BACnet® TESTING LABORATORIES ADDENDA

Addendum fix2 to BTL Test Package 26.0

Revision Final
Revised 5/27/2025

Approved by the BTL Working Group on April 17, 2025
Approved by the BTL Working Group Voting Members on May 23, 2025
Published on May 29, 2025

[This foreword and the "Overview" on the following pages are not part of this Test Package. They are merely informative and do not contain requirements necessary for conformance to the Test Package.]

FOREWORD

The purpose of this addendum is to present current changes being made to the BTL Test Package. These modifications are the result of change proposals made pursuant to the continuous maintenance procedures and of deliberations within the BTL-WG Committee. The changes are summarized below.

In the following document, language to be added to existing clauses within the BTL Test Package 26.0 is indicated through the use of *italics*, while deletions are indicated by ~~strikethrough~~. Where entirely new subclauses are proposed to be added, plain type is used throughout.

In contrast, changes to BTL Specified Tests also contain a yellow highlight to indicate the changes made by this addendum. When this addendum is applied, all highlighting will be removed. Change markings on tests will remain to indicate the difference between the new test and an existing 135.1 test. If a test being modified has never existed in 135.1, the applied result should not contain any change markings. When this is the case, square brackets will be used to describe the changes required for this test.

Each addendum can stand independently unless specifically noted via dependency within the addendum. If multiple addenda change the same test or section, each future released addendum that changes the same test or section will note in square brackets whether or not those changes are reflected.

BTL-26.0 fix2-1: Trend Log Tests Require Display of Records with Status_Flags [BTLWG-1375, CR-0542]

Overview:

ic-135-2024-3 stipulates: "A device that claims T-V-A or T-AVM-A must be able to retrieve and display the Timestamp, LogDatum, and StatusFlags fields contained in a log record."

Changes:

Checklist Changes

None

Test Plan Changes

[Change all test 8.21.9 references from 135.1-2023 to BTL Specified Tests]

5.12.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB.

~~135.1-2023~~*BTL* - 8.21.9 - Presents Log Records		
	Test Conditionality	Must be executed
	Test Directives	The information conveyed to the user shall at minimum be as described in the AE-ELV-A BIBB or as described in AE-VN-A. Any other information conveyed to the user shall be consistent with the data contained in the notification.
	Testing Hints	

[Change Test Plan 7.1.3, test 8.21.9 references from 135.1-2023 to BTL Specified Tests and modify]

7.1 Trending - View - A

7.1.3 Interoperates with Trend Logs

The IUT can interoperate with Trend Log objects.

~~135.1-2023~~*BTL* - 8.21.9 - Presents Log Records		
	Test Conditionality	Must be executed
	Test Directives	*Execute this test against* ~~This test shall be executed with a reference server device containing~~ a Trend Log object. Repeat the test for *Boolean, Real, ENUMERATED, Unsigned32, Integer32, BitString, and Null* ~~Real, INTEGER32, BOOLEAN, Bit String, Enumerated, and NULL~~ datatypes. *For at least one iteration of this test, the log records shall include LogDatum and StatusFlags fields.*
	Testing Hints	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that should accept a written NULL.

7.1.4 Interoperates with Trend Log Multiple Objects

The IUT can interoperate with Trend Log Multiple objects in devices claiming Protocol_Revision 7 or higher.

~~135.1-2023~~*BTL* - 8.21.9 - Presents Log Records		
	Test Conditionality	Must be executed
	Test Directives	~~This test shall be executed with a reference server device containing a Trend Log Multiple object claiming Protocol_Revision 7 or higher.~~ Execute test against *a* Trend Log Multiple that contains all required

		datatypes (Boolean, Real, *ENUMERATED*~~Enumerated~~, Unsigned32, Integer32, *BitString*~~Bit String~~, and *Null*~~NULL~~).
	Testing Hints	Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that should accept a written NULL.

7.1.5 Supports Reading Items by Time with a Positive Count

The IUT can initiate one or more ReadRange requests that specify the Time form and a Positive 'Count' that access a tester-specified portion of log records.

135.1-2023 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	Test Conditionality	Must be executed
	Test Directives	*Perform this test once with a Trend Log object and once with a Trend Log Multiple object.* ~~This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.~~
	Testing Hints	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Time form of the ReadRange service, with a Positive 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

7.1.6 Supports Reading Items by Time with a Negative Count

The IUT can initiate one or more ReadRange requests that specify the Time form and a Negative 'Count' that access a tester-specified portion of log records.

135.1-2023 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	Test Conditionality	Must be executed
	Test Directives	*Perform this test once with a Trend Log object and once with a Trend Log Multiple object.* ~~This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.~~
	Testing Hints	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Time form of the ReadRange service, with a Negative 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

7.1.7 Supports Reading Items by Position with a Positive Count

The IUT can initiate one or more ReadRange requests that specify the Position form and a Positive 'Count' that access a tester-specified portion of log records.

135.1-2023 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	Test Conditionality	Must be executed
	Test Directives	*Perform this test once with a Trend Log object and once with a Trend Log Multiple object.* ~~This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.~~
	Testing Hints	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Position form of the ReadRange service, with a Positive 'Count'.

		The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

7.1.8 Supports Reading Items by Position with a Negative Count

The IUT can initiate one or more ReadRange requests that specify the Position form and a Negative 'Count' that access a tester-specified portion of log records.

135.1-2023 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	Test Conditionality	Must be executed
	Test Directives	*Perform this test once with a Trend Log object and once with a Trend Log Multiple object.* ~~This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.~~
	Testing Hints	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Position form of the ReadRange service, with a Negative 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

7.1.9 Supports Reading Items by Sequence Number with a Positive Count

The IUT can initiate one or more ReadRange requests that specify the Sequence Number form and a Positive 'Count' that access a tester-specified portion of log records.

135.1-2023 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	Test Conditionality	Must be executed
	Test Directives	*Perform this test once with a Trend Log object and once with a Trend Log Multiple object.* ~~This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.~~
	Testing Hints	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Sequence Number form of the ReadRange service, with a Positive 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

7.1.10 Supports Reading Items by Sequence Number with a Negative Count

The IUT can initiate one or more ReadRange requests that specify the Sequence Number form and a Negative 'Count' that access a tester-specified portion of log records.

135.1-2023 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	Test Conditionality	Must be executed
	Test Directives	*Perform this test once with a Trend Log object and once with a Trend Log Multiple object.* ~~This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.~~
	Testing Hints	The 'Range' value in Steps 2 and 4 must include at least one occurrence of the Sequence Number form of the ReadRange service, with a Negative 'Count'. The IUT shall be capable of presenting any portion of records selected by the tester. The set of records in this test shall contain at least one log-status entry.

7.1.11 Supports Reading Items with no Range

The IUT can initiate one or more ReadRange requests that specify no range that access a tester-specified portion of log records.

135.1-2023 - 8.21.8 - Reading a Range of Items Using Any Valid Range		
	Test Conditionality	Must be executed
	Test Directives	*Perform this test once with a Trend Log object and once with a Trend Log Multiple object.* ~~This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher for Trend Logs and Protocol_Revision 7 or higher for Trend Log Multiple. Perform the test on the Trend Log object and repeat on Trend Log Multiple object if Protocol_Revision 7 or higher is supported.~~
	Testing Hints	The 'Range' value in Steps 2 and 4 specify no range The set of records in this test shall contain at least one log-status entry.

7.1.12 Is able to present Double datatypes in trend logging objects

The IUT can present optional datatypes.

~~135.1-2023~~*BTL* - 8.21.9 - Presents Log Records		
	Test Conditionality	Must be executed~~ and repeated if IUT claims Protocol_Revision 7 or higher~~.
	Test Directives	*Perform this test once with a Trend Log object containing the specified data type and once with a Trend Log Multiple object containing the specified data type.* ~~For Trend Log objects, this test shall be executed with a reference server device claiming Protocol_Revision 3 or higher and for Trend Log Multiple objects, this test shall be executed with a reference server device claiming Protocol_Revision 7 or higher. For Protocol_Revision 3 or higher, perform the test using Double datatype and for Protocol_Revision 7 or higher, repeat the test using a Trend Log Multiple that contains a Double datatype.~~
	Testing Hints	

7.1.13 Is able to present Octet String datatypes in trend logging objects

The IUT can present optional datatypes.

~~135.1-2023~~*BTL* - 8.21.9 - Presents Log Records		
	Test Conditionality	Must be executed~~ and repeated if IUT claims Protocol_Revision 7 or higher~~.
	Test Directives	*Perform this test once with a Trend Log object containing the specified data type and once with a Trend Log Multiple object containing the specified data type.* ~~For Trend Log objects, this test shall be executed with a reference server device claiming Protocol_Revision 3 or higher and for Trend Log Multiple objects, this test shall be executed with a reference server device claiming Protocol_Revision 7 or higher. For Protocol_Revision 3 or higher, perform the test using Octet String datatype and for Protocol_Revision 7 or higher, repeat the test using a Trend Log Multiple that contains a Octet String datatype.~~
	Testing Hints	

7.1.14 Is able to present Character String datatypes in trend logging objects

The IUT can present optional datatypes.

~~135.1-2023~~*BTL* - 8.21.9 - Presents Log Records		
	Test Conditionality	Must be executed~~ and repeated if IUT claims Protocol_Revision 7 or higher~~.
	Test Directives	*Perform this test once with a Trend Log object containing the specified data type and once with a Trend Log Multiple object containing the specified data type.* ~~For Trend Log objects, this test shall be executed with a reference server device claiming Protocol_Revision 3 or higher and for Trend Log Multiple objects, this test shall be executed with a~~

		~~reference server device claiming Protocol_Revision 7 or higher. For Protocol_Revision 3 or higher, perform the test using Character String datatype and for Protocol_Revision 7 or higher, repeat the test using a Trend Log Multiple that contains a Character String datatype.~~
	Testing Hints	

7.1.15 Is able to present Date datatypes in trend logging objects

The IUT can present optional datatypes.

~~135.1-2023~~*BTL* **- 8.21.9 - Presents Log Records**		
	Test Conditionality	Must be executed~~ and repeated if IUT claims Protocol_Revision 7 or higher~~.
	Test Directives	*Perform this test once with a Trend Log object containing the specified data type and once with a Trend Log Multiple object containing the specified data type.* ~~For Trend Log objects, this test shall be executed with a reference server device claiming Protocol_Revision 3 or higher and for Trend Log Multiple objects, this test shall be executed with a reference server device claiming Protocol_Revision 7 or higher. For Protocol_Revision 3 or higher, perform the test using Date datatype and for Protocol_Revision 7 or higher, repeat the test using a Trend Log Multiple that contains a Date datatype.~~
	Testing Hints	

7.1.16 Is able to present Time datatypes in trend logging objects

The IUT can present optional datatypes.

~~135.1-2023~~*BTL* **- 8.21.9 - Presents Log Records**		
	Test Conditionality	Must be executed~~ and repeated if IUT claims Protocol_Revision 7 or higher~~.
	Test Directives	*Perform this test once with a Trend Log object containing the specified data type and once with a Trend Log Multiple object containing the specified data type.* ~~For Trend Log objects, this test shall be executed with a reference server device claiming Protocol_Revision 3 or higher and for Trend Log Multiple objects, this test shall be executed with a reference server device claiming Protocol_Revision 7 or higher. For Protocol_Revision 3 or higher, perform the test using Time datatype and for Protocol_Revision 7 or higher, repeat the test using a Trend Log Multiple that contains a Time datatype.~~
	Testing Hints	

7.1.17 Is able to present BACnetObjectIdentifier datatypes in trend logging objects

The IUT can present optional datatypes.

~~135.1-2023~~*BTL* **- 8.21.9 - Presents Log Records**		
	Test Conditionality	Must be executed~~ and repeated if IUT claims Protocol_Revision 7 or higher~~.
	Test Directives	*Perform this test once with a Trend Log object containing the specified data type and once with a Trend Log Multiple object containing the specified data type.* ~~For Trend Log objects, this test shall be executed with a reference server device claiming Protocol_Revision 3 or higher and for Trend Log Multiple objects, this test shall be executed with a reference server device claiming Protocol_Revision 7 or higher. For Protocol_Revision 3 or higher, perform the test using BACnet Object Identifier datatype and for Protocol_Revision 7 or higher, repeat the test using a Trend Log Multiple that contains a BACnet Object Identifier datatype.~~
	Testing Hints	

7.2 Trending - Advanced View and Modify - A

7.2.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB.

~~**135.1-2023 - 8.21.9 - Presents Log Records**~~		
	~~**Test Conditionality**~~	~~Must be executed~~
	~~**Test Directives**~~	~~This test shall be executed with a reference server device claiming Protocol_Revision 3 or higher.~~ ~~Repeat the test for BOOLEAN, Real, Enumerated, Unsigned32, Integer23, Bit String, and Null datatypes~~
	~~**Testing Hints**~~	~~Schedule_Default and Present_Value of the Schedule Object, Alarm_Values and Fault_Values of the CharacterString Value Object and Low_Diff_Limit in the Loop Object are standard properties that should accept a written NULL.~~
135.1-2023 - 8.18.3 - Reading and Presenting Properties		
	Test Conditionality	Must be executed
	Test Directives	
	Testing Hints	Repeat the test for a Trend Log, Trend Log Multiple, Event Enrollment, and Notification Class object. Repeat the test for each of the properties listed in the table in the BIBB definition.
135.1-2023 - 8.22.4 - Accepting Input and Modifying Properties		
	Test Conditionality	Must be executed.
	Test Directives	
	Testing Hints	Repeat the test for a Trend Log, Trend Log Multiple, Event Enrollment, and Notification Class object. Repeat the test for each of the properties listed in the table in the BIBB definition. Repeat the test for a variety of values that cover the range of values required by the BIBB.

Specified Test Changes

[Move test 8.21.9 from 135.1-2019 into the BTL Specified Tests, and modify]

8.21.9 Presents Log Records

Purpose: To verify that the IUT can initiate one or more ReadRange requests that access and present a tester-specified portion of log records. It is a generic test used to test data presentation requirements.

Test Concept: Run test in Clause 8.21.8 and verify that the data presentation meets the criteria specified by the BIBB being tested.

Notes to Tester: The values presented by the IUT may differ from the values transmitted on the wire due to rounding, truncation, formatting, language, conversion, etc.

~~Notes to Tester: The IUT is not required to display records containing log-status values.~~

BTL-26.0 fix2-2: Router Binding via Who-Is_Router-To-Network - Alternate Source [BTLWG-1647, CR-0573]

Overview:

This issue is for BTLWG-1647. The test steps of 135.1-2023 10.7.3 Router Binding via Who-Is-Router-To-Network do not accommodate a router IUT that lacks the ability to initiate its own confirmed requests. Clarification request BTL-CRR-0573 states that it is acceptable for these types of IUTs to forward a confirmed request initiated by another device to pass the test.

Additionally, 10.7.3's Test plan Test Directives did not state if the test should be executed with or without a network number in the Who-Is-Router-To-Network.

Changes:

Checklist Changes

None

Test Plan Changes

[Modify10.7.3 test name and test directives in sections 2.1.5 and 2.1.6 of BTL Test Plan]
[Remove 10.5.1.1 from section 2.1.5 of BTL Test Plan]
[Remove 10.5.1.2 from section 2.1.6 of BTL Test Plan]

2.1 Basic Functionality (Applies To All BACnet Devices)

2.1.5 Uses Who-Is-Router-To-Network (any Network) for Router Address Discovery

The IUT can generate a Who-Is-Router-To-Network message with no network number specified. The IUT examines the I-Am-Router-To-Network message(s) sent in response to determine the MAC address of a router that can forward messages to a particular network.

~~135.1-2023~~*BTL* - 10.7.3 - Router Binding via Who-Is-Router-To-Network		
	Test Conditionality	Must be executed.
	Test Directives	*Execute this test using a Who-Is-Router-To-Network with no network number specified.*
	Testing Hints	
~~135.1-2023 - 10.5.1.1 - Who-Is-Router-To-Network - General Query~~		
	~~Test Conditionality~~	~~Must be executed.~~
	~~Test Directives~~	
	~~Testing Hints~~	

2.1.6 Uses Who-Is-Router-To-Network (Specific Network) for Router Address Discovery

The IUT can generate a Who-Is-Router-To-Network message, with a specific network specified. The IUT examines the I-Am-Router-To-Network message sent in response to determine the MAC address of a router that can forward messages to the specific network.

~~135.1-2023~~*BTL* - 10.7.3 - Router Binding via Who-Is-Router-To-Network		
	Test Conditionality	Must be executed.
	Test Directives	*Execute this test using a Who-Is-Router-To-Network with a network number specified.*
	Testing Hints	
~~135.1-2023 - 10.5.1.2 - Who-Is-Router-To-Network - Specific Network Number~~		
	~~Test Conditionality~~	~~Must be executed.~~
	~~Test Directives~~	
	~~Testing Hints~~	

Specified Test Changes

[Move section 10.7 from 135.1-2023 into the BTL Specified Tests, and modify]

10.7 Route Binding Tests
This subclause defines the tests necessary to demonstrate the portion of BACnet Network Layer functionality that is used to determine network routes. These tests are generic and are meant to be applied to both router and non-router nodes.

The tests assume that the IUT is located on network DNET1 and the TD appears to be a router to network DNET2. The value DNET3 is assigned a unique network number. If the IUT can initiate requests, it will be configured to send those requests to a device (D2A) on network DNET2. The IUT will also be expected to respond to requests from device D2A. The test descriptions assume that the TD will mimic device D2A. *If the IUT is a router then it is a router to network DNET4 and device D4A resides locally on that network.*

Note: Clauses 6.5.1 and 6.5.3 indicate that there are only two ways for a non-router to transmit a request (on the local network and destined for a remote network), neither of which includes network layer source routing information. If the IUT is configured as a non-router and it emits any NPDU with SNET/SADR fields during the tests in this subclause, then it shall fail.

Note: Clauses 6.6 and 6.6.3.3 define BACnet routers and the network layer services reserved for routers. If the IUT is configured as a non-router and it emits any I-Am-Router-To-Network or I-Could-Be-Router-To-Network NPDUs during the tests in this subclause (including during test 10.7.2), then it shall fail.

[Move test 10.7.3 from 135.1-2023 into BTL Specified Tests, and modify]

10.7.3 Router Binding via Who-Is-Router-To-Network

Reason for Change: Modify test to allow for the case where IUT is a router that does not support initiating confirmed requests.

Purpose: To verify that the IUT can ~~initiate~~*send* requests to a remote network after the IUT uses the Who-Is-Router-To-Network Network Layer service to discover the MAC address of the router to that remote network.

Test Concept: The IUT broadcasts a Who-Is-Router-To-Network request to discover the router to the desired network. The ~~TD~~*IUT* transmits a request to a device on the remote network without performing any further form of dynamic router binding. ~~If the IUT does not support Who-Is-Router-To-Network router binding, then this test shall be omitted.~~ If the IUT cannot initiate a ReadProperty request, then another confirmed service can be substituted. *If the IUT is a router that does not support initiating confirmed service requests then it may forward a confirmed service request from device D4A instead.* The IUT may use either the general query or specific network number query form of the Who-Is-Router-To-Network service.

Note that Clause 6.5.3 specifically mentions router binding via Who-Is-Router-To-Network and does not mention router binding by lurking and noting unsolicited I-Am-Router-To-Network messages.

Test Steps:

1. MAKE (IUT transmit Who-Is-Router-To-Network to discover the router to DNET2)
2. RECEIVE
   DA = BROADCAST,
   SA = IUT,
   Who-Is-Router-To-Network,
   | (DA = BROADCAST,
   SA = IUT,
   Who-Is-Router-To-Network,
   DNET = DNET2)
3. TRANSMIT
   DESTINATION = BROADCAST,
   SOURCE = TD,
   I-Am-Router-To-Network,
   Network Numbers = DNET2

4. *IF (the IUT can initiate a ReadProperty request or any other confirmed request) THEN*
~~4~~5. MAKE (IUT transmit a ReadProperty request to the D2A device on the remote network)
~~5~~6. RECEIVE
DA = TD,
SA = IUT,
DNET = DNET2,
DADR= D2A,
Hop Count = 255,
BACnet-Confirmed-Request-PDU,
'Service Choice' = ReadProperty-Request,
'Object Identifier' = (O1, any BACnet standard object in D2A),
'Property Identifier' = (P1, any required property of the specified object)
~~6~~7. TRANSMIT
DA = IUT,
SA = TD,
SNET = DNET2,
SADR = D2A,
BACnet-ComplexACK-PDU,
'Service ACK Choice' = ReadProperty-ACK,
'Object Identifier' = O1,
'Property Identifier' = P1,
'Property Value' = (any valid value)
ELSE -- (IUT is a router that does not support initiating confirmed service requests)
8. TRANSMIT
DA = IUT,
SA = D4A,
DNET = DNET2,
DADR = D2A,
Hop Count = 255,
BACnet-Confirmed-Request-PDU,
'Service Choice' = ReadProperty-Request,
'Object Identifier' = (O1, any BACnet standard object in D2A),
'Property Identifier' = (P1, any required property of the specified object)
9. RECEIVE
DA = TD,
SA = IUT,
DNET = DNET2,
DADR = D2A,
SNET = DNET4
SADR = D4A
Hop Count = (any integer x: 0 < x < 255),
BACnet-Confirmed-Request-PDU,
'Service Choice' = ReadProperty-Request,
'Object Identifier' = O1
'Property Identifier' = P1
10. TRANSMIT
DA = IUT,
SA = TD,
DNET = DNET4,
DADR = D4A,
SNET = DNET2,
SADR = D2A
Hop Count = 254,
BACnet-Confirmed-Request-PDU,
'Service Choice' = ReadProperty-ACK,
'Object Identifier' = O1
'Property Identifier' = P1
'Property Value' = (any valid value)

BTL-26.0 fix2-3: Lighting Command Operation RAMP_TO Test[BTLWG-1658]

Overview:
The requested value of Tracking_Value < 0 in step 25 must be a typing error. Tracking_Value cannot be <0 and in step 25 we are fading down therefore the value for Tracking_Value is supposed to be <100.

Changes:

Checklist Changes

None

Test Plan Changes

[Change all references to test 7.3.2.39.5 Lighting Command Operation RAMP_TO Test from 135.1 to BTL]

Specified Test Changes

7.3.2.39.5 Lighting Command Operation RAMP_TO Test

Reason for change: wrong expected value for Tracking_Value in Step 25

Purpose: To verify the correct operation of RAMP_TO lighting command by observing the value of Present_Value, In_Progress and Tracking_Value.

Test Concept: The TD writes to Present_Value at each end of the range (i.e., 0% or 100%), and then writes to the Lighting Command Operation with RAMP_TO with a slow enough ramp rate to allow In_Progress and Tracking_Value to be observed while set to RAMP_ACTIVE. The Tracking_Value will be checked at the end of the ramp to verify that it tracked the target level. The IUT shall be tested for ramp up (0% to 100%) and ramp down (100% to 0%).

Configuration Requirements: O1 shall be configured such that all slots in the Priority_Array numerically less than PTY1 have a value of NULL and no internal algorithms are issuing commands to O1 at a priority numerically less than or equal to PTY1. V1 > 1 and V2 < 100%

Test Steps:

-- Start with 0% Present_Value to test ramp up
1. WRITE Present_Value = 0, ARRAY INDEX = PTY1
2. VERIFY Present_Value = 0
3. WAIT Internal Processing Fail Time
4. VERIFY Tracking_Value = 0

-- Write a RAMP_TO command (operation, target-value, priority, ramp-rate)
5. WRITE Lighting_Command = (RAMP_TO, V1, PTY1, any valid rate)
6. WAIT Internal Processing Fail Time
7. VERIFY Priority_ Array = V1, ARRAY INDEX = PTY1
8. VERIFY Present_Value =V1

-- Check In_Progress while ramping up
9. VERIFY In_Progress = RAMP_ACTIVE

-- Make sure that Tracking_Value increases with the ramp-rate
10. WHILE (In_Progress <> IDLE) DO {
11. VERIFY Tracking_Value > 0 < V1
12. CHECK (Tracking_Value is increasing with the ramp-rate)}

-- When ramping up is completed, check In_Progress and Tracking_Value
13. VERIFY In_Progress = IDLE
14. VERIFY Tracking_Value = V1

-- Now repeat the test with 100% Present_Value to test ramp down
15. WRITE Present_Value = 100, ARRAY INDEX = PTY1
16. VERIFY Present_Value = 100
17. WAIT Internal Processing Fail Time
18. VERIFY Tracking_Value = 100

-- Write a RAMP_TO command (operation, target-value, priority, ramp-rate)
19. WRITE Lighting_Command = (RAMP_TO, V2, PTY1, any valid rate)
20. WAIT Internal Processing Fail Time
21. VERIFY Priority_ Array = V2, ARRAY INDEX = PTY1
22. VERIFY Present_Value = V2

-- Check In_Progress while ramping up
23. VERIFY In_Progress = RAMP_ACTIVE,

-- Make sure that Tracking_Value decreases with the ramp-rate
24. WHILE (In_Progress <> RAMP_ACTIVE) DO {
25. VERIFY Tracking_Value < ~~0~~ *100*
26. VERIFY Tracking_Value > V2
27. CHECK (Tracking_Value is decreasing with the ramp-rate)}

-- Check In_Progress and Tracking_Value
28. VERIFY In_Progress = IDLE
29. VERIFY Tracking_Value = V2

BTL-26.0 fix2-4: Authorization Mode Test - Authentication_Status Requirement [BTLWG-1675]

Overview:

ic-135-2024-2 stipulates: " It is a local matter what the value of Authentication_Status property is when the Authorization_Mode property is written to NONE."

Changes:

Checklist Changes

None

Test Plan Changes

[Change test 7.3.2.41.4 in Test Plan 3.44.1 from 135.1-2023 to BTL Specified Tests]

Specified Test Changes

[Move test 7.3.2.41.4 from 135.1-2019 into the BTL Specified Tests, and modify]

7.3.2.41.4 Authorization Mode Test

Reason for Change: Remove testing for NONE Authorization_Mode.

Purpose: To verify each authorization mode supported by this IUT.

Test Concept: For each authorization mode supported by the IUT, a valid credential is presented at the access point to verify that the appropriate action is taken.

Configuration Requirements: See Clause 7.3.2.41. This test requires the following additional configuration:

a) An active credential with valid access rights for the access point shall be represented by Access Credential object C1.
b) An active credential with no valid access rights shall be represented by Access Credential object C2.

Note: If the VERIFICATION_REQUIRED or AUTHORIZATION_DELAYED mode is supported, the vendor must provide a mechanism for external verification to be performed.

Test Steps:

-- verify GRANT_ACTIVE mode
1. IF (GRANT_ACTIVE is supported) THEN
   READ EventTag = Access_Event_Tag
   WRITE Authorization_Mode = GRANT_ACTIVE
   MAKE (present credential C2 at credential reader for this access point)
   VERIFY Access_Event = GRANTED
   VERIFY Access_Event_Time = (the time that credential C2 was presented)
   VERIFY Access_Event_Credential = C2
   VERIFY Access_Event_Tag = EventTag + 1

-- verify DENY_ALL mode
2. IF (DENY_ALL is supported) THEN
   READ EventTag = Access_Event_Tag
   WRITE Authorization_Mode = DENY_ALL
   MAKE (present credential C1 at credential reader for this access point)
   VERIFY Access_Event = DENIED_DENY_ALL
   VERIFY Access_Event_Time = (the time that credential C1 was presented)

VERIFY Access_Event_Credential = C1
VERIFY Access_Event_Tag = EventTag + 1

-- verify VERIFICATION_REQUIRED mode (verification authorized)
3. IF (VERIFICATION_REQUIRED is supported) THEN
READ EventTag = Access_Event_Tag
WRITE Authorization_Mode = VERIFICATION_REQUIRED
MAKE (present credential C1 at credential reader for this access point)
VERIFY Access_Event = VERIFICATION_REQUIRED
VERIFY Access_Event_Time = (the time that credential C1 was presented most recently)
VERIFY Access_Event_Credential = C1
VERIFY Authentication_Status = WAITING_FOR_VERIFICATION
MAKE (external verification process grants access)
VERIFY Access_Event = GRANTED
VERIFY Access_Event_Time = (the time that verification process granted access)
VERIFY Access_Event_Credential = C1
VERIFY Access_Event_Tag = EventTag + 1

-- verify VERIFICATION_REQUIRED mode (verification denied)
READ EventTag = Access_Event_Tag
WRITE Authorization_Mode = VERIFICATION_REQUIRED
MAKE (present credential C1 at credential reader for this access point)
VERIFY Access_Event = VERIFICATION_REQUIRED
VERIFY Access_Event_Time = (the time that credential C1 was presented)
VERIFY Access_Event_Credential = C1
VERIFY Authentication_Status = WAITING_FOR_VERIFICATION
MAKE (external verification process denies access)
VERIFY Access_Event = DENIED_VERIFICATION_FAILED
VERIFY Access_Event_Time = (the time that verification process denied access)
VERIFY Access_Event_Credential = C1
VERIFY Access_Event_Tag + 1

-- verify VERIFICATION_REQUIRED mode (verification timeout)
READ EventTag = Access_Event_Tag
WRITE Authorization_Mode = VERIFICATION_REQUIRED
MAKE (present credential C1 at credential reader for this access point)
VERIFY Access_Event = VERIFICATION_REQUIRED
VERIFY Access_Event_Time = (the time that credential C1 was presented)
VERIFY Access_Event_Credential = C1
VERIFY Authentication_Status = WAITING_FOR_VERIFICATION
MAKE (external verification process does not respond within verification time)
WAIT Verification_Time
VERIFY Access_Event = DENIED_VERIFICATION_TIMEOUT
VERIFY Access_Event_Time = (the time that verification process timed out)
VERIFY Access_Event_Credential = C1
VERIFY Access_Event_Tag = EventTag + 1

-- verify AUTHORIZATION_DELAYED mode (access granted)
4. IF (AUTHORIZATION_DELAYED is supported) THEN
WRITE Authorization_Mode = AUTHORIZATION_DELAYED
MAKE (present credential C1 at credential reader for this access point)
VERIFY Access_Event = AUTHORIZATION_DELAYED
VERIFY Access_Event_Time = (the time that credential C1 was presented)
VERIFY Access_Event_Credential = C1
MAKE (external verification process does not respond within verification time)
WAIT Verification_Time
VERIFY Access_Event = GRANTED
VERIFY Access_Event_Time = (the time that verification process timed out)
VERIFY Access_Event_Credential = C1

-- verify AUTHORIZATION_DELAYED mode (access denied)
WRITE Authorization_Mode = AUTHORIZATION_DELAYED
READ EventTag = Access_Event_Tag
MAKE (present credential C1 at credential reader for this access point)
VERIFY Access_Event = AUTHORIZATION_DELAYED
VERIFY Access_Event_Time = (the time that credential C1 was presented)
VERIFY Access_Event_Credential = C1
MAKE (external verification process denies access)
VERIFY Access_Event = DENIED_VERIFICATION_FAILED
VERIFY Access_Event_Time = (the time that verification process denied access)
VERIFY Access_Event_Credential = C1
VERIFY Access_Event_Tag = EventTag + 1

-- verify NONE mode - *no tests to perform*
~~5. IF (NONE is supported) THEN~~
~~WRITE Authorization_Mode = NONE~~
~~WAIT **Internal Processing Fail Time**~~
~~VERIFY Authentication_Status = DISABLED~~

BTL-26.0 fix2-5: AcknowledgeAlarm Service Initiation Test [BTLWG-1678]

Overview:

In BTL Test Plan 26.0 v2 for Test “8.1.1 AcknowledgeAlarm Service Initiation Test” the chapter “8.1 AcknowledgeAlarm Service Initiation Tests” is used.
The information in the test plan regarding 8.1 could mean that all the tests in this group are to be taken. But that doesn't make much sense either. The tests under 8.1 in the specified tests are: 8.1.1 and 8.1.X2. In the test plan in chapter 5.4.1, 8.1 and 8.1.X2 are to be carried out.

Changes:

Checklist Changes

None

Test Plan Changes

[Reason for change: wrong test number]

5.4.1 Base Requirements

Base requirements must be met by any IUT claiming conformance to this BIBB.

8.1.*1* - AcknowledgeAlarm Service Initiation ~~Tests~~ *Test*

5.4.2 Generates AcknowledgeAlarm Requests with Acknowledge Timestamps of the BACnetDateTime Form

The IUT can generate AcknowledgeAlarm requests with a Time of Acknowledgement of the BACnetDateTime form.

8.1.*1* - AcknowledgeAlarm Service Initiation ~~Tests~~ *Test*

5.4.3 Generates AcknowledgeAlarm Requests with Acknowledge Timestamps of the Time Form

The IUT can generate AcknowledgeAlarm requests with a Time of Acknowledgement of the Time form.

8.1.*1* - AcknowledgeAlarm Service Initiation ~~Tests~~ *Test*

5.4.4 Generates AcknowledgeAlarm Requests with Acknowledge Timestamps of the Sequence Number Form

The IUT can generate AcknowledgeAlarm requests with a Time of Acknowledgement of the Sequence Number form.

8.1.*1* - AcknowledgeAlarm Service Initiation ~~Tests~~ *Test*

5.31.9 Supports AE-ACK-A

The IUT must support AE-ACK-A if it claims support for AE-AC-A.

8.1.*1* - AcknowledgeAlarm Service Initiation ~~Tests~~ *Test*

Specified Test Changes

None

BTL-26.0 fix2-6: Fix Typo in Execute Original Broadcast NPDU Test [BTLWG-1680]

Overview:

Fix a typographical error because the device 'ODIUT' does not exist.

Changes:

Checklist Changes

None

Test Plan Changes

[Change all references to test 12.3.2.2.2 from 135.1 to BTL]

Specified Test Changes

12.3.2.2.2 Execute Original-Broadcast-NPDU (Two-hop Distribution)

Reason for Change: Possibly a typographical error because the device 'ODIUT' does not exist.

Configuration Requirements: The IUT shall be configured with a BDT that contains:

B/IP Address Broadcast Distribution Mask
IUT 255.255.255.255
BBMD1 255.255.255.255

When the IUT is configured for NAT, the Originating-Device in Forwarded-NPDUs that originate at the IUT, OD, is equal to the Global IP Address and Port of the IUT's Internet Router. When the IUT is not configured for NAT operation, OD is equal to the IUT.

Notes to Tester: The order of the messages transmitted by the IUT is not significant.

Test Steps:

1. TRANSMIT
   DA = Local IP Broadcast,
   SA = D1,
   Original-Broadcast-NPDU,
   NPDU = Who-Is
2. RECEIVE
   DA = BBMD1,
   Forwarded-NPDU,
   Originating-Device = D1,
   NPDU = Who-Is
3. IF (the IUT responds with Unicast I-Am) THEN
   RECEIVE DESTINATION = D1,
      Original-Unicast-NPDU,
      NPDU = I-Am
   ELSE
      RECEIVE
         DA = Local IP Broadcast,
         Original-Broadcast-NPDU,
         NPDU = I-Am

RECEIVE
DA=BBMD1,
Forwarded-NPDU,
Originating-Device = ~~OD~~IUT,
NPDU = I-Am

BTL-26.0 fix2-7: Fix Status_Flags Value in Acked_Transitions Test [BTLWG-1693, CR-0576]

Overview:

Why is this change needed -> CR576

Problem:
After generating a fault condition, the status flags shall be observed in step 9. Usually the first flag (In-Alarm) will be 'TRUE' and not 'FALSE'. In similar tests (e.g. 7.3.1.1.1 Out_Of_Service, Status_Flags, and Reliability Test), the first flag is verified to be '?'.

Changes:

Checklist Changes

None

Test Plan Changes

None

Specified Test Changes

7.3.1.11.X2 Acked_Transitions Test for Normal to Fault Transitions

Reason for Change: New test.

Purpose: To verify that the Acked_Transitions property tracks whether or not an acknowledgment has been received for a previously issued fault event notification. It also verifies the interrelationship between Status_Flags and Event_State.

Test Concept: The IUT is configured such that the Event_Enable property indicates that fault event transitions are to trigger an event notification. The Acked_Transitions property shall have the value (?, TRUE, ?). The fault event transition is triggered and the Acked_Transitions property is monitored to verify that the appropriate bit is cleared when a notification message is transmitted and reset when an acknowledgment is received.

Configuration Requirements: The Event_Enable and Acked_Transitions properties shall be configured with a value of (?, TRUE, ?). The referenced event-triggering property shall be set to a value that results in a NORMAL condition. The value of the Transitions parameter for all recipients shall be (?, TRUE, ?).

Notes to Tester: The UnconfirmedEventNotification service may be substituted for the ConfirmedEventNotification service, in which case the TD shall skip sending the BACnet-SimpleACK-PDU messages after receiving the notifications.

Test Steps:

1. VERIFY pCurrentState = NORMAL
2. VERIFY Acked_Transitions = (?, TRUE, ?)
3. IF (Protocol_Revision is present AND Protocol_Revision >= 13) THEN
   VERIFY Status_Flags = (FALSE, FALSE, ?, ?)
4. MAKE (a condition exist that will cause the object to generate a fault condition)
5. BEFORE **Notification Fail Time**
   RECEIVE ConfirmedEventNotification-Request,
   'Process Identifier' = (PI3: any valid process ID),
   'Initiating Device Identifier' = IUT,
   'Event Object Identifier' = (the event-generating object configured for this test),
   'Time Stamp' = (Tfault: any valid time stamp),
   'Notification Class' = (the class corresponding to the object being tested),

'Priority' = (Pfault: the value configured to correspond to a TO-FAULT transition),
'Event Type' = IF (Protocol_Revision < 13) THEN
(any valid event type),
ELSE
CHANGE_OF_RELIABILITY,
'Message Text' = (optional, any valid message text),
'Notify Type' = (the notify type configured for this event),
'AckRequired' = TRUE,
'From State' = NORMAL,
'To State' = FAULT,
'Event Values' = (values appropriate to the event type)

6. TRANSMIT BACnet-SimpleACK-PDU
7. VERIFY pCurrentState = FAULT
8. VERIFY Acked_Transitions = (?, FALSE, ?)
9. IF (Protocol_revision is present AND Protocol_Revision >= 13 THEN
   VERIFY Status_Flags = (~~FALSE~~?, TRUE, ?, ?)
10. TRANSMIT AcknowledgeAlarm-Request,
    'Acknowledging Process Identifier' = (PI3),
    'Event Object Identifier' = (the event-generating object configured for this test),
    'Event State Acknowledged' = FAULT,
    'Acknowledgement Source' = (a character string),
    'Time Stamp' = (Tfault),
    'Time of Acknowledgment' = (the TD's current time)
11. RECEIVE BACnet-SimpleACK-PDU
12. IF (Protocol_Revision is present AND Protocol_Revision ≥ 1) THEN
    BEFORE **Notification Fail Time**
    RECEIVE ConfirmedEventNotification-Request,
    'Process Identifier' = (PI3),
    'Initiating Device Identifier' = IUT,
    'Event Object Identifier' = (the event-generating object configured for this test),
    'Time Stamp' = (Tfault the IUT's current time or sequence number),
    'Notification Class' = (the class corresponding to the object being tested),
    'Priority' = (Pfault),
    'Event Type' = IF (Protocol_Revision < 13)
    (any valid event type),
    ELSE
    CHANGE_OF_RELIABILITY,
    'Message Text' = (optional, any valid message text),
    'Notify Type' = ACK_NOTIFICATION,
    'To State' = FAULT
    ELSE
    BEFORE **Notification Fail Time**
    RECEIVE ConfirmedEventNotification-Request,
    'Process Identifier' = (PI3),
    'Initiating Device Identifier' = IUT,
    'Event Object Identifier' = (the event-generating object configured for this test),
    'Time Stamp' = (Tfault the IUT's current time or sequence number),
    'Notification Class' = (the class corresponding to the object being tested),
    'Priority' = (Pfault),
    'Event Type' = (any valid event type),
    'Message Text' = (optional, any valid message text),
    'Notify Type' = ACK_NOTIFICATION
13. TRANSMIT BACnet-SimpleACK-PDU
14. VERIFY Acked_Transitions = (?, TRUE, ?)