802.1ap Port Creation

Ken Young

Jan 28, 2008

Version 3

Conclusions from 802.1ah v4.0

- Based on the resolution of comment 57, BEB ports can be created and deleted
- During the discussion, a proposal for a general mechanism for dealing with port creation was discussed.
  - Apply to all port types
  - Apply to logical applications of a MAC relay
    - i.e., “These MIBs must apply to VPLS”
- What follows is a proposal to implement the functions now described in clause 12.

Requirements: port creation must…

1. Be applicable to all port types
2. Be forward compatible
   - New port type could be created without touching the current tables
3. Leverage existing SNMP constructs
   - Clear which table is used to create a port of a particular type
   - Possible to create a port using just a RowStatus
4. Be optional
   - Does not make sense for all systems
   - A 1D or a 1Q bridge which has only physical Ethernet ports does not require this mechanism.

Requirements: port creation must…(cont)

5. Be Explicit
   - Allow for addition or removal of a port over time
   - For example, add a VIP to an I-Component and associate it to an existing I-SID
6. Be technology agnostic
   - A Base Bridge Port has an attribute which points to the interface which the state is associated.
   - The interface (identified by ifIndex) can be of ifType
     - bridge (209), ethernetCsmacd (6), frDlciEndPt (193), atmVciEndPt (194), etc.
     - Whatever else is imagined in the future…

Port creation proposal

- Follow the ifTable methodology developed in the IETF
  - Certainly forward compatible
    (has stood the test of time with many new interface types)
  - Well understood by management applications
- To implement this, the following is required:
  1. A read-only port type in the ieee8021BridgeBasePortTable
     - Similar to IANAifType (Dan Romascanu and Bert Wijnen)
     - DONE in d3.1
  2. A consolidation all common attributes into the base table
     - Almost DONE in d3.1
     - Move BEB Port Table "external" boolean to the base table
  3. A table for each port type containing at least a RowStatus
- To create a port, set the RowStatus in the port table. As a consequence:
  - Entries are created in the parent tables: ieee8021BridgeBasePortTable, etc.

How this changes 802.1ap d3.1…

1. Add RowStatus to ieee8021PbbVipTable
2. Add a “per port type” table in the appropriate MIBs
   - CBP in PBB-MIB
   - CEP / CNP / PNP in PB-MIB
   - C-VLAN Port in Q-BRIDGE-MIB
   - 1D Port Table in BRIDGE-MIB (Kevin Nolish)

Discussion related to this proposal

1. How are the standard port types maintained in the long term?
   - Maintaining a spreadsheet or external MIB under the OID registry is possible
2. How does a port change its type? ➔ CLOSED
   - One method is to
     1. delete the original bridge port
     2. create a new bridge port of the desired type in its place
   - Stated in response to comment 58.

Discussion related to this proposal (cont)

3. Is the port capabilities field still relevant if this dynamic port creation is implemented? ➔ CLOSED
   - Resolution of comment 74 removes port capability
4. How extensive are the changes to the current MIBs?
   - Small ➔ The recipe is outlined in this presentation

Discussion related to this proposal (cont)

5. An assessment of the attributes related to creation of a port is required
   - For a CreateAndGo operation, all the attributes must fit into 1 PDU
   - Reasonable defaults which can be changed later must be applied to as many port attributes as possible
6. Are references to the ENTITY-MIB required?
   - One possible reason to explore this is to deal with port capabilities (mentioned by Paul Bottorff)

Next steps

- Now that ports can be dynamically created, is there a requirement to create PIPs?
  - Yes. This is now a requirement based on the resolution of comment 57 for 802.1ah v4.0
- Whatever approach is selected for ports, the same approach must be used for components for consistency.
  - This is now a requirement based on the resolution of comment 56 for 802.1ah v4.0
- How are PIPs and CBPs associated if they can be created dynamically?

Conclusions

- For port creation, follow the same model as the IETF did for interfaces
- Once we have resolved port creation, we need to work through service examples to ensure
  - Address the “Next steps”
  - The model is consistent

BACKUP – OLD SLIDES

Conclusion for port creation from Atlanta

- At the plenary in Atlanta, the ability to create components and ports dynamically was accepted.
- The next draft will contain atleast 2 methods to create ports using the RowStatus in the:
  - ieee8021BridgeBasePortTable as proposed by Kevin Nolish in comment 119 (ACCEPTED)
  - ieee8021PbbBebPortTable as proposed by Zehavit Alon in comment 162 (ACCEPTED)

Conclusion for port creation from Atlanta (cont)

- A third method was proposed Nurit Sprecher in comment 151
  - Modify the ieee8021PbbVipTable to have a row status
  - The discussion on this topic was not completed due to time.
- The result is too many options for creating a port
  - It is also difficult to explain
  - Touching multiple tables to create a port is error prone
    - Is this even correct?
- Glenn Parson requested a proposal to clarify the issue

Thoughts on each proposal

Base Port Table Row Status

- Generic and applies to all ports
- **Issue**: How is the port type determined?

BEB Port Table Row Status

- Allows the creation of any port type using row status and port type
- **Issue**: This approach is not forward compatible. How is a new port type handled?

VIP Table Row Status

- This approach since it is similar to the ifTable structure
- **Issue**: Currently only applies to VIPs

Bridge port type

- Requires a Textual Convention similar to IANAifType
- The usage of IANAifType is discussed in RFC 2863
  - ifType's SYNTAX is a textual convention defined in a different document
  - This allows additional values to be documented without having to re-issue a new version of this document
- A smilar method to manage our port types is required
  - Currently, 6 types are defined

How this changes the upcoming draft… Based on d3-0

1. The ieee8021PbbBebPortTable is removed
   - Attributes are moved to the ieee8021BridgeBasePortTable
   - The exception is the RowStatus is Removed
2. ieee8021PbProviderBridgePortTable is removed.
3. The RowStatus in the ieee8021BridgeBasePortTable is removed
4. Add RowStatus to ieee8021PbbVipTable
5. Add the new per port tables in the appropriate MIBs
   - CBP in PBB-MIB
   - CEP / CNP / PNP in PB-MIB
   - C-VLAN Port in Q-BRIDGE-MIB
   - 1D Port Table in BRIDGE-MIB (Kevin Nolish)