BACnet® TESTING LABORATORIES ADDENDA

Addendum bt to BTL Test Package 20.0.1

Revision V4
Revised 10/27/2022

Approved by the BTL Working Group on 2022-09-22
Approved by the BTL Working Group Voting Members on 2022-10-27.
Published on 2022-10-28.

[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.

BTL-20.0.1 bt-1: Re-alerting CHANGE_OF_LIFE_SAFETY [BTLWG-777] 2

BTL-20.0.1 bt-2: Add Test for LifeSafetyOperation Object [BTLWG-778] 4

In the following document, language to be added to existing clauses within the BTL Test Package 20.0.1 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-20.0.1 bt-1: Re-alerting CHANGE_OF_LIFE_SAFETY [BTLWG-777]

Overview:

The CHANGE_OF_LIFE_SAFETY algorithm was changed to allow transitions from OFFNORMAL -> OFFNORMAL and LIFE_SAFETY -> LIFE_SAFETY whenever the device feels the need to re-alert.

Changes:

Checklist Changes

[Add new option into AE-LS-B Checklist]

Alarm and Event Management - LifeSafety - B		
	R	Base Requirements
	R	Supports the Notification Class Object
	R	Supports AE-INFO-B
	C[1]	Implements intrinsic alarming
	C[1]	Supports the Event Enrollment object
	R	Supports the CHANGE_OF_LIFE_SAFETY Algorithm in Event_Parameters
	C[2]	Supports AE-ACK-B
	C[3]	Generates event notifications with timestamps of the BACnetDateTime form
	C[3]	Generates event notifications with timestamps of the Time form
	C[3]	Generates event notifications with timestamps of the Sequence Number form
	O	Supports mode transition when Event State is maintained
	O	Supports Event_Message_Texts property
	O	Supports Event_Message_Texts_Config property
	O	Supports Re-alerting CHANGE_OF_LIFE_SAFETY events

Test Plan Changes

[In BTL Test Plan, add new entry to AE-LS-B]

5.22.X1 Supports Re-alerting CHANGE_OF_LIFE_SAFETY Events

The IUT supports re-alerting OFFNORMAL or LIFE_SAFETY_ALARM events states for the CHANGE_OF_LIFE_SAFETY algorithm.

BTL - 8.4.8.X1 - Re-alerting CHANGE_OF_LIFE_SAFETY Transitions		
	Test Conditionality	Must be executed.
	Test Directives	Apply for either OFFNORMAL or LIFE_SAFETY_ALARM state for a single event generating object type, as supported by the IUT.
	Testing Hints	

Specified Test Changes

[Add new test which verifies correct re-alerting of a CHANGE_OF_LIFE_SAFETY alarm]

8.4.8.X1 Re-alerting CHANGE_OF_LIFE_SAFETY Transitions

Reason for Change: No test exists for this functionality.

Purpose: This test case verifies the correct operation of the CHANGE_OF_LIFE_SAFETY event algorithm for objects re-alerting OFFNORMAL or LIFE_SAFETY_ALARM event states.

Test Concept: The object begins the test in the NORMAL state. The conditions are made which would cause a transition to a LIFE_SAFETY_ALARM or OFFNORMAL state, E1. Then, the conditions are made which cause the device to re-alert its current state (usually resulting from no alarm acknowledgement or transition away from the current state within some vendor defined time period). If the device does not support re-alerting of OFFNORMAL nor LIFE_SAFETY_ALARM states, then this test shall be skipped.

Configuration Requirements: The IUT shall be configured such that the Event_Enable property has a value of TRUE for the TO-OFFNORMAL, TO-FAULT and TO-NORMAL transitions. The 'Issue_Confirmed_Notifications' parameter shall have a value of TRUE. The event-generating objects shall be in the NORMAL state at the start of the test.

Test Steps:

1. VERIFY Event_State = NORMAL
2. MAKE (the conditions exist which will cause a transition to E1)
3. BEFORE **Notification Fail Time**
   RECEIVE ConfirmedEventNotification-Request,
   'Process Identifier' = (P1: any valid process ID),
   'Initiating Device Identifier' = IUT,
   'Event Object Identifier' = (O1: object being tested),
   'Time Stamp' = (T1: the IUT's current time),
   'Notification Class' = (NC1: the configured notification class),
   'Priority' = (PTY1: the value configured to correspond to a TO-OFFNORMAL transition),
   'Event Type' = CHANGE_OF_LIFE_SAFETY,
   'Message Text' = (S2: optional, any valid message text),
   'Notify Type' = (NT1: the configured notify type),
   'AckRequired' = (AR1: TRUE | FALSE),
   'From State' = NORMAL,
   'To State' = E1,
   'Event Values' = (EVS1: pMonitoredValue, pMode, pStatusFlags, pOperationExpected)
4. MAKE(wait for IUT to send each of its Number_Of_APDU_Retries)
5. MAKE (the conditions exist which will cause the re-alert of the current state E1)
6. BEFORE **Notification Fail Time**
   RECEIVE ConfirmedEventNotification-Request,
   'Process Identifier' = P1,
   'Initiating Device Identifier' = IUT,
   'Event Object Identifier' = O1,
   'Time Stamp' = (T2: any valid time stamp > T1),
   'Notification Class' = NC1,
   'Priority' = PTY1,
   'Event Type' = CHANGE_OF_LIFE_SAFETY,
   'Message Text' = (S2: if provided in the original notification),
   'Notify Type' = NT1,
   'AckRequired' = AR1,
   'From State' = E1,
   'To State' = E1,
   'Event Values' = EVS1
7. TRANSMIT BACnet-SimpleACK-PDU

BTL-20.0.1 bt-2: Add Test for LifeSafetyOperation Object [BTLWG-778]

Overview:

135-2016bt-2. Add specific error codes for LifeSafetyOperation error situations

Changes:

Checklist Changes

None

Test Plan Changes

[In the Test Plan section 5.22.1 add a new test:
BTL - 9.9.2.4 - LifeSafetyOperation on an Object which does not support the operation specified in the 'Request' parameter.]

5.22 Alarm and Event Management - LifeSafety - B

5.22.1 Base Requirements

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

BTL - 9.9.2.4 - LifeSafetyOperation on an Object which does not support the operation specified in the 'Request' parameter.		
	Test Conditionality	If the IUT claims Protocol_Revision 20 or lower, this test shall be skipped. If all Life Safety objects in the IUT support all LifeSafetyOperation request parameter values, this test shall be skipped.
	Test Directives	
	Testing Hints	

Specified Test Changes

[Add a new test into BTL Specified Tests]

9.9.2.4 LifeSafetyOperation on an Object which does not support the operation specified in the 'Request' parameter.

Purpose: To verify that the IUT correctly responds when a LifeSafetyOperation is received that targets an object which does not support the operation specified in the 'Request' parameter.

Test Concept: Send a LifeSafetyOperation request, with an Object Identifier referencing an object in the IUT which does not support the operation specified in the 'Request' parameter.

Test Steps:

1. TRANSMIT LifeSafetyOperation-Request,
   'Requesting Process Identifier' = (any valid value),
   'Requesting Source' = (any valid value),
   'Request' = (any valid value not supported by the referenced Object Identifier),
   'Object Identifier' = (an object in the IUT which does not support the operation specified in the 'Request' parameter.)
2. RECEIVE BACnet-Error PDU,
   'Error Class' = OBJECT,

'Error Code' = VALUE_OUT_OF_RANGE