

## Clarification Request

**References:** “e.g” Specified Tests 18.1, 20.0 14.Y.Y.1.1.16

**Date of BTL-WG Response:** June 2, 2022

**Background:** “e.g” Specified Tests 14.0.Final or 135.1-2013 Test-Number

### 14.YY.2.1.9 Duplicate Connection Test

Reference: YY.6.2, YY.6.2.1, YY.6.2.3

Purpose: To verify that duplicate hub connection requests result in the original connection being dropped.

Test Concept: With the IUT operating as hub, connect device D3 to the IUT's hub URI. When the connection is complete, D3 attempts to bring up a second connection to the IUT's hub URI. Verify that the IUT accepts the second connect request and closes the first connection. Repeat the reconnection, but with a new VMAC for D3 and ensure that the new request is accepted and the existing one dropped.

Test Steps:

1. MAKE(D3 connect to the IUT's hub function)
2. TRANSMIT PORT (D3-IUT hub first WebSocket),
  - Connect-Request,
  - 'Message ID' = (M1: any valid value),
  - 'Originating Virtual Address' absent
  - 'Destination Virtual Address' absent
  - 'Destination Options' absent
  - 'Data Options' absent
  - 'VMAC Address' = (D3's VMAC),
  - 'Device UUID' = (D3's UUID),
  - 'Maximum BVLC Length' = (the D3's maximum BVLC accepted length),
  - 'Maximum NPDU Length' = (the D3's maximum NPDU accepted length)
3. RECEIVE PORT (D3-IUT hub first WebSocket),
  - Connect-Accept,
  - 'Message ID' = M1,
  - 'Originating Virtual Address' absent
  - 'Destination Virtual Address' absent
  - 'VMAC Address' = (IUT's VMAC),
  - 'Device UUID' = (IUT's UUID),
  - 'Maximum BVLC Length' = (the IUT's maximum BVLC accepted length),
  - 'Maximum NPDU Length' = (the IUT's maximum NPDU accepted length)
4. MAKE(D1 connect a second WebSocket to the IUT's hub function)
5. TRANSMIT PORT (D3-IUT hub second WebSocket),
  - Connect-Request,
  - 'Message ID' = (M2: any valid value),
  - 'Originating Virtual Address' absent
  - 'Destination Virtual Address' absent
  - 'Destination Options' absent
  - 'Data Options' absent
  - 'VMAC Address' = (D3's VMAC),
  - 'Device UUID' = (D3's UUID),
  - 'Maximum BVLC Length' = (the D3's maximum BVLC accepted length),
  - 'Maximum NPDU Length' = (the D3's maximum NPDU accepted length)
6. RECEIVE PORT (D3-IUT hub second WebSocket),
  - Connect-Accept,

- 'Message ID' = M2,  
 -- 'Originating Virtual Address' absent  
 -- 'Destination Virtual Address' absent  
 'Destination Options' = (absent or a list of valid options),  
 -- 'Data Options' absent  
 'VMAC Address' = (IUT's VMAC),  
 'Device UUID' = (IUT's UUID),  
 'Maximum BVLC Length' = (the IUT's maximum BVLC accepted length),  
 'Maximum NPDU Length' = (the IUT's maximum NPDU accepted length)
7. RECEIVE PORT (D3-IUT hub first WebSocket),  
 Disconnect-Request,  
 'Message ID' = M3,  
 -- 'Originating Virtual Address' absent  
 -- 'Destination Virtual Address' absent  
 'Destination Options' = (absent or a list of valid options),  
 -- 'Data Options' absent
8. TRANSMIT PORT (D3-IUT hub first WebSocket),  
 Disconnect-ACK,  
 'Message ID' = M3,  
 -- 'Originating Virtual Address' absent  
 -- 'Destination Virtual Address' absent  
 -- 'Destination Options' absent  
 -- 'Data Options' absent
9. CHECK(that the IUT closed D3's initial WebSocket)
10. MAKE(D3 connect a third WebSocket to the IUT's hub function)
11. TRANSMIT PORT (D3-IUT hub second WebSocket),  
 Connect-Request,  
 'Message ID' = (M4: any valid value),  
 -- 'Originating Virtual Address' absent  
 -- 'Destination Virtual Address' absent  
 -- 'Destination Options' absent  
 -- 'Data Options' absent  
 'VMAC Address' = (a new VMAC for D3 which does not conflict with any other VMACs),  
 'Device UUID' = (D3's UUID),  
 'Maximum BVLC Length' = (the D3's maximum BVLC accepted length),  
 'Maximum NPDU Length' = (the D3's maximum NPDU accepted length)
12. RECEIVE PORT (D3-IUT hub third WebSocket),  
 Connect-Accept,  
 'Message ID' = M4,  
 -- 'Originating Virtual Address' absent  
 -- 'Destination Virtual Address' absent  
 'Destination Options' = (absent or a list of valid options),  
 -- 'Data Options' absent  
 'VMAC Address' = (IUT's VMAC),  
 'Device UUID' = (IUT's UUID),  
 'Maximum BVLC Length' = (the IUT's maximum BVLC accepted length),  
 'Maximum NPDU Length' = (the IUT's maximum NPDU accepted length)
13. RECEIVE PORT (D3-IUT hub second WebSocket),  
 Disconnect-Request,  
 'Message ID' = M5,  
 -- 'Originating Virtual Address' absent  
 -- 'Destination Virtual Address' absent  
 'Destination Options' = (absent or a list of valid options),  
 -- 'Data Options' absent

14. TRANSMIT PORT (D3-IUT hub second WebSocket),  
Disconnect-ACK,  
'Message ID' = M5,  
-- 'Originating Virtual Address' absent  
-- 'Destination Virtual Address' absent  
-- 'Destination Options' absent  
-- 'Data Options' absent
15. CHECK(that the IUT closed D3's second WebSocket)

#### Conflicting direction from the Standard...

##### AB.6.2.1 Duplicate Connections and VMAC Address Collisions

The BACnet/SC connections shall ensure that only one connection exists in a given context.

For the node switch, this shall ensure that only one direct connection to another node is used at a time. The local BVLL entity with its VMAC address and Device UUID is considered a connection peer of the node switch as well.

For the BACnet/SC hub function, this shall ensure that only one hub connection to a node is used at a time. The BACnet/SC node in the network port shall not be considered a connection peer unless it is currently connected to the local hub function by its hub connection.

The duplicate connection detection also provides detection of VMAC address collisions of initiating peers.

#### And from Clause AB.6.2.3 BACnet/SC Connection Accepting Peer State Machine

Connect-Request received, known Device UUID

On receipt of a Connect-Request message from the initiating peer whose 'Device UUID' is equal to the initiating peer device UUID of an existing connection, then return a Connect-Accept message, disconnect and close the existing connection to the connection peer node with matching Device UUID, and enter the CONNECTED state.

#### Problem:

The test mandates that the IUT return a Connect-Accept message (Step 6) prior to sending a Disconnect-Request message (Step 7) and again in Steps 12 and 13. While this sequence is consistent with the sequence explicitly described in clause AB.6.2.3, it violates the 'one connection' mandate in clause AB.6.2.1.

#### Question:

**Is it permissible for the IUT to Disconnect the existing connection prior to sending the Connect-Accept message and still pass the test?**

#### Response:

**Yes. The test will be adjusted to so that the socket can be disconnected before Connect-Accept.**