27 May 2016 Webex

IPv6 over the TSCH mode of IEEE 802.15.4e

Chairs:
Pascal Thubert
Thomas Watteyne

Etherpad for minutes:
http://etherpad.tools.ietf.org:9000/p/6tisch?useMonospaceFont=true

Note Well

This summary is only meant to point you in the right direction, and doesn't have all the nuances. The IETF's IPR Policy is set forth in BCP 79; please read it carefully.

The brief summary:

- By participating with the IETF, you agree to follow IETF processes.
- If you are aware that a contribution of yours (something you write, say, or discuss in any IETF context) is covered by patents or patent applications, you need to disclose that fact.
- You understand that meetings might be recorded, broadcast, and publicly archived.

For further information, talk to a chair, ask an Area Director, or review the following:

- BCP 9 (on the Internet Standards Process)
- BCP 25 (on the Working Group processes)
- BCP 78 (on the IETF Trust)
- BCP 79 (on Intellectual Property Rights in the IETF)

Reminder:

Minutes are taken *
This meeting is recorded **
Presence is logged ***

* Scribe; please contribute online to the minutes at http://etherpad.tools.ietf.org:9000/p/6tisch?useMonospaceFont=true
** Recordings and Minutes are public and may be subject to discovery in the event of litigation.
*** From the Webex login

Agenda

- Administrivia [3min]
  - Agenda bashing
  - Approval minutes from last meeting
- Update Security Discussions [5min]
- 6P error code for indicating schedule/cells are locked [20min]
- interaction between 6top and SF [20min]
- AOB [2min]

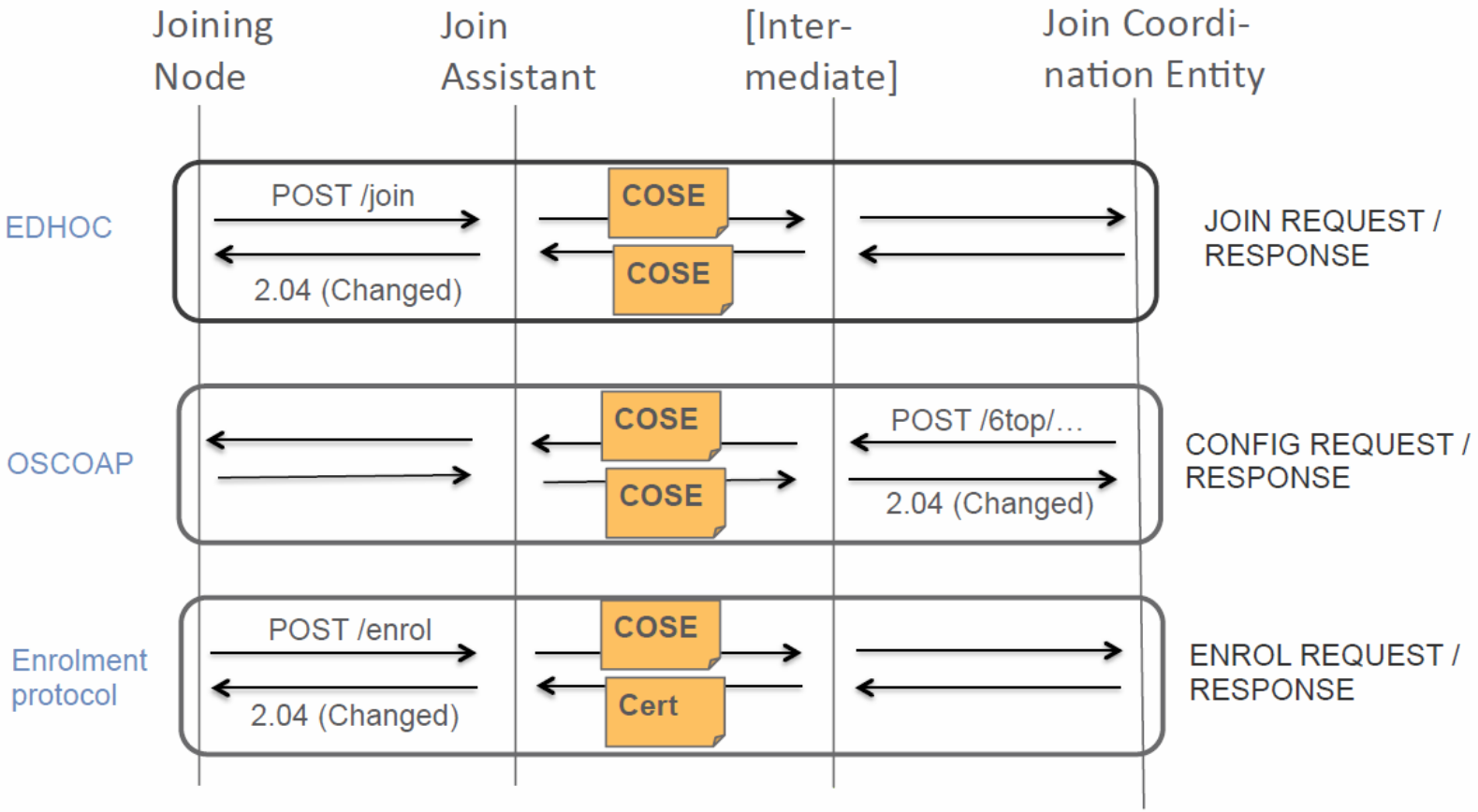
Administrivia

Admin is trivia

- Approval Agenda
- Approval minutes

Update Security Discussions

Securing 6tisch with COSE



› Key establishment, Join, Configuration, Certificate enrolment, and P2P security could all be embedded in CoAP and COSE

6P error code for indicating schedule/cells are locked

OLD

A node MAY support concurrent 6P Transactions from different neighbors. In this case, in Figure 1, node C can have a different ongoing 6P Transaction with nodes B and E. In case a node does not have enough resources to handle concurrent 6P Transactions from different neighbors, when it receives a 6P Request from a neighbor while already handling a different request from a different neighbor, it MUST reply to that second request with a 6P Response with return code IANA_6TOP_RC_BUSY.

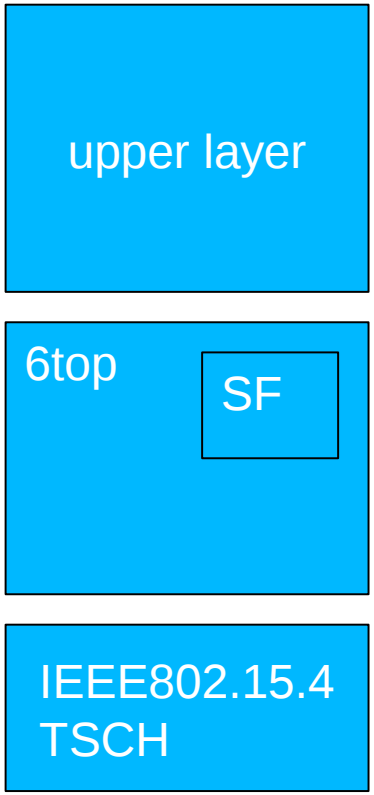
NEW

A node MAY support concurrent 6P Transactions from different neighbors. In this case, in Figure 1, node C can have a different ongoing 6P Transaction with nodes B and E. In case a node does not have enough resources to handle concurrent 6P Transactions from different neighbors, when it receives a 6P Request from a neighbor while already handling a different request from a different neighbor, it MUST reply to that second request with a 6P Response with return code IANA_6TOP_RC_BUSY. In case concurrent transactions are supported, the protocol needs to guarantee atomicity.
If concurrent transactions are supported, during a particular transaction, the cells involved in that transaction MUST be locked until the transaction finishes. When transaction finishes, those cells are either allocated, or released. If a node receives a transaction while another transaction is ongoing, and that subsequent transaction involved the same cells, the mote MUST reply to that second request with a 6P Response with return code IANA_6TOP_RC_BUSY.

OPEN

use metadata to distinguish between unsupported concurrent transactions and no cells locked

interaction between 6top and SF



- upper layer
  - sends/receives frames with neighbor node
- SF
  - MUST specify the rule for a Transaction source to select cells to add to the CellList field in the 6P ADD Request.
  - MUST specify the rule for a Transaction destination to select cells from CellList to add to its schedule.
  - MUST specify the rule for a Transaction source to select cells to add to the CellList field in the 6P ADD Request.
  - MUST specify the rule for a Transaction destination to select cells from CellList to add to its schedule.
  - MUST specify what to do after an error has occurred (either the node sent a 6P Response with an error code, or received one).
- 6top
  - terminates the 6P
- TSCH
  - executes the schedule

Requirements for an SF

https://tools.ietf.org/html/draft-ietf-6tisch-6top-protocol-00#section-5.2

1.MUST specify an identifier for that SF.
2.MUST specify the rule for a node to decide when to add/delete one or more cells to a neighbor.
3.MUST specify the rule for a Transaction source to select cells to add to the CellList field in the 6P ADD Request.
4.MUST specify the rule for a Transaction destination to select cells from CellList to add to its schedule.
5.MUST specify a value for the 6P Timeout, or a rule/equation to calculate it.
6.MUST specify a meaning for the "Metadata" field in the 6P ADD Request.
7.MUST specify the behavior of a node when it boots.
8.MUST specify what to do after an error has occurred (either the node sent a 6P Response with an error code, or received one).
9.MUST specify the list of statistics to gather. An example statistic if the number of transmitted frames to each neighbor. In case the SF requires no statistics to be gathered, the specific of the SF MUST explicitly state so.
10.SHOULD clearly state the application domain the SF is created for.
11.SHOULD contain examples which highlight normal and error scenarios.
12.SHOULD contain a list of current implementations, at least during the I-D state of the document, per [RFC6982].
13.SHOULD contain a performance evaluation of the scheme, possibly through references to external documents.
14.MAY redefine the format of the CellList field.

Layers

Upper: manage bandwidth in abstract units

Middle: map units to cells

Lower: update schedule

AOB ?

Thank you!