802.1AXdz
YANG for Link Aggregation:

Editor's Report: March 2025

Version 1

Stephen Haddock

March 10, 2025

802.1AXdz status

- AXdz-d0.2 went to Working Group recirculation ballot on February 10, 2025 through February 25, 2025.

- Cumulative WG ballot results:
  - 23 yes; 0 no; 22 abstain (ballot passes).
  - 23 comments from 6 commenters.

- Preliminary disposition:
  - 802-1AXdz-d1-1-pdis-v01.pdf

Discussion Topics

- Easy ER comments: 5, 7, 8, 9
- T and TR comments: 1-4, 10, 11, 15
- Text string patterns: 23, 13, 14
- Actor-admin-state in key-group: 17
- Remaining E comments: 6, 12, 16, 18-22

Comment #23

- The current pattern constrains the string to have valid values as much as possible, but is more processing intensive than a simpler pattern.
  - Current pattern limits individual values and values at the beginning or end a range to 0-4095, but the increment value is loosely constrained.
    - This could be solved if we could restrict the increment to be less than the end value minus the beginning value of the range, but there is not an obvious way to put this restriction in the pattern.
  - When is the amount of processing an issue?
    - Does validity checking of the values after validating pattern conformance require more or less processing?
    - If we choose the simpler pattern, why not simplify further and allow leading zeroes?
- Note that ieee802-1q-types has a vid-range-type that is very similar to the proposed simplified pattern.
  - We cannot simply use the vid-range-type because it does not allow 0 which is a valid CID value, and it does not have the increment value which makes specifying all-even and all-odd values difficult.

Comment #17: actor-admin-state

- Comment asks for further explanation of:
  1. The reasoning behind the resolution to comment #36 on draft 1.0 that moved the actor-admin-state leaf from the aggport container to the key-group container.
  2. Are there any use cases for having differing configuration values of actor-admin-state on the Aggregation Ports attached to a single Aggregator (i.e. in a Link Aggregation Group)?

Comment #17 (continued)

1. Why put actor-admin-state in the key-group container?
   - Link Aggregation has a large number of parameters specified per-port or per-aggregator (or both) that are always the same among ports that can potentially form a LAG. This leads to needless repetition of configuration and status values.
   - The key-group container collects such parameters in order to:
     1. Reduce the size of configuration and data models by eliminating a large number of repetitive values.
     2. Simplify configuration.
     3. Reduce the probability of unexpected behavior due to inconsistent configuration.
   - Parameters are put into the key-group container if:
     1. They must be the same among ports that can potentially form a LAG.
        e.g. agg-system, actor-admin-key, actor-protocol-da, cscd parameters
     2. There is no compelling reason to use different values on different ports on the same station in the same LAG.
        e.g. partner-admin-system, partner-admin-system-priority

Comment #17 (continued)

2. There is no compelling reason to have different values of the three configuration bits in actor-admin-state on Aggregation Ports in the same system in the same LAG.
   a) aggregation: This value must be set (TRUE) to allow a LAG to form, so all the bit is necessarily the same for Aggregation Ports in a LAG.
   b) lacp-activity: Controls whether an Aggregation Port will always transmit LACPDUs (active), or only “speaks when spoken to” (passive). This value should always be set (TRUE) when the aggregation bit is set, to avoid the case where a LAG cannot form because both actor and partner are passive.

      It could be argued that topology considerations might justify setting one station to passive when the other is active (such as a passive end station forming a LAG with an active bridge), however it is difficult to come up with any situation where it would be desired to have both active and passive ports on the same station in the same LAG.

Comment #17 (continued)

c) lacp-timeout: Tells the LACP partner whether to transmit periodic LACPDUs frequently (1s interval) or infrequently (30s interval). LACPDUs will be sent as needed, regardless of lacp-timeout setting, when there is a change of state in either station. Periodic transmissions have two purposes. First, they provide a retry mechanism in case an LACPCU is lost. Second, they allow a station to detect lost connectivity to its partner. This value is typically TRUE when aggregation is TRUE to allow rapid detection of lost connectivity, and FALSE when aggregation is FALSE to reduce bandwidth consumption and processing when no LAG is going to be formed.

There may be situations where rapid detection of a lost link in a LAG is not required, and lacp-timeout is set to FALSE to reduce processing, but this argument applies to all links in the LAG. It is difficult to come up with any situation where it is essential that some links in a LAG use infrequent LACPDU transmission while the others use frequent transmission.

Next steps

1. Working Group recirculation
2. Conditional approval for SA ballot

Back up slides

MD5 Digests

- 16 octets in length
- Hex string has 32 characters
  - ietf-yang-types:hex-string type
    - separates octets with ":" which increases size to 47 characters
    - accepts upper or lower case for a-f|A-F, but says lower case is canonical
  - NULL = zero = 00:00:00:00:00:00:00:00:00:00:00:00:00:00:00:00
- Base64 has 24 characters
  - NULL = zero = AAAAAAAAAAAAAAAAAAAAAA==

4096 bit masks

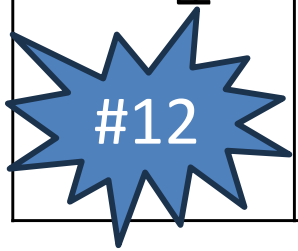
- 512 octets in length
  - Think of it as a bit per VLAN-ID
- Hex string has 1024 characters
  - ietf-yang-types:hex-string type
    - separates octets with “:” which increases size to 1535 characters
    - accepts upper or lower case, but says lower case is canonical
  - Define a new hex string pattern?
    - E.g. “[0-9a-fA-F]{1024}” or “[0-9a-fA-F]{32}(-[0-9a-fA-F]{32}){31}”
- Base64 has 684 characters
  - Default (for masks that are config true) is all ones, which encodes to 680 “/” characters followed by “//8=“:
    - //////// … //8=

Making the masks more manageable

- The drni-mask grouping provides three choices for specifying the mask:
  1. Pattern: pre-defined bit pattern
     - Examples: “all-ones”, “even-odd”
     - Can be augmented with new identity statements
  2. CID-list: list of CIDs with a “1” in the mask
     - Good when the mask is sparsely filled
  3. Mask: specify the full 4096 bit mask
     - Worst case, but covers all possibilities

Effect of Activity flags in the admin port states when Sonny connects to Cher

Sonny / Cher		Actor_Admin.Activity : Partner_Admin.Activity			
		1 : 1	1 : 0	0 : 0	0 : 1
Actor_Admin.Activity : Partner_Admin.Activity	1 : 1	Both sing (Both transmit LACPDUs)	Both sing	Cher sings. Sonny sings when hears Cher.	Both sing
	1 : 0	Both sing	Both sing	Cher sings. Sonny sings when hears Cher.	Both sing
	0 : 0	Sonny sings. Cher sings when hears Sonny.	Sonny sings. Cher sings when hears Sonny.	Neither sing	Sonny sings
	0 : 1	Both sing	Both sing	Cher sings	Both sing, then both stop when hear the other. Timeout and repeat.



Thank You