P802.1CQ Flowchart Update

Roger Marks (EthAirNet Associates)  
(roger@ethair.net)  
Antonio de la Oliva (Interdigital)  
(aoliva@it.uc3m.es)

presentation to 802.1 TSN TG  
2020-11-09
P802.1CQ Flowcharts

- P802.1CQ/D0.5 includes flowcharts for PALMA
  - illustrative, since details are in the state machines
- Prior comment resolution indicates that PALMA will be replaced by updated MAAP
- Functionality described in PALMA flowcharts needs to be updated to MAAP flowcharts
  - will guide the state machines
- New functionality needs to be added
  - mainly considering challenges of multiple servers that join and leave the network
Message flows with self-assignment [updated]

Client 1

- DISCOVER/PROBE [random1, mcast(setX)] (setX)
  - random1 \not\in setX

- DISCOVER/PROBE [random2, mcast(setX)] (setX)
  - random2 \not\in setX

- DISCOVER/PROBE [random3, mcast(setX)] (setX)
  - random3 \not\in setX

- ADVERTISE/ANNOUNCE [client1, mcast(setX)] (setX)
  - client1 \in setX

Client 2

- DISCOVER/PROBE [random4, mcast(setX)] (setX)

Client 3

- DEFEND [client1, random4] (setX)
Message flows with server-based allocation

Client 1

DISCOVER/PROBE [random1, mcast(setX)] (setX)

DISCOVER/PROBE [random2, mcast(setX)] (setX)

OFFER [server1, random1] (setY1)

OFFER [server2, random2] (setY2)

REQUEST [unicast1, server1] (setY1)

ACK [server1, unicast1] (setY1)

Server 1

random1 ∉ setX

random2 ∉ setX

Note: If client address is from the random set, then setY includes at least one unicast address, for the client address.

Server 2

unicast1 ∈ setY1

where

priority(server1) > priority(server2)

for servers with matching Network ID

Client 2
Server Flow Diagram – ACTIVE State

Server START state

begin listening to block claim, null-block, and client/server multicast addresses

Server ACTIVE state

Issue NOTICE to null-block multicast address

Start Notice Timer

Issue NOTICE to null-block multicast address

null block multicast address

client/server multicast address

Yes

Notice Timer Timeout

Yes

Network ID< Network ID of NOTICE

Server INACTIVE state

Receive NOTICE

Yes

Receive several inactivation NOTICES to client/server multicast address

Server FLOW Diagram – ACTIVE State

Yes

receive REQUEST

valid set request

Issue OFFER per requested set

address set objection

Send ACK (error)

Send ACK

Issue OFFER with alternate set

Receive DISCOVER, PROBE, ADVERTISE, ANNOUNCE, or REQUEST message

ADVERTISE/ANNOUNCE

address set objection

Yes

Issue OFFER with alternate set

Send ACK (error)

Send ACK
Server Flow Diagram – INACTIVE State

1. Server INACTIVE state
   - continue listening to block claim, null-block, and server multicast addresses

2. Issue NOTICE to null-block multicast address
   - Start Notice Timer
   - Yes
   - Notice Timer Timeout

3. Start Reactivation Timer
   - Reactivation Timer Timeout
   - Server ACTIVE state

4. Receive NOTICE
   - Yes
   - Network ID of NOTICE
     - Yes

5. Receive NOTICE
   - Yes

Client Flow Diagram: START State

- Begin
- known server?
  - Yes: enter Client REQUESTING state
  - No: enter Client DISCOVERY state
- Continue?
  - Yes: Continue?
  - No: stop
- Restart

Options:
- No
- Yes

Continue?
Client Flow Diagram: DISCOVERY State

- **Enter Client DISCOVERY state from Client START State**
  - Generate block claim
  - Begin listening to block claim multicast address
  - Initialize Discovery Repeat Count

- **Disc Repeat Count = 0**
  - Yes
    - Disc Repeat Count
  - No
    - Decrement Discovery Repeat Count

- **Randomize source address**

- **Start Discovery Timer**

- **Issue DISCOVER/PROBE**

- **Enter Client DEFENDING state**
  - Adopt block claim

- **Enter Client REQUESTING state**
  - Stop listening to block claim multicast address

- **Enter Client START state at Restart**
  - Stop listening to block claim multicast address

- **Valid response DEFEND(s) received**
  - Yes
    - Adopt block claim
  - No
    - Discovery timer timeout
      - Yes
        - OFFER(s) received
      - No
    - Yes
      - Acceptable OFFER
      - No
Client Flow Diagram: REQUESTING State

- Enter Client REQUESTING state from Client DISCOVERY, DEFENDING, BOUND, or START State
- Select request set
- Initialize Request Repeat Count
- Decrement Request Repeat Count
  - Req Repeat Count = 0
    - START/Restart
  - No
    - Issue REQUEST
    - Start Request Timer
      - Yes
        - Error?
          - No
            - Adopt address assignment
            - Enter Client BOUND State
          - Yes
            - Request Timer Timeout
    - No
      - Receive ACK
        - Yes
          - Enter Client REQUESTING State
        - No
          - Enter Client REQUESTING State
          - Yes
            - Request Timer Timeout
            - Enter Client REQUESTING State
Client Flow Diagram: BOUND State

- **enter Client BOUND state from Client REQUESTING State**
- **start Assignment Timer**
- **90% Assignment Timer Timeout**
  - **No**
  - **Yes**
    - **enter Client REQUESTING State (retain assigned set)**
- **begin listening to client/server multicast address**
- **Receive NOTICE message**
  - **No**
  - **Yes**
    - **Network ID< Network ID of NOTICE**
Client Flow Diagram: DEFENDING State

1. Enter Client DEFENDING state from Client DISCOVERY State

2. Start Assignment Lifetime Timer

3. Issue ADVERTISE/ANNOUNCE

4. Start Advertise timer

5. Advertise Timer Timeout

   a. Yes
   b. No

   i. Receive OFFER message
      a. Yes
      b. No

   ii. Receive DEFEND message
      a. Yes
      b. No

   iii. Receive ADVERTISE/ANNOUNCE message
       a. Yes
       b. No

   iv. Receive DISCOVER/PROBE message
       a. Yes
       b. No

   v. Address set conflict?
       a. No
       b. Yes

   vi. Release assignment?
       a. Yes
       b. No

   vii. Issue DEFEND message

6. Assignment Lifetime Timer Timeout

   a. Yes
   b. No

   i. Start Assignment Lifetime Timer

   ii. Enter Client REQUESTING state

7. Acceptable OFFER

   a. Yes
   b. No

8. Stop listening to block claim multicast address

9. START/Restart

10. Remove conflicting address set from internal self-assignment
Proposed Comment Resolution

• CID 38
  - Revise, “Update draft in accordance with algorithms described in cq-marks-flowcharts-1120. Replace PALMA with upgraded version of MAAP, aligned with MAAP message formats. Specify PDU accordingly, with a common PDU structure among the message types. Structure messages so as to convey status of sender.”