

Clarification Request

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

Date of BTL-WG Response: May 19, 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)

From Standard

AB.7.5.5 Closing WebSocket Connections

WebSocket connections may be closed by either end at any time. See RFC 6455.

Problem:

The test currently mandates a Disconnect Request/Ack sequence, but the standard allows for the IUT to close the WebSocket Connection 'at any time'. Especially in this case, where there is an error condition being induced by the duplicate connection and the IUT must close the connection anyway, there is little purpose in sending a Disconnect Request and waiting for an Ack.

Question:

Is the IUT permitted to close the WebSocket connection without initiating the Disconnect-Request message in Steps 7 and 13 and still pass the test?

Response:

No. The AWAITING_REQUEST state requires disconnection of the original connection before closing the web socket when a duplicate connection is received.

Reference: AB.6.2.3