 2020)

Motivation for Storing-Root-ACK

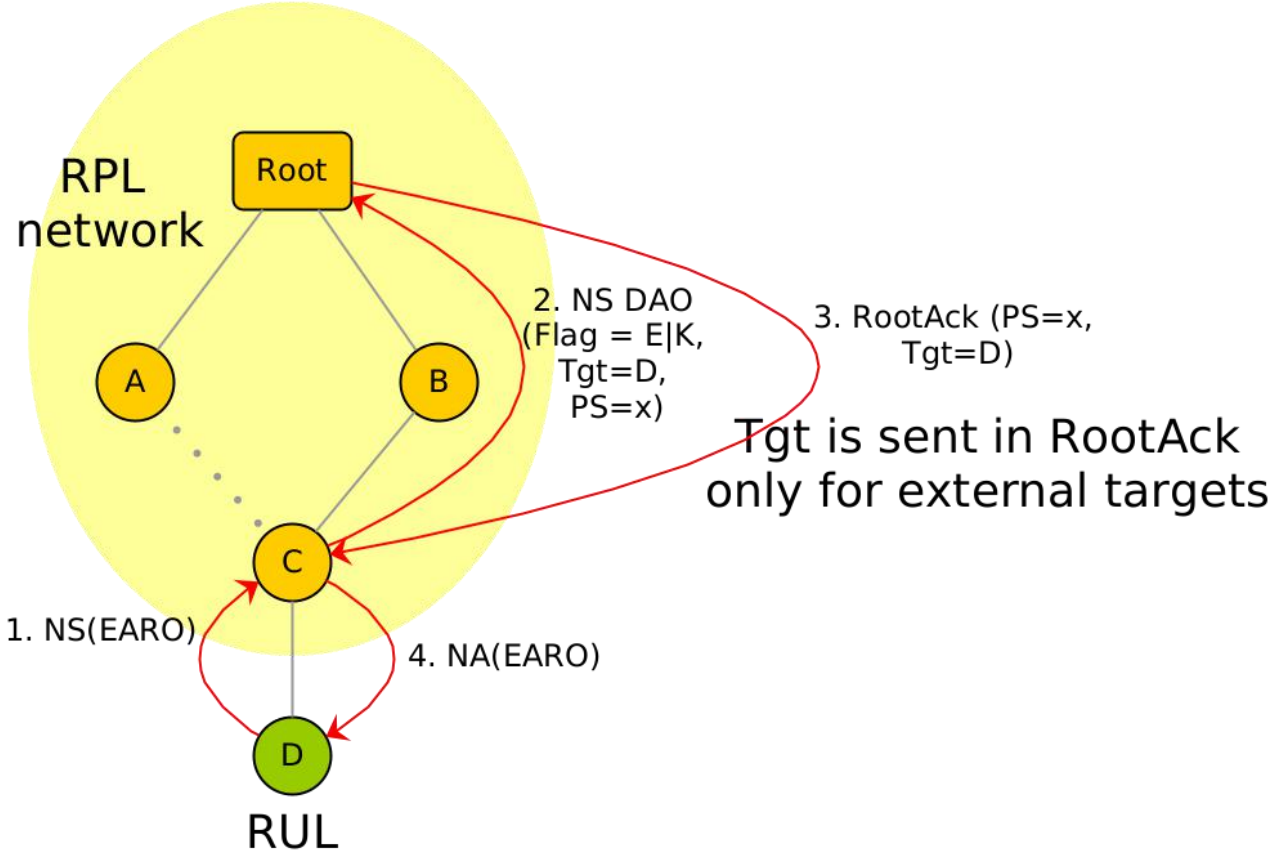
- End to end path establishment indication
  - Node can initiate app traffic
- Query target node’s capabilities

K flag in TIO

- K flag is set by the target node in the TIO
  - used by the root to send the RootAck
  - Ideally the node may set the K flag only once after startup
  - RootAck may be sent asyncly by the root without any K flag too
    - useful for cap query
- Intermediate 6LRs: K flag handling
  - DAO is regenerated on 6LRs on behalf of target node
  - K flag has to be stored in context to the target. Similar to E flag.
  - When the intermediate nodes see the K flag disabled from the target the K flag could be reset

RULs with RootACK

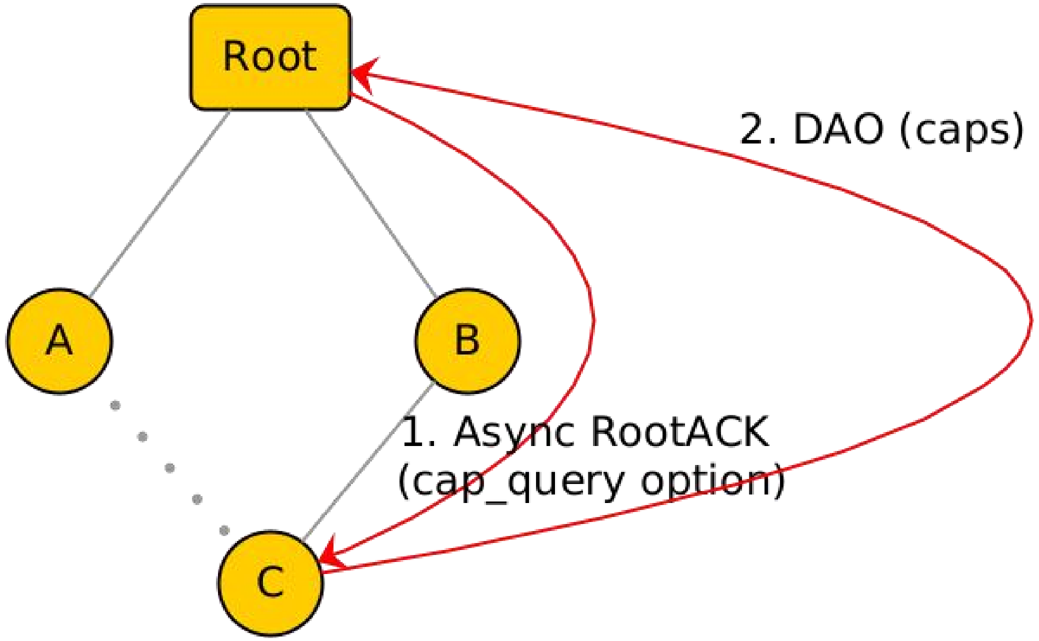
- Send NA to RUL only when E2E path is established
  - Send NA in response to RootACK
- Note that for external targets DAO is directly addressed to Root even in storing MOP
  - Implies source routed data path for external targets
  - Similar to unaware-leaves
- For external targets, the RootACK needs to carry the Target Option



Tgt is sent in RootAck only for external targets

Handling Capability Query

- Root sends async RootACK to query caps
  - Capability Query Option needed
    - Already pending work item in cap draft
- Target responds with DAO with caps
  - No Target/TIO Option needed, since DAO addressed directly to Root



Open Floor

When we meet next time?