

802.1AX -- Link Aggregation:

Editor's Report: November 2018

Version 1

Stephen Haddock
November 13, 2018

802.1AX status

- AX-Rev-d0.4 went to Task Group ballot in October 17 through November 2, 2018.
 - Significant changes to clause 3 Definitions and informative text in 6.6.1-6.6.3 and 9.1-9.4.
 - Changed Distributed Relay description from “component model” to “sublayer model”
 - Reconciled clause 7 Management with current variable definitions.
 - PICS and MIB clauses (A, D) mostly untouched.
- Ballot results:
 - 5 yes; 1 no; 8 abstain.
 - 130 comments from 5 commenters.

Easy comments

- Very straightforward “Proposed Accept” or “Proposed Accept in Principle”:
1, 13, 14, 15, 17, 18, 19, 21, 22, 23, 24, 25, 28, 29, 31, 32, 34, 37, 38,
39, 41, 42, 45, 47, 50, 51, 52, 53, 54, 55, 56, 57, 58, 59, 61, 64, 65,
67, 70, 72, 74, 85, 86, 87, 91, 92, 93, 94, 95, 100, 101, 105, 106,
108, 109, 110, 111, 113, 114, 115, 118, 121
- “Proposed Accept in Principle” where editor doesn’t think the alternative resolution would be controversial:
4, 5, 7, 12, 30, 33, 35, 36, 43, 44, 46, 66, 73, 77, 78, 79, 80, 103, 120, 129
And 48, 51, 54, 90, 98, 104, 107, 112 (not included in email sent last Sunday)
- Hoping to resolve off-line:
20, 26, 27, 49, 71, 76, 88, 119
- Plan to change all of these to final mode when all other comments have been resolved. Do not plan to discuss unless someone explicitly asks for them to be discussed.

Hot topics

- Link Numbers and Port Numbers: 11, 96, 123, 124, 125
- Configuration and Misconfiguration: 99, 126
- Service ID map: 122
- CSCD diagram: 6, 130
- Management questions: 40, 89, 97, 116, 117, 128
- Parser/Multiplexer: 2, 3
- Organization: 9, 68
- Marker Protocol: 8
- Terminology and Definitions: 10, 16, 60, 62, 63, 127
- Distribution algorithms: 69, 75
- DRCP MAC Address: 76
- TLV Structure: 102
- Individual Links: 81, 82, 83, 84
- Aggregator Enabled: 88

Plan going forward:

- Complete draft 1.0 for Working Group ballot
 - Implement resolution of d0.4 comments
 - Change bars?
 - Update MIB
 - Deprecate/obsolete old DRNI stuff and create new
 - Update PICS
- YANG ??

First Hot Topic: Link_Numbers

- General idea:
 - A Link_Number is an identifier of a link in a LAG. Actor and Partner each assign an Admin_Link_Number to each AggPort.
 - When both Actor and Partner are using the same CSCD parameters (distribution algorithm, Service ID map digest, Link map digest), they use the Admin_Link_Numbers of the system with the lowest System ID as the operational Link_Numbers. This results in both systems using the same link for each Conversation ID.
 - Otherwise the Actor and Partner each use their own Admin_Link_Numbers as the operational Link_Numbers.

Admin_Link_Numbers

- Constraints on Admin_Link_Number assignment
 1. Each AggPort's Admin_Link_Number is (should be) unique among all AggPorts with the same Key value.
 - Otherwise could get more than one link with the same Link_Number in the same LAG. Each system would use only one of the links, and not necessarily the same link. This underutilizes the links and, if Discard_Wrong_Conversation is TRUE, could result in the loss of all frames distributed to the link.
 - This is typical consequence of violating most configuration constraints, e.g. "any given Service ID value can appear in at most one entry in the [Service ID map]"
 2. Each Admin_Link_Number is (should be) in the set of Link_Numbers in the Admin_Conversation_Link_Maps of any Aggregators with the same Key value.
 - Otherwise no frames will be distributed to the link.
- Per comment #124, these constraints will be incorporated into the Admin_Link_Number definition in clause 6.6.3.2 and 7.3.2.
- Comment #126 suggests documenting the consequences of misconfiguration. Where?

Link_Numbers and DRNI


- Additional constraint on Admin_Link_Number assignment for DRNI:
 - Each AggPort's Admin_Link_Number is (should be) unique among all DRNI AggPorts associated with the DR-sublayer of both DRNI Systems.
 - Otherwise, in addition to the consequences of a duplicate link number within a LAG at a single Aggregation System, frames with a given Port Conversation ID could be distributed to different links.
 - Per comment #11, this constraint needs to be documented somewhere.
 - Comment #99 suggests having a subsection early in clause 9 with all configuration requirements for DRNI. Where should it go?

Alternative resolutions to duplicate Link_Numbers

1. Rather than accept the underutilization of links and potential frame loss resulting from a misconfiguration that causes duplicate Link_Numbers on a LAG, we could try to have the protocol resolve the conflict:
 - Among the links with the same Link_Number, only use the link that lowest Port ID on the system with the lowest System ID.
 - Prevents the potential frame loss, but only uses one link so still underutilizes the links.
 - Could be extended to DRNI, but gets a bit more complicated.
 - Pushes the problem to situation where there are duplicate Link_Numbers and duplicate Port IDs.
 - Port IDs are supposed to be unique within a system, but there are generally no ill effects in Link Aggregation to having duplicate Port IDs.
 - Editor recommends **NOT** doing this.
 - A lot of complexity for little benefit.
2. Set DWC FALSE when have duplicate link numbers
 - Doesn't resolve underutilization, but prevents frame loss.


Service ID Map

- 802.1AX-2014
 - Supports a Port Algorithm both using the Service ID map and not using the Service ID map, but there is no standard way to configure whether or not the map is used.
- 802.1AX-Rev-d0.4
 - Added a RW object for the Service ID Map Digest. Forcing this to zero indicates that the map is not used. Kind of a kludge.
 - Overloads the value zero (since the digest of the map might come out zero).
 - Awkward to have object that is RW but also updated as a side-effect of writing to the map itself.
 - Alternative is to add a separate Boolean to indicate if map is used.
 - Actually need aggActorAdmin, aggPortActorOper, aggPortPartnerOper, and aggPartnerOper versions of this Boolean.
 - Change digest object to RO. Don't need to deprecate existing object since it is a new object that hasn't been published in a MIB yet.



Comment #122 suggests changing this to read-only

Moving Conversations

- One of the two primary reasons for Discard_Wrong_Conversation is to allow moving conversations between links without the Marker protocol
 - During transient while Actor and Partner move the conversation, frames are discarded rather than mis-ordered.
 - Targeted at the transient that occurs when a new link joins the LAG.
 - Changing the Link_Number of an active link is effectively the same as a new link joining the LAG.  Relevant to Comment #126
 - Can be used to move some conversations moving to that link and some away.
 - Changing the Admin_Conversation_Link_Map moves conversations, but with a “long transient” while the map updated on both machines.
 - Currently turn off Discard_Wrong_Conversation to avoid pro-longed frame loss on the affected conversation, but also removes DWC enforcement on other conversations.
 - May want to consider adding an “AUTO-STRICT” value for Admin_DWC to keep Discard_Wrong_Conversation on during the “long transient”. Maintains DWC enforcement on other conversations.

Back up slides

YANG

- 802.1AX YANG status:
 - Someone (Marc?) has created a module on github based on 802.1AX-2008.
 - Mick has created a UML that includes the new Conversation-Sensitive Collection and Distribution (CSCD) parameters, but not Distributed Relay/DRNI parameters.
- Have previously concluded that adding a YANG annex would be in scope, but would be done as a follow-up project if no YANG module is contributed.
 - Don't want to include a module that does not match this revision, including CSCD and DRNI.
 - Draft 0.4 should be sufficiently complete to use to develop the module.
 - **If anyone is interested in contributing a YANG module, now is the time!**

Thank You