Discovery and Association in IEEE 802.1CS Link-local Registration Protocol

Norman Finn
Huawei Technologies Co. Ltd
cs-finn-discovery-association-0118-v02

P802.1CS Next-rev
Creating LRP Instances and Portals

- After processing the comments, here are the editor's suggestions for resolving Issues 4.1, 4.2, and 4.3, Discovery and Connection Management, described in 802-1CS-d1-2-pdis-v1.pdf.
- **This presentation will be updated to cs-finn-discovery-association-0118-v03 to reflect the resolution of the Task Group ballot comments on P802.1CS D1.2.**

Task group responses recorded in blue.

LLDP and LRP

- LLDP can run several instances using different destination MAC addresses, to distinguish between a Two-Port MAC Relay, a Provider Bridge, and a Customer Bridge all reachable on the same point-to-point link.
- This is another way of describing a “shared medium”.
- **Question #1: Is running one application on one instance of LRP with the Provider Bridge, and another application on another instance of LRP with the remote Customer Bridge in the scope of LRP or not?**
- **Editor’s answer: Not in scope.**

Task Group answer:

LLDP and LRP

- **Proposal #1: The LDP TLV(s) are only allowed to use the "nearest bridge" instance of LLDP (01-80-C2-00-00-0E).**
  - This is the one that does not pass though an 802.1-defined device.
  - This is the original 802.1AB-2005 choice, and the one most commonly implemented.
  - This is the least likely address to be forwarded, and thus see a real or simulated shared medium.
  - (new v2) LRP has a per-port configuration of what LLDP instance to use, and the default is -0E. LRP TLV matches are possible on only that LLDP instance.

Task group response:

ECP Issue (added to v2)

- In IEEE Std 802.1Q-2014, ECP is described only in the context of an Edge Virtual Bridge.
  - It appears to this author that the MIBs allow one to create a single instance of ECP on an ordinary Bridge Port.
  - But, this is not supported by the descriptions of the managed objects in Clause 12. (See, for example, 12.26 EVB management.)
- The Upper Layer Protocol (ULP) determines the ECP destination MAC address. But, there is no provision to have difference sequence numbers for different remote addresses or different ULPs.
- There is no indication in 802.1Q of what would happen if two different ULPs on the same Bridge Port want to use two different destination MAC addresses for ECP.

LLDP, LRP, and ECP

- Only one instance of ECP can be instantiated on a port, using one destination MAC address, which can be any one of the 16 reserved addresses or can be a unicast address.
- The ECP destination MAC address is chosen by “the application”. There can be more than one application, and more than one LLDP instance. **What address to choose?**

LLDP, LRP, and ECP

- **Proposal #2: We add a single MAC address to the LRP ECP TLV that says, "use this destination MAC address with ECP to reach me."**
  - An application specification could say what address that application wants.
  - Ultimately, the network administrator has the responsibility for reconciling the needs of the applications using ECP and the varying reaches of LLDP. I would imagine that the MAC address would usually be a unicast address, but it might be an LLDP address.
  - ~~We need to check to see if IEEE 802.1Q has a variable for specifying the ECP destination address, and add a maintenance item, if not. (My copy of 802.1Q is not searchable.)~~
  - The source MAC address is always the port's unicast MAC address.

Task group response:

LLDP, LRP, and TCP

- The present (D1.2) LLDP LRP TCP TLV allows a different IP address for each appID.
- We can consider this overkill, or we can consider this a feature that makes it easy to punt control of different applications to different controllers and/or bridges.
- But, the latter would be equivalent to making point-to-point connections over a shared medium.
- One connection per application would eliminate application multiplexing over TCP, but is difficult for ECP (802.1Q issues).
- So, let's just say, "No!"

LRP Discovery

- **Proposal #3: Have only one TCP address in the LLDP LRP TCP TLV.**
  - There is at most one TCP connection for LRP per port.
  - All applications must share that connection.
  - The AppId is sufficient for any LRPDU other than a Hello to differentiate the databases.
  - A bridge cannot use TCP to punt the processing of LRP to a remote controller on a per-application basis, but only all-applications-or-none.

Task group response:

ECP, TCP, or both?

Question 2: How many LRP instances (ends of LRP-DT transport connections) do we support? 1 or 2?

Sub-questions:

a) Can the choice of using ECP or TCP for a given application be arbitrary per-system? (What if I pick ECP-only and you pick TCP-only?)

b) Can an application specification make an ECP-only or TCP-only choice?

- **Note that ECP and TCP are different – ECP silently fails after *N* retries.**

ECP vs. TCP: Pick one

1. ECP must be implemented. TCP is optional, but a given application specification can require TCP. If both are present, TCP is used.
2. TCP is required. Remove ECP from the document.
3. ECP is required. Remove TCP from the document.
   - All three choices lead to having only one LRP instance per port.
   - The editor suggests that other combinations are not viable. “TCP required, ECP optional” doesn’t make sense, because TCP is more capable. “Implementation choice” doesn’t make sense, because an ECP-only system cannot talk to a TCP-only system.
   - ~~TCP is, in the editor’s opinion, the likely solution for shared media.~~
   - (new v2) An 802.1Q amendment is required to use ECP on shared media.

Task group response:

Matching appIDs to ECP/TCP choice

- **Proposal #4: Have only one LLDP LRP TLV that has (depending on the ECP vs. TCP choice):**
  - **0 or 1 ECP destination MAC address**
  - **0 or 1 IP address for TCP connections**
  - **A list of appIDs (only if TCP is supported)**
- ECP connections do not require the transmission of a frame. So, the only reason for including a list of appIDs is to avoid creating a TCP connection when the ends have no applications in common. So, if we drop TCP, we can drop the appID list.

Task group response:

Shared media

- If the above proposals are accepted, then the shared medium question, Issue 4.1, is easily resolved:
- **Proposal #5: If the nearest-bridge instance of LLDP database has more than one neighbor offering an LDP TLV, then an error is reported up the stack to the applications.**
  - No TCP connection is made, if not already established.
  - If a TCP connection is established, then one final Hello is sent to report the error.
  - Existing portals are destroyed and the TCP connection is terminated.

Task group response:

Discovery vs. Connection Management

- The above proposals simplify the relationship between discovery and connection management
- **Proposal #6: LRP Discovery uses LLDP to decide how to set up LRP-DT. Connection management uses Hello LRPDUs to maintain connections between applications.**
  - Connection management failures do not affect the LRP-DT connections; they remain even if no LRP application connection (association) can be made. (You could get into a loop making and breaking TCP connections.)
  - LRPDUs are not restricted to those mentioned in the LLDP TLVs. This is an unnecessary error detection mechanism.

Task group response:

Multiplexing over LRP-DT (added to v2)

- What is needed to differentiate the Portals using one LRP instance?
- **Proposal #6.5: There can be only one instance of a given application over one LRP instance (LRP-DT connection).**
  - If one application wants to run multiple instanced, it supplies its own mux point.
  - The cost would be additional discovery mechanisms, in order to associate the right database on this system to the right database on the neighbor system.

Task group response:

Connection management

- **Proposal #7: change the name of “connection management” to “Portal association management.”**
  - The tem “connection management” leads to confusion with TCP connections.
- If the above proposals are agreed to, then an **LRP-DT Portal** is created using **Hello LRPDUs**.
  - Only the Hello needs carry MySysId, MyPortId, YourSysId, YourPortId.
  - All other messages (except End) carry only an appID.
- Information passed via LRP-DT is ignored unless/until the Hellos are exchanged.

Task group response:

Thank you