BACnet® TESTING LABORATORIES ADDENDA

Addendum cr-fix2 to BTL Test Package 26.1

Revision final
Revised 8/11/2026

Approved by the BTL Working Group on June 11, 2026
Approved by the BTL Working Group Voting Members on August 11, 2026;
Published on August 12, 2026.

BACnet is a registered trademark of ASHRAE. ASHRAE does not endorse, approve or test products for compliance with ASHRAE standards. Compliance of listed products to the requirements of ASHRAE Standard 135 is the responsibility of BACnet International. BTL is a registered trademark of BACnet International.

[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-26.1 cr-fix2-1: Fix for Reliability_Evaluation_Inhibit Object Test [BTLWG-1798, CR-0602] 2
BTL-26.1 cr-fix2-2: Allow Some Tests to be Skipped If Effective_Period is Not Writable [BTLWG-1835, CR-0609] 4
BTL-26.1 cr-fix2-3: Fix Distribute_Broadcast_To_Network Test to Not Expect ACK Response [BTLWG-1839, CR-0610] .. 6
BTL-26.1 cr-fix2-4: pTimeDelay Test Updates [BTLWG-1612, CR-0571] 7
BTL-26.1 cr-fix2-5: Fix Test 9.23.1.4 to Reference Correct Object [BTLWG-1843, CR-0611] 12
BTL-26.1 cr-fix2-6: Fix Command_Time_Array Test[BTLWG-1863, CR-0613] 14
BTL-26.1 cr-fix2-7: 7.3.2.46.2 - Network-Number-Is Local Broadcast Only Changes [BTLWG-1870, CR-0614] 15

In the following document, language to be added to existing clauses within the BTL Test Package 26.1 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.

© 2026 by BACnet International. All rights reserved.

BTL-26.1 cr-fix2-1: Fix for Reliability_Evaluation_Inhibit Object Test [BTLWG-1798, CR-0602]

Overview:

7.3.1.21.3 Reliability_Evaluation_Inhibit Object Test
Step 5 correction to handle a condition for objects which do not support 'Event' reporting.

Changes:

Checklist Changes

None

Test Plan Changes

[Change all occurrences of test 7.3.1.21.3 from 135.1-2025 to BTL]

Specified Test Changes

[Move **7.3.1.21.3 Reliability_Evaluation_Inhibit Object Test** from 135.1 to BTL]

[Note: Renumbered test steps to follow new numbering convention]

7.3.1.21.3 Reliability_Evaluation_Inhibit Object Test

Reason for Change: Updating step 5 with condition, so that Devices having objects which do not support Event Reporting should be verified against correct state of Event.

Purpose: To verify that Reliability_Evaluation_Inhibit controls whether or not an object performs the reliability-evaluation process.

Test Concept: Select an object, O1, which supports the Reliability_Evaluation_Inhibit property. With Reliability_Evaluation_Inhibit FALSE, make a fault condition exist. Verify that the Reliability property changes. Set Reliability_Evaluation_Inhibit to TRUE. Verify that the Reliability changes to NO_FAULT_DETECTED. Make a fault condition exist and verify that Reliability property remains NO_FAULT_DETECTED.

Configuration Requirements: The Event_State of O1 is NORMAL, and Reliability_Evaluation_Inhibit equals FALSE.

Test Steps:
1. VERIFY Event_State = NORMAL
2. VERIFY Reliability = NO_FAULT_DETECTED
3. MAKE (a fault condition exist for O1)
4. VERIFY Reliability <> NO_FAULT_DETECTED
~~5. VERIFY Event_State = FAULT~~
5. *IF (event reporting supported) THEN*
6. *VERIFY Event_State = FAULT*
   ELSE
7. *VERIFY Event_State = NORMAL*
8. IF Reliability_Evaluation_Inhibit is writable THEN
9. WRITE Reliability_Evaluation_Inhibit = TRUE
   ELSE
10. MAKE (Reliability_Evaluation_Inhibit TRUE)
11. VERIFY Reliability = NO_FAULT_DETECTED
12. VERIFY Event_State = NORMAL
13. MAKE (remove the fault condition)

© 2026 by BACnet International. All rights reserved.

14. VERIFY Reliability = NO_FAULT_DETECTED
15. MAKE (a fault condition exist for O1)
16. VERIFY Reliability = NO_FAULT_DETECTED
17. VERIFY Event_State = NORMAL

© 2026 by BACnet International. All rights reserved.

BTL-26.1 cr-fix2-2: Allow Some Tests to be Skipped If Effective_Period is Not Writable [BTLWG-1835, CR-0609]

Overview:

Update Test Plan sections 6.4.4 and 6.6.6 to allow tests to be skipped in Effective_Period is not writable. Add a test that checks the vendor's tool allows configurable Effective_Period to contain either a specified date or a fully unspecified date.

Changes:

Checklist Changes

None

Test Plan Changes

6.4 Scheduling - Internal - B

6.4.4 Supports Configurable Effective_Period

The IUT supports the Effective_Period property and it is configurable.

...		
135.1-2025 - 7.2.10 - BACnetDateRange Non-Pattern Properties Test		
	Test Conditionality	~~This test shall only be applied to devices claiming Protocol_Revision 11 or higher.~~ *Must be executed if Effective_Period is writable.*
	Test Directives	Apply to the Effective_Period property.
	Testing Hints	
135.1-2025 - 7.2.11 - BACnetDateRange Open-Ended Pattern Properties Test		
	Test Conditionality	~~This test shall only be applied to devices claiming Protocol_Revision 11 or higher.~~ *Must be executed if Effective_Period is writable.*
	Test Directives	Apply to the Effective_Period property.
	Testing Hints	
135.1-2025 - 9.23.2.22 - BACnetDateRange Non-Pattern Properties Test using WritePropertyMultiple Service		
	Test Conditionality	~~This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports~~ *Must be executed if Effective_Period is writable and* execution of WritePropertyMultiple *is supported*.
	Test Directives	Apply to the Effective_Period property.
	Testing Hints	

6.6 Scheduling - Weekly Schedule - Internal - B

6.6.6 Supports Configurable Effective_Period

The IUT supports the Effective_Period property and it is configurable.

...		
135.1-2025 - 7.2.10 - BACnetDateRange Non-Pattern Properties Test		
	Test Conditionality	~~This test shall only be applied to devices claiming Protocol_Revision 11 or higher.~~ *Must be executed if Effective_Period is writable.*
	Test Directives	Apply to the Effective_Period property.
	Testing Hints	

© 2026 by BACnet International. All rights reserved.

135.1-2025 - 7.2.11 - BACnetDateRange Open-Ended Pattern Properties Test		
	Test Conditionality	~~This test shall only be applied to devices claiming Protocol_Revision 11 or higher.~~ *Must be executed if Effective_Period is writable.*
	Test Directives	Apply to the Effective_Period property.
	Testing Hints	
135.1-2025 - 9.23.2.22 - BACnetDateRange Non-Pattern Properties Test using WritePropertyMultiple Service		
	Test Conditionality	~~This test shall only be applied to devices claiming Protocol_Revision 11 or higher and which supports~~ *Must be executed if Effective_Period is writable and* execution of WritePropertyMultiple *is supported*.
	Test Directives	Apply to the Effective_Period property.
	Testing Hints	

Specified Test Changes

None

© 2026 by BACnet International. All rights reserved.

BTL-26.1 cr-fix2-3: Fix Distribute_Broadcast_To_Network Test to Not Expect ACK Response [BTLWG-1839, CR-0610]

Overview:

135-2024, Clause J.2.1 states " This message provides a mechanism to acknowledge the result of those BVLL service requests that require an acknowledgment, whether successful (ACK) or unsuccessful (NAK). These are: Write-Broadcast-Distribution-Table (ACK, NAK); Read-Broadcast-Distribution-Table (NAK only); Register-Foreign-Device (ACK, NAK); Read-Foreign-Device-Table (NAK only); Delete-Foreign-Device-Table-Entry (ACK, NAK); and Distribute-Broadcast-To-Network (NAK only)."

Changes:

Checklist Changes

None

Test Plan Changes

[Change test 12.3.8.5 in Test Plan Clause 9.3.3 from 135.1 to BTL]

Specified Test Changes

12.3.8.5 Transmits a Broadcast at Startup preceded by Register-Foreign-Device

Reason for Change: Remove BVLC Result response (step 6) from test per BTL-CR-0610.

Purpose: Verify that mode for use of Register-Foreign-Device and setting of 'BBMD Address' parameter are persistent across reset, and that the issuance of Register-Foreign-Device precedes the first issuance of any broadcast, when in that mode.

Test Concept: IUT is put in a mode to use Register-Foreign-Device requests persistently so it will be re-established, then IUT is reset, and the timing of Register-Foreign-Device request to re-establish that precedes the first issuance of any broadcast.

Configuration Requirements: The product's setting of 'BBMD Address' parameter is configured as BBMD1. BBMD1 is the TD simulating a correctly functioning BBMD implementation.

Notes to Tester: For the I-Am, one can precede the Register-Foreign-Device command, as long as then after the Register-Foreign-Device occurs, it is followed by a Distribute-Broadcast-To-Network again, of that I-Am.

Test Steps:

1. MAKE (IUT enter mode for use of Register-Foreign-Device requests, persistently
   so it will be re-established after any reset)
2. MAKE (IUT reset)
3. RECEIVE DA = BBMD1,
   Register-Foreign-Device
4. TRANSMIT BVLC-Result,
   'Result Code' = Successful completion
5. RECEIVE DA = BBMD1,
   Distribute-Broadcast-To-Network,
   NPDU = (any broadcast)
6. ~~TRANSMIT BVLC-Result,~~
   ~~'Result Code' = Successful completion~~

© 2026 by BACnet International. All rights reserved.

BTL-26.1 cr-fix2-4: pTimeDelay Test Updates [BTLWG-1612, CR-0571]

Overview:

Fix 7.3.1.12 based on CR-0571 (BTLWG_1611) to factoring event generation objects that do not support pTimeDelay and pTimeDelayNormal.

BTLWG-1826 will address updating other tests to factor in even generation objects that do not support pTimeDelay and pTimeDelayNormal.

Changes:

Checklist Changes

None

Test Plan Changes

3.22 Event Log Object

3.22.4 Supports BUFFER_READY Notification

The IUT supports the BUFFER_READY notification from the Event Log object.

…		
135.1-2025 - 7.3.1.10.2 - Event_Enable Tests for TO_NORMAL only Algorithms		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	Test Directives	The ‘Event Object Identifier’ in this test must be either an Event Log object contained in the IUT or an Event Enrollment object in the IUT monitoring the Event Log Object contained in the IUT.
	Testing Hints	
~~**135.1-2025**~~***BTL*** **- 7.3.1.12 - Notify_Type Test**		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	Test Directives	The ‘Event Object Identifier’ in this test must be either an Event Log object contained in the IUT or an Event Enrollment object in the IUT monitoring the Event Log Object contained in the IUT.
	Testing Hints	
135.1-2025 - 7.3.2.24.10 - Notification_Threshold Test		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	Test Directives	The ‘Event Object Identifier’ in this test must be either an Event Log object contained in the IUT or an Event Enrollment object in the IUT monitoring the Event Log Object contained in the IUT.
	Testing Hints	
…		

© 2026 by BACnet International. All rights reserved.

5.2 Alarm and Event Management - Notification - Internal - B

5.2.1 Base Requirements

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

135.1-2025 - 7.3.1.10.1 - Event_Enable Test for TO_OFFNORMAL and TO_NORMAL, and TO_FAULT		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	Test Directives	If Event Enrollment objects are supported, ensure this functionality is tested on Event Enrollment objects.
	Testing Hints	The BTL will apply this to a single object. The pretester should apply it to all objects that support alarm generation.
~~135.1-2025~~*BTL* - 7.3.1.12 - Notify_Type Test		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	Test Directives	If Event Enrollment objects are supported, ensure this functionality is also tested on Event Enrollment objects.
	Testing Hints	
135.1-2025 - 8.4 - ConfirmedEventNotification Service Initiation Tests		
	Test Conditionality	Must be executed unless IUT only supports read-only Recipient_List properties and does not claim Notification Forwarder objects. Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications, then this test case shall be satisfied.
	Test Directives	
	Testing Hints	
…		

5.19 Alarm and Event Management - LifeSafety - B

5.19.1 Base Requirements

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

135.1-2025 - 7.3.1.10.1 - Event_Enable Test for TO_OFFNORMAL and TO_NORMAL, and TO_FAULT		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	Test Directives	If Event Enrollment objects are supported, ensure this functionality is tested on Event Enrollment objects.
	Testing Hints	The BTL will apply this to a single object. The pretester should apply it to all objects that support alarm generation.
~~135.1-2025~~*BTL* - 7.3.1.12 - Notify_Type Test		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	Test Directives	If Event Enrollment objects are supported, ensure this functionality is tested on Event Enrollment objects.
	Testing Hints	
135.1-2025 - 8.4 - ConfirmedEventNotification Service Initiation Tests		
	Test Conditionality	Must be executed.

© 2026 by BACnet International. All rights reserved.

		Any of the 8.4 tests can be used to ensure that the IUT properly generates ConfirmedEventNotification requests. The specific tests that can be executed are detailed under the test cases for the specific algorithms. As long as one of the tests is executed using ConfirmedEventNotifications, then this test case shall be satisfied.
	Test Directives	
	Testing Hints	
…		

5.29 Alarm and Event Management - Access Control - B

5.29.1 Base Requirements

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

…		
135.1-2025 - 7.3.1.10.2 - Event_Enable Tests for TO_NORMAL only Algorithms		
	Test Conditionality	Must be executed.
	Test Directives	Apply to an object which generates ACCESS_EVENT notifications.
	Testing Hints	
~~**135.1-2025**~~***BTL*** **- 7.3.1.12 - Notify_Type Test**		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	Test Directives	Apply to an object which generates ACCESS_EVENT notifications.
	Testing Hints	
BTL - 7.3.1.22.1 - Event_Detection_Enable Inhibits Event Generation		
	Test Conditionality	If Protocol_Revision < 13, then this test shall be skipped. If Event_Detection_Enable cannot be set to FALSE, this test shall be skipped.
	Test Directives	Apply to an object which generates ACCESS_EVENT notifications.
	Testing Hints	
…		

7.6 Trending - Automated Trend Retrieval - B

7.6.4 Implements Intrinsic Alarming

The IUT contains, or can be made to contain, a Trend Log object that can generate BUFFER_READY notifications.

Each Trend Log object must include all of the Trend Log properties required for intrinsic reporting, i.e. Notification_Threshold, Records_Since_Notification, Last_Notify_Record, Notification_Class, Event_Enable, Acked_Transitions, Notify_Type, & Event_Time_Stamps.

135.1-2025 - 7.3.1.10.2 - Event_Enable Tests for TO_NORMAL only Algorithms		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	Test Directives	
	Testing Hints	
~~**135.1-2025**~~***BTL*** **- 7.3.1.12 - Notify_Type Test**		
	Test Conditionality	If the IUT cannot be configured to meet the ~~nfiguration~~*configuration* requirements then this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2025 - 7.3.2.24.10 - Notification_Threshold Test		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.

© 2026 by BACnet International. All rights reserved.

	Test Directives	
	Testing Hints	
…		

7.10 Trending - Automated Multiple Value Retrieval - B

7.10.4 Implements Intrinsic Alarming

The IUT contains, or can be made to contain, a Trend Log Multiple object that can generate BUFFER_READY notifications.

Each Trend Log Multiple object must include all of the properties required for intrinsic reporting, i.e. Notification_Threshold, Records_Since_Notification, Last_Notify_Record, Notification_Class, Event_Enable, Acked_Transitions, Notify_Type, & Event_Time_Stamps.

135.1-2025 - 7.3.1.10.2 - Event_Enable Tests for TO_NORMAL only Algorithms		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	Test Directives	
	Testing Hints	
~~**135.1-2025**~~***BTL*** **- 7.3.1.12 - Notify_Type Test**		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	Test Directives	
	Testing Hints	
135.1-2025 - 7.3.2.24.10 - Notification_Threshold Test		
	Test Conditionality	If the IUT cannot be configured to meet the configuration requirements then this test shall be skipped.
	Test Directives	
	Testing Hints	
…		

Specified Test Changes

[Move 7.3.1.12 to BTL Specified Tests]

7.3.1.12 Notify_Type Test

Reason For Change: Add new Notes to Tester to include a note on how to execute the tests with algorithms that do not support pTimeDelay or pTimeDelayNormal. Add a period to the sentence at end of Configuration Requirement's first paragraph. Renumber test steps to new numbering schema.

Purpose: To verify that the value of the Notify_Type property determines whether an event notification is transmitted as an alarm or as an event.

Configuration Requirements: The IUT shall be configured with two event-generation objects, E1 and E2. Object E1 shall be configured with a Notify_Type of ALARM and E2 shall be configured with a Notify_Type of EVENT. ~~The value of the Transitions parameter for all recipients shall be (TRUE, TRUE, TRUE)~~*The value of the Transitions parameter for all recipients shall be (TRUE, TRUE, TRUE).*

In the test description below X1 and X2 are used to designate the pMonitoredValue algorithm parameter linked to E1 and E2 respectively. X1 and X2 shall be set to a value that results in a NORMAL condition of E1 and E2, respectively.

Notes to Tester: For algorithms that do not apply the pTimeDelay or pTimeDelayNormal parameters, assume pTimeDelay or pTimeDelayNormal equals zero when executing the test.

Notes to Tester: If Notify_Type is writable this test may be performed with one event generating object by changing

© 2026 by BACnet International. All rights reserved.

Notify_Type from ALARM to EVENT in order to cover both cases.

Test Steps:

1. VERIFY (E1), pCurrentState = NORMAL
2. VERIFY (E2), pCurrentState = NORMAL
3. WAIT (pTimeDelay + **Notification Fail Time**)
4. IF (X1 is writable) THEN
5. WRITE X1 = (a value that will cause a transition in E1)
ELSE
6. MAKE (X1 a value that will cause a transition in E1)
~~5.~~*7.* IF (the transition is not a FAULT transition) THEN
8. IF (the transition is a TO-OFFNORMAL transition) THEN
9. WAIT (pTimeDelay)
ELSE
10. WAIT (pTimeDelayNormal)
~~6.~~*11.* BEFORE **Notification Fail Time**
12. RECEIVE ConfirmedEventNotification-Request,
'Process Identifier' = (any valid process ID),
'Initiating Device Identifier' = IUT,
'Event Object Identifier' = (E1),
'Time Stamp' = (any valid time stamp),
'Notification Class' = (the class corresponding to the object being tested),
'Priority' = (any valid value),
'Event Type' = (any valid event type),
'Message Text' = (optional, any valid message text),
'Notify Type' = ALARM,
'AckRequired' = TRUE | FALSE,
'From State' = NORMAL,
'To State' = (any valid value),
'Event Values' = (values appropriate to the event type)
~~7.~~*13.* TRANSMIT BACnet-SimpleACK-PDU
~~8.~~*14.* IF (X2 is writable) THEN
15. WRITE X2 = (a value that will cause a transition in E2)
ELSE
16. MAKE (X2 a value that will cause a transition in E2)
~~9.~~*17.* IF (the transition is not a FAULT transition) THEN
18. IF (the transition is a TO-OFFNORMAL transition) THEN
19. WAIT (pTimeDelay)
ELSE
20. WAIT (pTimeDelayNormal)
~~10.~~*21.* BEFORE **Notification Fail Time**
22. RECEIVE ConfirmedEventNotification-Request,
'Process Identifier' = (any valid process ID),
'Initiating Device Identifier' = IUT,
'Event Object Identifier' = (E2),
'Time Stamp' = (any valid time stamp),
'Notification Class' = (the class corresponding to the object being tested),
'Priority' = (any valid value),
'Event Type' = (any valid event type),
'Message Text' = (optional, any valid message text),
'Notify Type' = EVENT,
'AckRequired' = TRUE | FALSE,
'From State' = NORMAL,
'To State' = (any valid value),
'Event Values' = (values appropriate to the event type)
~~11.~~*23.* TRANSMIT BACnet-SimpleACK-PDU

© 2026 by BACnet International. All rights reserved.

BTL-26.1 cr-fix2-5: Fix Test 9.23.1.4 to Reference Correct Object [BTLWG-1843, CR-0611]

Overview:

Fix Test 9.23.1.4.

Changes:

Checklist Changes

None

Test Plan Changes

[Change test 9.23.1.4 in Test Plan Clause 484 from 135.1 to BTL]

Specified Test Changes

[Move test 9.23.1.4 from 135.1 into BTL Specified Tests and modify as shown below]

9.23.1.4 Writing Multiple Properties to Multiple Objects

Reason for Change: Swap which the objects in the VERIFY steps 8 and 9 so the match with the property being checked, per BTL-CR-0611.

Purpose: To verify the ability to write multiple properties to multiple objects.

Test Concept: This test case attempts to write properties, P1 and P2, that reside in Object1, and properties P3 and P4 that reside in Object2. P1, P2, P3 and P4 are not commandable properties. If four such writable properties do not exist the test can be modified to write to an array property or to a commandable property with a write priority high enough to ensure that the commandable property's value will change.

Test Steps:

1. READ X = (Object1), P1
2. READ Y = (Object1), P2
3. READ Z = (Object2), P3
4. READ A = (Object2), P4
5. TRANSMIT WritePropertyMultiple-Request,
   'Object Identifier' = Object1,
   'Property Identifier' = P1,
   'Property Value' = (V1, any valid value of the appropriate datatype for this property subject to the restrictions specified in the EPICS as defined in 4.4.2, except the value X read in step 1)
   ' Property Identifier' = P2,
   'Property Value' = (V2, any valid value of the appropriate datatype for this property subject to the restrictions specified in the EPICS as defined in 4.4.2, except the value Y read in step 2)
   'Object Identifier' = Object2,
   'Property Identifier' = P3,
   'Property Value' = (V3, any valid value of the appropriate datatype for this property subject to the restrictions specified in the EPICS as defined in 4.4.2, except the value Z read in step 3)
   'Object Identifier' = Object2,
   'Property Identifier' = P4,
   'Property Value' = (V4, any valid value of the appropriate datatype for this property subject to

© 2026 by BACnet International. All rights reserved.

the restrictions specified in the EPICS as defined in 4.4.2, except the value A read in step 4)

6. RECEIVE BACnet-BACnet-SimpleACK-PDU
7. VERIFY (Object1), P1 = V1
8. VERIFY (~~Object2~~*Object1*), P2 = V2
9. VERIFY (~~Object1~~*Object2*), P3 = V3
10. VERIFY (Object2), P4 = V4

© 2026 by BACnet International. All rights reserved.

BTL-26.1 cr-fix2-6: Fix Command_Time_Array Test[BTLWG-1863, CR-0613]

Overview:

The value of the Command_Time_Array at PRIO in Step #7 should not have to exactly the Last_Command_Time.

Changes:

Checklist Changes

None

Test Plan Changes

None

Specified Test Changes

[Modify existing test in BTL Specified Tests as shown]

7.3.1.28.3 Value_Source Property None Test

Reason for Change: The other Value Source properties are not tested. The minimum on/off requirement from 19.5.1.3 paragraph 4 is not tested. Allow Command_Time_Array to approximately equal T1.

Purpose: To verify that the Value_Source property shall have the value 'None' when there is no active value source.

Test Concept: If there is no active value source, i.e., the Present_Value has taken on the value of Relinquish_Default, then the Value_Source property shall have the value 'None' *and the other value source properties are updated. If minimum on/off is supported verify the Value_Source_Array at priority 6 is commanded object, O1, and the Command_Time_Array has the present local time.*

Configuration Requirements: The object (O1) to be tested shall have 1 non-NULL entry in its Priority_Array and the Current_Command_Priority has a value other than NULL or 6.

Test Steps:

1. READ PRIO = Current_Command_Priority
2. ~~CHECK(PRIO <> 6 and PRIO <> NULL)~~
2. VERIFY Value_Source = (is not 'None')
3. WRITE Present_Value = NULL, PRIORITY = PRIO
4. VERIFY Last_Command_Time ~= (*T1*, the current local time)
5. *VERIFY Value_Source_Array = TD, ARRAY_INDEX = PRIO*
6. *IF (Command_Time_Array is present) THEN*
7. *VERIFY Command_Time_Array ~= T1, ARRAY_INDEX = PRIO*
8. IF (O1 has Minimum_On_Time or Minimum_Off_Time properties) THEN
9. WAIT the larger of Minimum_Off_Time and Minimum_On_Time
10. *VERIFY Value_Source_Array = O1, ARRAY_INDEX = 6*
11. *IF (Command_Time_Array is present) THEN*
12. *VERIFY Command_Time_Array ~= (the current local time), ARRAY_INDEX = 6*
13. VERIFY Current_Command_Priority = NULL
14. VERIFY Value_Source = 'None' -- the value is the choice 'none'

© 2026 by BACnet International. All rights reserved.

BTL-26.1 cr-fix2-7: 7.3.2.46.2 - Network-Number-Is Local Broadcast Only Changes [BTLWG-1870, CR-0614]

Overview:

Standard 135-2024 requires Network-Number-Is to only be transmitted as a LOCAL_BROADCAST. Test 7.3.2.46.2 allows the TD to initiate a unicast Network-Number-Is and expects the IUT to accept the message.

Changes:

Checklist Changes

None

Test Plan Changes

[Change all occurrences of test 7.3.2.46.2 from 135.1 to BTL. (9.1.12, 9.3.12, 9.4.6, 9.5.6, 9.6.6, 9.7.6, 9.8.10, 9.9.17, 9.10.4)]

Specified Test Changes

[Move 135.1-2025 - 7.3.2.46.2 to Specified tests and modify]

7.3.2.46.2 Network-Number-Is Updates Network_Number_Quality Test

Reason for Change: Removed unicast Network-Number-Is.

Purpose: To verify that Network_Number_Quality is updated when the IUT learns its Network_Number from Network-Number-Is.

Test Concept: Write 0 to Network_ Number to set Network_Number_Quality to UNKNOWN. Send a Network-Number-Is message to the IUT indicating that the Network_Number is learned and verify that Network_Number_Quality changes to LEARNED. Send a Network-Number-Is message to the IUT indicating that the Network_Number is configured and verify that Network_Number_Quality changes to LEARNED_CONFIGURED. Write 0 to Network_ Number and verify that Network_Number_Quality changes to UNKNOWN.

Configuration Requirements: Select a Network Port object, O1, which is enabled and has a writable Network_Number. Connect the TD to the network associated with Network Port O1. This test shall be skipped if the TD cannot be directly connected to the IUT's network.

Test Steps:

-- set network number quality to UNKNOWN
1. WRITE Network_Number = 0
2. TRANSMIT ReinitializeDevice-Request,
   'Reinitialized State of Device' = ACTIVATE_CHANGES,
   'Password' = (any valid password)
3. RECEIVE BACnet-SimpleACK-PDU
4. WAIT **Activate Changes Fail Time**
5. VERIFY Network_Number_Quality = UNKNOWN

-- make IUT learn the network number
6. TRANSMIT Network-Number-Is
   DESTINATION = LOCAL_BROADCAST~~ IUT~~,
   'Network Number' = (N1: any valid value)
   'Flag' = 0 -- learned

© 2026 by BACnet International. All rights reserved.

7. VERIFY Network_Number_Quality = LEARNED
8. VERIFY Network_Number = N1

-- make IUT learn the network number from a configure device
9. TRANSMIT Network-Number-Is
   DESTINATION = LOCAL_BROADCAST~~ | IUT~~,
   'Network Number' = (N2: any valid value)
   'Flag' = 1 -- configured
10. VERIFY Network_Number_Quality = LEARNED_CONFIGURED
11. VERIFY Network_Number = N2

-- configure the IUT's network number *by writing the Network_Number property*
12. WRITE Network_Number = (N3: any valid value other than 0)
13. TRANSMIT ReinitializeDevice-Request,
   'Reinitialized State of Device' = ACTIVATE_CHANGES,
   'Password' = (any valid password)
14. RECEIVE BACnet-SimpleACK-PDU
15. WAIT **Activate Changes Fail Time**
16. VERIFY Network_Number_Quality = CONFIGURED

-- test Network-Number-Is is ignored when Network_Number_Quality is CONFIGURED
17. TRANSMIT Network-Number-Is
   DESTINATION = LOCAL_BROADCAST~~ | IUT~~,
   'Network Number' = (N4: any valid value)
   'Flag' = 1 -- configured
18. VERIFY Network_Number_Quality = CONFIGURED
19. VERIFY Network_Number = N3

-- revert network number quality to UNKNOWN
20. WRITE Network_Number = 0
21. TRANSMIT ReinitializeDevice-Request,
   'Reinitialized State of Device' = ACTIVATE_CHANGES,
   'Password' = (any valid password)
22. RECEIVE BACnet-SimpleACK-PDU
23. WAIT **Activate Changes Fail Time**
24. VERIFY Network_Number_Quality = UNKNOWN

© 2026 by BACnet International. All rights reserved.