Clarification Request

_____

References: 135.1-2019 - Tests 9.24.1.1 through 9.24.1.4 and BTL Specified Tests 9.24.1.5

Date of BTL-WG Response: February 18, 2021

⊠ Mark when All Actions Necessitated have been Completed

Background:

From the standard 135-2020: DeviceCommunicationControl Service

16.1.2 Service Procedure

After verifying the validity of the request, including the 'Time Duration' and 'Password' parameters, the responding BACnet- user shall respond with a 'Result(+)' service primitive. If the request is valid and the 'Enable/Disable' parameter is DISABLE_INITIATION, the responding BACnet-user shall discontinue the initiation of messages except for I-Am requests issued in accordance with the Who-Is service procedure and ConfirmedAuditNotification and UnconfirmedAuditNotification requests. Communication shall be disabled in this manner until either the 'Time Duration' has expired or a valid DeviceCommunicationControl (with 'Enable/Disable' = ENABLE) or, if supported, a valid ReinitializeDevice (with 'Reinitialized State of Device' = WARMSTART or COLDSTART) message is received.

. . .

If the request is valid and the 'Enable/Disable' parameter is the deprecated value DISABLE, the request shall be ignored and an Error-PDU with 'Error Class' = SERVICES and 'Error Code' = SERVICE_REQUEST_DENIED shall be issued.

16.1.1.1.2 Enable/Disable

This parameter is an enumeration that may take on the values ENABLE, DISABLE, or DISABLE_INITIATION. It is used to indicate whether the responding BACnet-user is to enable all, disable initiation, or disable all communications on the network interface. When this parameter has a value of ENABLE, communications shall be enabled. When this parameter has a value of DISABLE, network communications shall be disabled as described in the DeviceCommunicationControl service procedure. When this parameter has a value of DISABLE_INITIATION, the initiation of communications shall be disabled as described in the DeviceCommunicationControl service procedure.

From BTL Specified Test 18.0, one of those test is here shown:

9.24 DeviceCommunicationControl Service Execution Test

9.24.1 Positive DeviceCommunicationControl Service Execution Tests

9.24.1.5 Finite Time Duration Restored by ReinitializeDevice

Reason for Change: Modified test to remove dependency on EPICS values.

Purpose: To verify the correct execution of the DeviceCommunicationControl request service procedure when a finite time duration is specified and communication is restored using the ReinitializeDevice service.

Test Steps:
1. *READ Y = (Device, X), Object_Name*
2. TRANSMIT DeviceCommunicationControl-Request,
        'Time Duration' = (a value T > 1, in minutes, selected by the tester)
        'Enable/Disable' = DISABLE,
        'Password' = (any appropriate password as described in the Test Concept)
3. RECEIVE BACnet-SimpleACK-PDU
4. WAIT **Internal Processing Fail Time**
5. TRANSMIT ReadProperty-Request,
        'Object Identifier' = (Device, X),
        'Property Identifier' = (any required non-array property of the Device object)
6. WAIT (an arbitrary time > **Internal Processing Fail Time** selected by the tester, and < T as specified in the
                DeviceCommunicationControl-Request)
7. CHECK (Verify that the IUT has not transmitted any messages since the acknowledgment in step 2.)
8. TRANSMIT ReinitializeDevice-Request,
        'Reinitialize State of Device' = WARMSTART,
        'Password' = (any appropriate password as described in the Configuration Requirements)
9. RECEIVE BACnet-Simple-ACK-PDU
10.   CHECK (Did the IUT perform a WARMSTART reboot?)
11.   VERIFY (Device, X), *Object_Name = Y*~~(any required non-array property) = (the value for this property as described in the EPICS)~~

Problem:

The service no longer has a DISABLE parameter.

Question:

1. Should the tests be run?
2. Would a device claiming Protocol_Revision 16 that returns Error Class SERVICES Error Code: SERVICE_REQUEST_DENIED in response to any **DeviceCommunicationControl Service** that specifies the DISABLE parameter, pass BTL testing?

Response:

1. **Yes.**
2. **No. The device needs to behave as specified in the device's Protocol_Revision.**