Authorization Control Service (ACS)

***Bluetooth*® Test Suite**

- **Revision:** ACS.TS.p1
- **Revision Date:** 2025-02-18
- **Prepared By:** Medical Devices Working Group
- **Published during TCRL:** TCRL.2025-1

This document, regardless of its title or content, is not a Bluetooth Specification as defined in the Bluetooth Patent/Copyright License Agreement ("PCLA") and Bluetooth Trademark License Agreement. Use of this document by members of Bluetooth SIG is governed by the membership and other related agreements between Bluetooth SIG Inc. ("Bluetooth SIG") and its members, including the PCLA and other agreements posted on Bluetooth SIG's website located at www.bluetooth.com.

THIS DOCUMENT IS PROVIDED "AS IS" AND BLUETOOTH SIG, ITS MEMBERS, AND THEIR AFFILIATES MAKE NO REPRESENTATIONS OR WARRANTIES AND DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING ANY WARRANTY OF MERCHANTABILITY, TITLE, NON-INFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, THAT THE CONTENT OF THIS DOCUMENT IS FREE OF ERRORS.

TO THE EXTENT NOT PROHIBITED BY LAW, BLUETOOTH SIG, ITS MEMBERS, AND THEIR AFFILIATES DISCLAIM ALL LIABILITY ARISING OUT OF OR RELATING TO USE OF THIS DOCUMENT AND ANY INFORMATION CONTAINED IN THIS DOCUMENT, INCLUDING LOST REVENUE, PROFITS, DATA OR PROGRAMS, OR BUSINESS INTERRUPTION, OR FOR SPECIAL, INDIRECT, CONSEQUENTIAL, INCIDENTAL OR PUNITIVE DAMAGES, HOWEVER CAUSED AND REGARDLESS OF THE THEORY OF LIABILITY, AND EVEN IF BLUETOOTH SIG, ITS MEMBERS, OR THEIR AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

This document is proprietary to Bluetooth SIG. This document may contain or cover subject matter that is intellectual property of Bluetooth SIG and its members. The furnishing of this document does not grant any license to any intellectual property of Bluetooth SIG or its members.

This document is subject to change without notice.

Contents

1 Scope

This Bluetooth document contains the Test Suite Structure (TSS) and test cases to test the implementation of the Bluetooth Authorization Control Service (ACS) with the objective to provide a high probability of air interface interoperability between the tested implementation and other manufacturers' Bluetooth devices.

2 References, definitions, and abbreviations

2.1 References

This document incorporates provisions from other publications by dated or undated reference. These references are cited at the appropriate places in the text, and the publications are listed hereinafter. Additional definitions and abbreviations can be found in [1], [2], [3], and [4].

[1] Bluetooth Core Specification, Version 4.2 or later

[2] Test Strategy and Terminology Overview

[3] Authorization Control Profile Specification, Version 1.0

[4] Authorization Control Service Specification, Version 1.0

[5] ICS Proforma for Authorization Control Service (ACS)

[6] GATT Test Suite, GATT.TS

[7] Characteristic and Descriptor descriptions are accessible via the Bluetooth SIG Assigned Numbers

[8] IXIT Proforma for Authorization Control Service

2.2 Definitions

In this Bluetooth document, the definitions from [1], [2], [3], and [4] apply.

2.3 Acronyms and abbreviations

In this Bluetooth document, the definitions, acronyms, and abbreviations from [1], [2], [3], and [4] apply.

3 Test Suite Structure (TSS)

3.1 Overview

The Authorization Control Service (ACS) requires the presence of GAP, SM (for LE), SDP (for BR/EDR), and GATT. This is illustrated in Figure 3.1.

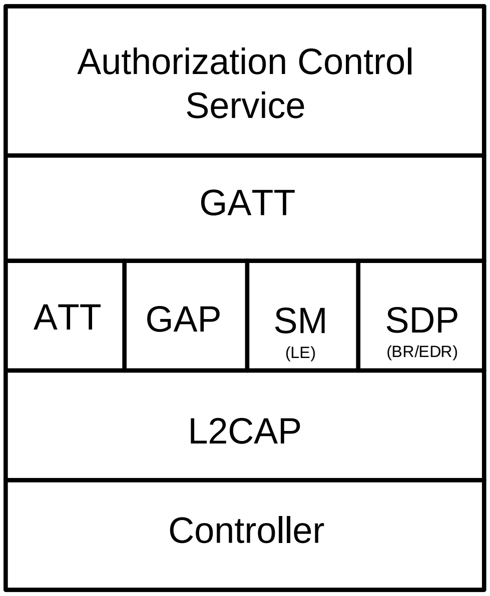

Figure 3.1: Authorization Control Service test model

3.2 Test Strategy

The test objectives are to verify the functionality of the Authorization Control Service within a Bluetooth Host and enable interoperability between Bluetooth Hosts on different devices. The testing approach covers mandatory and optional requirements in the specification and matches these to the support of the IUT as described in the ICS. Any defined test herein is applicable to the IUT if the ICS logical expression defined in the Test Case Mapping Table (TCMT) evaluates to true.

The test equipment provides an implementation of the Radio Controller and the parts of the Host needed to perform the test cases defined in this Test Suite. A Lower Tester acts as the IUT's peer device and interacts with the IUT over-the-air interface. The configuration, including the IUT, needs to implement similar capabilities to communicate with the test equipment. For some test cases, it is necessary to stimulate the IUT from an Upper Tester. In practice, this could be implemented as a special test interface, a Man Machine Interface (MMI), or another interface supported by the IUT.

This Test Suite relies on IXIT values that are described in the respective test cases that make use of the IXIT value.

This Test Suite contains Valid Behavior (BV) tests complemented with Invalid Behavior (BI) tests where required. The test coverage mirrored in the Test Suite Structure is the result of a process that started with catalogued specification requirements that were logically grouped and assessed for testability enabling coverage in defined test purposes.

3.3 Test groups

The following test groups have been defined:

- Generic GATT Integrated Tests
- Characteristic Write
- Characteristic Read
- Authorization Control Point Procedures
- General Error Handling

4 Test cases (TC)

4.1 Introduction

4.1.1 Test case identification conventions

Test cases are assigned unique identifiers per the conventions in [2]. The convention used here is: **<spec abbreviation>/<IUT role>**/<class>/**<feat>**/<func>/<subfunc>**/**<cap>**/<xx>-<nn>-<y>**.

Additionally, testing of this specification includes tests from the GATT Test Suite [6] referred to as Generic GATT Integrated Tests (GGIT); when used, the test cases in GGIT are referred to through a TCID string using the following convention:
<spec abbreviation>/**<IUT role>**/**<GGIT test group>**/< GGIT class >/<xx>-<nn>-<y>.

Identifier Abbreviation	Spec Identifier <spec abbreviation>
ACS	Authorization Control Service
Identifier Abbreviation	**Role Identifier <IUT role>**
SR	Server Role
Identifier Abbreviation	**Reference Identifier <GGIT test group>**
SGGIT	Server Generic GATT Integrated Tests
Identifier Abbreviation	**Reference Identifier <GGIT class>**
CHA	Characteristic
SDP	Validate SDP Record
SER	Service
Identifier Abbreviation	**Feature and Behaviors Identifier <feat>**
ACSCP	Authorization Control Service – Control Point
CR	Characteristic Read
CW	Characteristic Write
GEH	General Error Handling

Table 4.1: ACS TC feature naming conventions

4.1.2 Conformance

When conformance is claimed for a particular specification, all capabilities are to be supported in the specified manner. The mandated tests from this Test Suite depend on the capabilities to which conformance is claimed.

The Bluetooth Qualification Program may employ tests to verify implementation robustness. The level of implementation robustness that is verified varies from one specification to another and may be revised for cause based on interoperability issues found in the market.

Such tests may verify:

- That claimed capabilities may be used in any order and any number of repetitions not excluded by the specification
- That capabilities enabled by the implementations are sustained over durations expected by the use case
- That the implementation gracefully handles any quantity of data expected by the use case

- That in cases where more than one valid interpretation of the specification exists, the implementation complies with at least one interpretation and gracefully handles other interpretations
- That the implementation is immune to attempted security exploits

A single execution of each of the required tests is required to constitute a Pass verdict. However, it is noted that to provide a foundation for interoperability, it is necessary that a qualified implementation consistently and repeatedly pass any of the applicable tests.

In any case, where a member finds an issue with the test plan generated by the Bluetooth SIG qualification tool, with the test case as described in the Test Suite, or with the test system utilized, the member is required to notify the responsible party via an erratum request such that the issue may be addressed.

4.1.3 Pass/Fail verdict conventions

Each test case has an Expected Outcome section. The IUT is granted the Pass verdict when all the detailed pass criteria conditions within the Expected Outcome section are met.

The convention in this Test Suite is that, unless there is a specific set of fail conditions outlined in the test case, the IUT fails the test case as soon as one of the pass criteria conditions cannot be met. If this occurs, the outcome of the test is a Fail verdict.

4.2 Setup preambles

The procedures defined in this section are used to achieve specific conditions on the IUT and the test equipment within the tests defined in this document. The preambles here are commonly used to establish initial conditions.

4.2.1 ATT Bearer on LE transport

- Preamble Procedure

  1. Establish an LE transport connection between the IUT and the Lower Tester.
  2. Establish an L2CAP channel 0x0004 between the IUT and the Lower Tester over that LE transport.

4.2.2 ATT Bearer on BR/EDR transport

- Preamble Procedure

  1. Establish a BR/EDR transport connection between the IUT and the Lower Tester.
  2. Establish an L2CAP channel (PSM 0x001F) between the IUT and the Lower Tester over that BR/EDR transport.

4.2.3 ACS Characteristics and Control Point Configuration

- Preamble Purpose

  This preamble procedure enables the IUT for use with the ACS characteristics and Control Point.

- Preamble Procedure

  1. If a connection exists, it is disconnected.
  2. Establish an ATT Bearer connection between the Lower Tester and the IUT as described in Section 4.2.1, if using an LE transport, or Section 4.2.2 if using a BR/EDR transport.
  3. The handles of the ACS Status, ACS Data In, ACS Data Out Notify, ACS Data Out Indicate, and ACS Control Point, if supported, have been previously discovered by the Lower Tester during the test procedure in Section 4.3 or are known to the Lower Tester by other means.

4. The handles of the Client Characteristic Configuration descriptor of the ACS Status, ACS Data Out Notify, ACS Data Out Indicate, and ACS Control Point have been previously discovered by the Lower Tester during the test procedure in Section 4.3 or are known to the Lower Tester by other means.
5. The ACS Status, ACS Data Out Notify, ACS Data Out Indicate, and ACS Control Point are configured for indication or notification as described in Table 4.2.

Characteristic	Characteristic is configured for
ACS Status	Indication
ACS Data Out Notify	Notification
ACS Data Out Indicate	Indication
ACS Control Point	Indication

Table 4.2: Configuration preamble for ACS characteristics and Control Point

4.3 Generic GATT Integrated Tests

Execute the Generic GATT Integrated Tests defined in Section 6.3, Server test procedures (SGGIT), in [6] using Table 4.3 below as input:

TCID	Service / Characteristic	Reference	Properties	Value Length (Octets)	Type
ACS/SR/SGGIT/SER/BV-01-C [Service GGIT – Authorization Control]	Authorization Control Service	[4] 2	-	-	Primary Service
ACS/SR/SGGIT/CHA/BV-02-C [Characteristic GGIT – ACS Status]	ACS Status Characteristic	[4] 4, 4.2	0x22 (Read, Indicate)	3	-
ACS/SR/SGGIT/CHA/BV-03-C [Characteristic GGIT – ACS Data In]	ACS Data In Characteristic	[4] 4, 4.3	0x08 (Write)	Skip	-
ACS/SR/SGGIT/CHA/BV-04-C [Characteristic GGIT – ACS Data Out Notify]	ACS Data Out Notify Characteristic	[4] 4, 4.3	0x10 (Notify)	Skip	-
ACS/SR/SGGIT/CHA/BV-05-C [Characteristic GGIT – ACS Data Out Indicate]	ACS Data Out Indicate Characteristic	[4] 4, 4.3	0x20 (Indicate)	Skip	-
ACS/SR/SGGIT/CHA/BV-06-C [Characteristic GGIT – ACS Control Point]	ACS Control Point Characteristic	[4] 4, 4.4	0x28 (Write, Indicate)	Skip	-
ACS/SR/SGGIT/SDP/BV-07-C [SDP Record – Authorization Control]	Authorization Control Service	[4] 5	-	-	-

Table 4.3: Input for the GGIT Server test procedure

4.4 Characteristic Write

Verify that the ACS Data In characteristic can be written.

ACS/SR/CW/BV-01-C [Write characteristic value to ACS Data In]

- Test Purpose

  Verify that the ACS Data In characteristic on the IUT can be written.

- Reference

  [4] 4.3, 4.3.2

- Initial Condition

  - Enable the IUT for use with the ACS Control Point, ACS Data Out Notify, and ACS Data Out Indicate characteristics by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get All Active Descriptors procedure using the ACS Control Point characteristic to acquire the current restriction map descriptor, information security controls descriptor, and key descriptor of the IUT.
  - The IUT and the Lower Tester have established the necessary security.

- Test Procedure

  1. The Lower Tester issues an ATT_Write_Request to the ACS Data In characteristic with a Segmentation_Header field and Payload field consisting of the value for a protected resource. The Lower Tester applies the required security controls and includes necessary ACS Data In characteristic header fields.
  2. The Lower Tester receives an ATT_Write_Response indicating that the ATT_Write_Request was successful.

- Expected Outcome

  <u>Pass verdict</u>

  The IUT responds with an ATT_Write_Response indicating that the ATT_Write_Request was successful.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/CW/BV-02-C [Write long characteristic value to ACS Data In]

- Test Purpose

  Verify that the ACS Data In characteristic on the IUT can be written using the Write Long Characteristic Values sub-procedure.

- Reference

  [4] 4.3, 4.3.2

- Initial Condition
  - Enable the IUT for use with the ACS Control Point, ACS Data Out Notify, and ACS Data Out Indicate characteristics by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get All Active Descriptors procedure using the ACS Control Point characteristic to acquire the current restriction map descriptor, information security controls descriptor, and key descriptor of the IUT.
  - The IUT and the Lower Tester have established the necessary security.
- Test Procedure
  1. The Lower Tester writes to the ACS Data In characteristic by executing the GATT Write Long Characteristic Values sub-procedure with a Segmentation_Header field and Payload field consisting of the value for a protected resource. The Lower Tester applies the required security controls and includes necessary ACS Data In characteristic header fields.
  2. For each ATT_Prepare_Write_Request the Lower Tester sends, an ATT_Prepare_Write_Response is sent by the IUT.
  3. After receiving the ATT_Execute_Write_Request, the IUT sends an ATT_Execute_Write_Response.
- Expected Outcome

  Pass verdict

  The IUT sends correctly formatted ATT_Prepare_Write_Responses to the Lower Tester.

  The IUT sends a correctly formatted ATT_Execute_Write_Response to the Lower Tester.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

4.5 Characteristic Read

Verify that a protected resource can be read.

ACS/SR/CR/BV-01-C [Read characteristic with protected resource]

- Test Purpose

  Verify that the ACS Data In characteristic on the IUT can be used to read the value of a protected resource.
- Reference

  [4] 4.3
- Initial Condition
  - Enable the IUT for use with the ACS Control Point and ACS Data Out Notify characteristics by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get All Active Descriptors procedure using the ACS Control Point characteristic to acquire the current restriction map descriptor, information security controls descriptor, and key descriptor of the IUT.
  - The IUT and the Lower Tester have established the necessary security.

- Test Procedure

  1. The Lower Tester issues an ATT_Write_Request to the ACS Data In characteristic with a Segmentation_Header field and Payload field consisting of the value for reading a protected resource. The Lower Tester applies the required security controls and includes necessary ACS Data In characteristic header fields.
  2. The Lower Tester receives an ATT_Write_Response from the IUT.
  3. The Lower Tester receives ATT_Handle_Value_Notification(s), from the IUT, of the ACS Data Out Notify characteristic with the Segmentation_Header field and Payload field containing the value of the protected resource.

- Expected Outcome

  Pass verdict

  The IUT sends ATT_Handle_Value_Notification(s) of the ACS Data Out Notify characteristic with the value of the protected resource.

4.6 Authorization Control Point Procedures

Verify the IUT's ability to perform compliant operations and interpret values of the ACS Control Point characteristic.

Table 4.14 in [4] defines the opcode values used in the ACS Control Point procedure test cases in this section. Section 4.4.4 in [4] defines the operands for each opcode.

ACS/SR/ACSCP/BV-01-C [Get All Active Descriptors without Key Descriptor]

- Test Purpose

  Verify that the IUT can perform the Get All Active Descriptors procedure to report the active unprotected descriptors.

- Reference

  [4] 4.4.3.1

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The active descriptors of the IUT are unprotected.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get All Active Descriptors (0x01) opcode and no operand.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Restriction Map Descriptor Response opcode (0x03) and an operand with one or more Restriction Map records.
  3. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. After one or more Restriction Map records have been provided, the IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Information Security Configuration Descriptor Response opcode (0x0C) and an operand with one or more Information Security Configuration records.

5. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
6. After one or more Information Security Configuration records have been provided, the IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Get All Active Descriptors (0x01) and the Response_Code_Value field set to Success (0x01).
7. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-02-C [Get All Active Descriptors with Key Descriptor]

- Test Purpose

  Verify that the IUT can perform the Get All Active Descriptors procedure to report the active unprotected descriptors with a key descriptor.

- Reference

  [4] 4.4.3.1

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

  - The active descriptors of the IUT are unprotected.

- Test Procedure

1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get All Active Descriptors (0x01) opcode and no operand.
2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Restriction Map Descriptor Response opcode (0x03) and an operand with one or more Restriction Map records.
3. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
4. After one or more Restriction Map records have been provided, the IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Information Security Configuration Descriptor Response opcode (0x0C) and an operand with one or more Information Security Configuration records.
5. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
6. After one or more Information Security Configuration records have been provided, the IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Descriptor Response opcode (0x0E) and an operand with one or more Key records.

7. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
8. After one or more Key records have been provided, the IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Get All Active Descriptors (0x01) and the Response_Code_Value field set to Success (0x01).
9. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-03-C [Get All Active Protected Descriptors without Key Descriptor]

- Test Purpose

  Verify that the IUT can perform the Get All Active Descriptors procedure to report the active protected descriptors without a key descriptor.

- Reference

  [4] 4.4.3.1

- Initial Condition

  - Enable the IUT for use with the ACS Control Point and ACS Data Out Indicate characteristics by performing the preamble described in Section 4.2.3.

  - The Lower Tester requests the Get Restriction Map ID List procedure using the ACS Control Point characteristic to acquire the available restriction map IDs and their mapping to a security configuration ID.

  - The Lower Tester requests the Get Information Security Configuration Descriptor procedure using the ACS Control Point characteristic to acquire the information security configuration controls that are mapped to the restriction map IDs to be requested.

  - The active descriptors of the IUT are protected.

  - The IUT and the Lower Tester have established the necessary security.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Data In characteristic with a Segmentation_Header field and Payload field consisting of the Get All Active Descriptors (0x01) opcode and no operand. The Lower Tester applies the required security controls and includes necessary ACS Data In characteristic header fields.
  2. The IUT responds with the ACS Data Out Indicate characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Restriction Map Descriptor Response opcode (0x03) and an operand with one or more Restriction Map records. The response has the security configuration controls applied and the necessary ACS Data Out Indicate characteristic header fields.
  3. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

4. After one or more Restriction Map records have been provided, the IUT responds with the ACS Data Out Indicate characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Information Security Configuration Descriptor Response opcode (0x0C) and an operand with one or more Information Security Configuration records. The response has the security configuration controls applied and the necessary ACS Data Out Indicate characteristic header fields.
5. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
6. After one or more Information Security Configuration records have been provided, the IUT responds with the ACS Data Out Indicate characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Get All Active Descriptors (0x01) and the Response_Code_Value field set to Success (0x01).
7. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-04-C [Get All Active Protected Descriptors with Key Descriptor]

- Test Purpose

  Verify that the IUT can perform the Get All Active Descriptors procedure to report the active protected descriptors with a key descriptor.

- Reference

  [4] 4.4.3.1

- Initial Condition

  - Enable the IUT for use with the ACS Control Point and ACS Data Out Indicate characteristics by performing the preamble described in Section 4.2.3.

  - The Lower Tester requests the Get Restriction Map ID List procedure using the ACS Control Point characteristic to acquire the available restriction map IDs and their mapping to a security configuration ID.

  - The Lower Tester requests the Get Information Security Configuration Descriptor procedure using the ACS Control Point characteristic to acquire the information security configuration controls that are mapped to the restriction map IDs to be requested.

  - The active descriptors of the IUT are protected.

  - The IUT and the Lower Tester have established the necessary security.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Data In characteristic with a Segmentation_Header field and Payload field consisting of the Get All Active Descriptors (0x01) opcode and no operand. The Lower Tester applies the required security controls and includes necessary ACS Data In characteristic header fields.
  2. The IUT responds with the ACS Data Out Indicate characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Restriction Map Descriptor Response opcode (0x03) and an operand with one or more Restriction Map records. The response has the security configuration controls applied and the necessary ACS Data Out Indicate characteristic header fields.
  3. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. After one or more Restriction Map records have been provided, the IUT responds with the ACS Data Out Indicate characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Information Security Configuration Descriptor Response opcode (0x0C) and an operand with one or more Information Security Configuration records. The response has the security configuration controls applied and the necessary ACS Data Out Indicate characteristic header fields.
  5. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  6. After one or more Information Security Configuration records have been provided, the IUT responds with the ACS Data Out Indicate characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Descriptor Response opcode (0x0E) and an operand with one or more Key records. The response has the security configuration controls applied and the necessary ACS Data Out Indicate characteristic header fields.
  7. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  8. After one or more Key records have been provided, the IUT responds with the ACS Data Out Indicate characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Get All Active Descriptors (0x01) and the Response_Code_Value field set to Success (0x01).
  9. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

4.6.1 Get Restriction Map Descriptor

- Test Purpose

  Verify that, for each selected test case in Table 4.4, the IUT can perform the Get Restriction Map Descriptor procedure for an unprotected descriptor.

- Reference

  [4] 4.4.3.2

- Initial Condition
  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Restriction Map ID List procedure using the ACS Control Point characteristic to acquire the available restriction map IDs and their mapping to a security configuration ID.
  - TSPX_resource_handle_filter_field_value in [8] identifies the value used to filter the restriction map.
- Test Case Configuration

Test Case	Resource Handle Filter value	Response Operand
ACS/SR/ACSCP/BV-05-C [Get Restriction Map Descriptor]	The Resource_Handle_Filter field is set to no filtering (0xFFFF).	An operand with one or more Restriction Map records.
ACS/SR/ACSCP/BV-06-C [Get Restriction Map Descriptor based on Resource Handle Filter]	TSPX_resource_handle_filter_field_value [8].	An operand with the Restriction Map record for the resource handle identified in the request.

Table 4.4: Get Restriction Map Descriptor test cases

- Test Procedure
  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get Restriction Map Descriptor opcode (0x02) and an operand with the Restriction_Map_ID field set to a value of an available, unprotected Restriction Map ID (i.e., mapped to a security configuration ID of 0) and the Resource Handle Filter set to <Resource Handle Filter value> as described in Table 4.4.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Restriction Map Descriptor Response opcode (0x03) and an operand with the <Response Operand> as described in Table 4.4.
  3. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
- Expected Outcome

  Pass verdict

  For each selected test case in Table 4.4, the IUT sends indication(s) of the ACS Control Point characteristic until all records of the Restriction Map, identified by the Resource Handle Filter included in the request, have been sent.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

4.6.2 Get Restriction Map Descriptor – Protected

- Test Purpose

  Verify that, for each selected test case in Table 4.5, the IUT can perform the Get Restriction Map Descriptor procedure when the restriction map is protected.

- Reference

  [4] 4.4.3.2

- Initial Condition

  - Enable the IUT for use with the ACS Control Point and ACS Data Out Indicate characteristics by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Restriction Map ID List procedure using the ACS Control Point characteristic to acquire the available restriction map IDs and their mapping to a security configuration ID.
  - The Lower Tester requests the Get Information Security Configuration Descriptor procedure using the ACS Control Point characteristic to acquire the information security configuration controls that are mapped to the restriction map ID to be requested.
  - The restriction map ID to be requested is protected.
  - TSPX_resource_handle_filter_field_value in [8] identifies the value used to filter the restriction map.
  - The IUT and the Lower Tester have established the necessary security.

- Test Case Configuration

Test Case	Resource Handle Filter value	Response Operand
ACS/SR/ACSCP/BV-07-C [Get Restriction Map Protected Descriptor]	The Resource_Handle_Filter field is set to no filtering (0xFFFF).	An operand with one or more Restriction Map records.
ACS/SR/ACSCP/BV-08-C [Get Restriction Map Protected Descriptor based on Resource Handle Filter]	TSPX_resource_handle_filter_field_value [8].	An operand with the Restriction Map record for the resource handle identified in the request.

Table 4.5: Get Restriction Map Protected Descriptor test cases

- Test Procedure

1. The Lower Tester sends an ATT_Write_Request to the ACS Data In characteristic with a Segmentation_Header field and Payload field consisting of the Get Restriction Map Descriptor opcode (0x02) and an operand with the Restriction_Map_ID field set to a value of an available Restriction Map ID and the Resource Handle Filter set to <Resource Handle Filter value> as described in Table 4.5. The Lower Tester applies the required security controls and includes the necessary ACS Data In characteristic header fields.
2. The IUT responds with the ACS Data Out Indicate characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Restriction Map Descriptor Response opcode (0x03) and an operand with the <Response Operand> as described in Table 4.5. The response has the security configuration controls applied and the necessary ACS Data Out Indicate characteristic header fields.
3. For each indication, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  For each selected test case in Table 4.5, the IUT sends indication(s) of the ACS Data Out Indicate characteristic until all records of the Restriction Map, identified by the Resource Handle Filter included in the request, have been sent.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-09-C [Get Restriction Map ID List]

- Test Purpose

  Verify that the IUT can perform the Get Restriction Map ID List procedure.

- Reference

  [4] 4.4.3.3

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get Restriction Map ID List opcode (0x04) and no operand.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Restriction Map ID List Response opcode (0x05) and an operand with the available Restriction Map IDs each mapped to an Information Security Configuration ID.
  3. For each indication, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends indication(s) of the ACS Control Point characteristic with a listing of Restriction Map IDs each mapped to an Information Security Configuration ID.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-10-C [Activate Restriction Map]

- Test Purpose

  Verify that the IUT can perform the Activate Restriction Map procedure when the restriction map is unprotected.

- Reference

  [4] 4.4.3.4

- Initial Condition
  - Enable the IUT for use with the ACS Status and ACS Control Point characteristics by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Restriction Map ID List procedure using the ACS Control Point characteristic to acquire the available restriction maps to be activated.
  - The restriction map ID to be activated is unprotected (i.e., mapped to a security configuration ID of 0).
- Test Procedure
  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Activate Restriction Map opcode (0x06) and an operand with the Restriction_Map_ID field set to the Restriction Map ID to be activated.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field set to (0x06) followed by the Response_Code_Value field for Success (0x01).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. The Lower Tester sends an ATT_Read_Request to the ACS Status characteristic.
  5. The IUT responds with the ACS Status characteristic in ATT_Read_Response to the Lower Tester.
- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The IUT ACS Status characteristic Current_Restriction_Map_ID field contains the ID sent in the request.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-11-C [Activate Protected Restriction Map]

- Test Purpose

  Verify that the IUT can perform the Activate Restriction Map procedure when the restriction map is protected.

- Reference

  [4] 4.4.3.4

- Initial Condition
  - Enable the IUT for use with the ACS Status, ACS Control Point, and ACS Data Out Indicate characteristics by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Restriction Map ID List procedure using the ACS Control Point characteristic to acquire the available restriction maps to be activated.
  - The Lower Tester requests the Get Information Security Configuration Descriptor procedure using the ACS Control Point characteristic to acquire the information security configuration controls that are mapped to the restriction map to be activated.

- The restriction map ID to be activated is protected.
- The IUT and the Lower Tester have established the necessary security.

- Test Procedure

1. The Lower Tester sends an ATT_Write_Request to the ACS Data In characteristic with a Segmentation_Header field and Payload field consisting of the Activate Restriction Map opcode (0x06) and an operand with the Restriction_Map_ID field set to the Restriction Map ID to be activated. The Lower Tester applies the required security controls and includes the necessary ACS Data In characteristic header fields.
2. The IUT responds with the ACS Data Out Indicate characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field set to (0x06) followed by the Response_Code_Value field for Success (0x01). The response has the security configuration controls applied and the necessary ACS Data Out Indicate characteristic header fields.
3. For each indication, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
4. The Lower Tester sends an ATT_Read_Request to the ACS Status characteristic.
5. The IUT responds with the ACS Status characteristic in ATT_Read_Response to the Lower Tester.

- Expected Outcome

Pass verdict

The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

The IUT ACS Status characteristic Current_Restriction_Map_ID field contains the ID sent in the request.

The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-12-C [Get Resource Handle To UUID Map]

- Test Purpose

Verify that the IUT can perform the Get Resource Handle To UUID Map procedure.

- Reference

[4] 4.4.3.5

- Initial Condition

- Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

- Test Procedure

1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get Resource Handle To UUID Map opcode (0x07) and no operand.
2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Resource Handle To UUID

Map Response opcode (0x08) and an operand with one or more Resource Handle to UUID Map records.

3. For each indication, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-13-C [Get Service And Characteristic UUIDs For Characteristic Resource Handle]

- Test Purpose

  Verify that the IUT can perform the Get Service And Characteristic UUIDs For Characteristic Resource Handle procedure.

- Reference

  [4] 4.4.3.6

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

  - The Lower Tester requests the Get Resource Handle to UUID Map procedure using the ACS Control Point characteristic to acquire a Resource Handle.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get Service And Characteristic UUIDs For Characteristic Resource Handle opcode (0x09) and an operand with the Characteristic_Resource_Handle field value.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Service And Characteristic UUIDs For Characteristic Resource Handle Response opcode (0x0A) and an operand with the Service_UUID_Size and Service_UUID field values, Characteristic_UUID_Size and Characteristic_UUID field values for the requested characteristic Resource Handle.
  3. For each indication, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

4.6.3 Get Information Security Configuration Descriptor

- Test Purpose

  Verify that the IUT can perform the Get Information Security Configuration Descriptor procedure with the <Information Security Configuration ID Filter value> described in Table 4.6.

- Reference

  [4] 4.4.3.7

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - TSPX_information_security_configuration_id_filter_value in [8] identifies the information security configuration on which to filter the descriptor to be reported.

- Test Case Configuration

Test Case	Information Security Configuration ID Filter value	Response Operand
ACS/SR/ACSCP/BV-14-C [Get Information Security Configuration Descriptor]	The Information_Security_ Configuration_ID_Filter field is set to no filtering (0xFFFF).	An operand with one or more Information Security Configuration records.
ACS/SR/ACSCP/BV-15-C [Get Information Security Configuration Descriptor based on filter value]	TSPX_information_security_ configuration_id_filter_value [8].	An operand with the Information Security Configuration record for the Information Security Configuration ID Filter identified in the request.

Table 4.6: Get Information Security Configuration Descriptor test cases

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get Information Security Configuration Descriptor opcode (0x0B) and an operand with the Information Security Configuration ID Filter set to <Information Security Configuration ID Filter value> as described in Table 4.6.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Information Security Configuration Descriptor Response opcode (0x0C) and an operand with the <Response Operand> as described in Table 4.6.
  3. For each indication, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  For each selected test case in Table 4.6, the IUT sends indications(s) of the ACS Control Point characteristic until all records of the Information Security Configuration descriptor, identified by the Information Security Configuration ID Filter included in the request, have been sent.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

4.6.4 Get Key Descriptor procedure

- Test Purpose

  Verify that the IUT can perform the Get Key Descriptor procedure with the <Key ID Filter value> described in Table 4.7.

- Reference

  [4] 4.4.3.8

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - TSPX_key_id_filter_value in [8] identifies the key on which to filter the descriptor to be reported.

- Test Case Configuration

Test Case	Key ID Filter value	Response Operand
ACS/SR/ACSCP/BV-16-C [Get Key Descriptor]	The Key_ID_Filter field is set to no filtering (0xFFFF).	An operand with all available Key records.
ACS/SR/ACSCP/BV-17-C [Get Key Descriptor based on filter value]	TSPX_key_id_filter_value [8].	An operand with the Key record for the Key ID identified in the request.
ACS/SR/ACSCP/BV-18-C [Get Key Descriptor with key format uncompressed plain]	The Key_ID_Filter field is set to no filtering (0xFFFF).	An operand with all available Key records with at least one record using the key format Uncompressed Plain.
ACS/SR/ACSCP/BV-19-C [Get Key Descriptor with key format X.509]	The Key_ID_Filter field is set to no filtering (0xFFFF).	An operand with all available Key records with at least one record using the key format X.509.

Table 4.7: Get Key Descriptor test cases

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get Key Descriptor opcode (0x0D) and an operand with a Key ID filter set to the <Key ID Filter value> as described in Table 4.7.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Descriptor Response opcode (0x0E) and an operand with the <Response Operand> as described in Table 4.7.
  3. For each indication, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  For each selected test case in Table 4.7, the IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-20-C [Get Current Key List]

- Test Purpose

  Verify that the IUT can perform the Get Current Key List procedure.

- Reference

  [4] 4.4.3.9

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get Current Key List opcode (0x0F) and no operand.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Current Key List Response opcode (0x10) and an operand with the Number_Of_Keys_IDs and Key_IDs fields.
  3. For each indication, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  <u>Pass verdict</u>

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-21-C [Invalidate All Established Security]

- Test Purpose

  Verify that the IUT can perform the Invalidate All Established Security procedure when this procedure is unprotected.

- Reference

  [4] 4.4.3.11

- Initial Condition

  - Enable the IUT for use with the ACS Status and ACS Control Point characteristics by performing the preamble described in Section 4.2.3.

  - The IUT has the ACS Status characteristic Status_Flags field Security Established bit set to 1.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Invalidate All Established Security opcode (0x13) and no operand.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code

opcode (0x00) and an operand representing the Request_Opcode field set to (0x13) followed by the Response_Code_Value field for Success (0x01).

3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
4. The IUT responds with the ACS Status characteristic in ATT_Handle_Value_Indication to the Lower Tester.
5. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT ACS Status characteristic Status_Flags field Security Established bit is set to 0.

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-22-C [Invalidate All Established Security for a Protected Resource]

- Test Purpose

  Verify that the IUT can perform the Invalidate All Established Security procedure when this procedure is protected.

- Reference

  [4] 4.4.3.11

- Initial Condition

  - Enable the IUT for use with the ACS Status, ACS Data Out Indicate, and ACS Control Point characteristics by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Restriction Map Descriptor procedure using the ACS Control Point characteristic to acquire the protected resources.
  - The Lower Tester requests the Get Information Security Configuration Descriptor procedure using the ACS Control Point characteristic to acquire the information security configuration controls that are mapped to the restriction map to be activated.
  - The IUT and the Lower Tester have established the necessary security.
  - The IUT has the ACS Status characteristic Status_Flags field Security Established bit set to 1.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Data In characteristic with a Segmentation_Header field and Payload field consisting of the Invalidate All Established Security opcode (0x13) and no operand. The Lower Tester applies the required security controls and includes the necessary ACS Data In characteristic header fields.
  2. The IUT responds with the ACS Data Out Indicate characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field set to (0x13) followed by the Response_Code_Value field for Success (0x01). The response has the security configuration controls applied and the necessary ACS Data Out Indicate characteristic header fields.
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. The IUT responds with the ACS Status characteristic in ATT_Handle_Value_Indication to the Lower Tester.
  5. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The IUT ACS Status characteristic Status_Flags field Security Established bit is set to 0.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

4.6.5 Invalidate Key

- Test Purpose

  Verify that the IUT can perform the Invalidate Key procedure with the <Key ID value> described in Table 4.8.

- Reference

  [4] 4.4.3.12

- Initial Condition

  - Enable the IUT for use with the ACS Status and ACS Control Point characteristics by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Current Key List procedure using the ACS Control Point characteristic to acquire all the key IDs that have been exchanged and/or key IDs that have been generated.

- Test Case Configuration

Test Case	Key ID value
ACS/SR/ACSCP/BV-23-C [Invalidate Key]	The Key_ID field is set to a valid Key ID.
ACS/SR/ACSCP/BV-24-C [Invalidate All Keys]	The Key_ID field is set 0xFFFF.

Table 4.8: Invalidate Key test cases

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Invalidate Key opcode (0x14) and an operand with a Key ID set to the <Key ID value> as described in Table 4.8.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field set to (0x14) followed by the Response_Code_Value field for Success (0x01).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. The IUT responds with the ACS Status characteristic in ATT_Handle_Value_Indication to the Lower Tester.
  5. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  For each selected test case in Table 4.8:

  - The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.
  - The IUT's ACS Status characteristic Status_Flags field Security Established bit is set to 0.
  - The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-25-C [Abort]

- Test Purpose

  Verify that the IUT can perform the Abort procedure.

- Reference

  [4] 4.4.3.13

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get All Active Descriptors opcode (0x01) and no operand.
  2. The IUT starts to send ATT_Handle_Value_Indication(s) of the ACS Control Point characteristic.
  3. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic each with a Segmentation_Header field and Payload field consisting of the Abort opcode (0x15) and no operand.
  4. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to (0x15) followed by the Response_Code_Value field for Success (0x01).
  5. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  6. Verify that the indications stop.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-26-C [Set Security Controls Switch]

- Test Purpose

  Verify that the IUT can perform the Set Security Controls Switch procedure when this procedure is unprotected.

- Reference

  [4] 4.4.3.14

- Initial Condition

  - Enable the IUT for use with the ACS Status and ACS Control Point characteristics by performing the preamble described in Section 4.2.3.

  - The Lower Tester reads the ACS Status characteristic to acquire the current switch state of the IUT.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Set Security Controls Switch opcode (0x16) and an operand with a Switch_State field set to the opposite value of the current switch state value of the IUT.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field set to (0x16) followed by the Response_Code_Value field for Success (0x01).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. The IUT responds with the ACS Status characteristic in ATT_Handle_Value_Indication to the Lower Tester.
  5. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The IUT sets the ACS Status characteristic Status_Flags field Security Controls Switch bit as provided by the Set Security Controls Switch operand Switch_State field.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-27-C [Set Security Controls Switch for a Protected Resource]

- Test Purpose

  Verify that the IUT can perform the Set Security Controls Switch procedure when this procedure is protected.

- Reference

  [4] 4.4.3.14

- Initial Condition

  - Enable the IUT for use with the ACS Status, ACS Data Out Indicate, and ACS Control Point characteristics by performing the preamble described in Section 4.2.3.

  - The Lower Tester requests the Get Restriction Map Descriptor procedure using the ACS Control Point characteristic to acquire the protected resources.

  - The Lower Tester requests the Get Information Security Configuration Descriptor procedure using the ACS Control Point characteristic to acquire the information security configuration controls that are mapped to the restriction map to be activated.
  - The IUT and the Lower Tester have established the necessary security.
  - The Lower Tester reads the ACS Status characteristic to acquire the current Switch State of the IUT.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Data In characteristic with a Segmentation_Header field and Payload field consisting of the Set Security Controls Switch opcode (0x16) and an operand with a Switch_State field set to the opposite value of the current switch value of the IUT. The Lower Tester applies the required security controls and includes necessary ACS Data In characteristic header fields.
  2. The IUT responds with the ACS Data Out Indicate characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field set to (0x16) followed by the Response_Code_Value field for Success (0x01). The response has the security configuration controls applied and the necessary ACS Data Out Indicate characteristic header fields.
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. The IUT responds with the ACS Status characteristic in ATT_Handle_Value_Indication to the Lower Tester.
  5. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The IUT sets the ACS Status characteristic Status_Flags field Security Controls Switch bit as provided by the Set Security Controls Switch operand Switch_State field.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-28-C [Get Key URI]

- Test Purpose

  Verify that the IUT can perform the Get Key URI procedure.

- Reference

  [4] 4.4.3.15

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Key Descriptor procedure using the ACS Control Point characteristic to acquire the key ID of the key with URI.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get Key URI opcode (0x17) and an operand with the Key_ID field.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key URI Response opcode (0x18) and an operand comprising the Key ID used in Step 1, and the Key URI.
  3. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-29-C [Get ACS Feature]

- Test Purpose

  Verify that the IUT can perform the Get ACS Feature procedure.

- Reference

  [4] 4.4.3.16

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get ACS Feature opcode (0x19) and no operand.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the ACS Feature Response opcode (0x1A) and an operand comprising all the mandatory fields.
  3. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-30-C [OOB key exchange]

- Test Purpose

  Verify that the IUT can perform the Key Exchange ECDH procedures with a key ID that references an OOB key.

- Reference

  [4] 4.4.3.17.1

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get ACS Feature procedure using the ACS Control Point characteristic to acquire the IUT’s supported features, protection methods, and OOB capabilities.
  - The OOB key is defined by TSPX_key_oob_value in [8] or the Lower Tester requests the Get Key URI procedure, using the ACS Control Point characteristic, to acquire the key URI for the requested key ID.
  - The Lower Tester requests the Get Key Descriptor procedure using the ACS Control Point characteristic to acquire the key records (e.g., OOB Key Exchange record, the Nonce_Type field value). If the Nonce_Type field value, for the selected key ID in the security algorithm additional parameters structure, is Sequence Number Different Fixed Parts, then the Lower Tester requests the Set AC Client Nonce Fixed procedure using the ACS Control Point characteristic to have a nonce set.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Start Key Exchange opcode (0x11) and an operand with the Key ID, for the OOB key to be exchanged, and indicating how the exchanged key is to be confirmed.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Start Key Exchange (0x11) and the Response_Code_Value field set to Success (0x01).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. The Lower Tester sends ATT_Write_Requests to the ACS Control Point characteristic each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH opcode (0x1B) and an operand with the Key ID used in Step 1, and the AC Client Public Key of the Lower Tester. The exact number of ATT_Write_Requests depends on the payload size.
  5. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Key Exchange ECDH (0x1B) and the Response_Code_Value field set to Success (0x01).
  6. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  7. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Key Exchange KDF opcode (0x21) and an operand with the Key ID used in Step 1.
  8. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange KDF

Response opcode (0x22) and an operand with the Key ID used in Step 1, KDF Salt size and value, and KDF Info size and value.

9. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
10. The Lower Tester and the IUT exchange the OOB random number as described in the start key exchange request.
11. The Lower Tester sends two ATT_Write_Requests to the ACS Control Point characteristic each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH Confirmation Code opcode (0x1D) and an operand with the Key ID used in Step 1, and the AC Client Confirmation Code of the Lower Tester.
12. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange ECDH Confirmation Code Response opcode (0x1E) and an operand with the Key ID used in Step 1, and the AC Server Confirmation Code of the IUT.
13. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
14. The Lower Tester sends two ATT_Write_Requests to the ACS Control Point characteristic each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH Confirmation Random Number opcode (0x1F) and an operand with the Key ID used in Step 1, and the AC Client Confirmation Random Number of the Lower Tester.
15. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange ECDH Confirmation Random Number Response opcode (0x20) and an operand with the Key ID used in Step 1, and the AC Server Confirmation Random Number of the IUT.
16. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
17. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Key Exchange Response opcode (0x12) and an operand with the Key ID used in Step 1, and the key exchange result.
18. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-31-C [ECDH key exchange]

- Test Purpose

  Verify that the IUT can perform the Key Exchange ECDH procedures with a key ID that references an ECDH key.

- Reference

  [4] 4.4.3.17.1

- Initial Condition
  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get ACS Feature procedure using the ACS Control Point characteristic to acquire the IUT’s supported features, protection methods, and OOB capabilities.
  - The Lower Tester requests the Get Key Descriptor procedure using the ACS Control Point characteristic to acquire the key records (e.g., ECDH Key Exchange record, the Nonce_Type field value). If the Nonce_Type field value, for the selected key ID in the security algorithm additional parameters structure, is Sequence Number Different Fixed Parts, then the Lower Tester requests the Set AC Client Nonce Fixed procedure using the ACS Control Point characteristic to have a nonce set.
- Test Procedure
  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Start Key Exchange opcode (0x11) and an operand with the Key ID, for the ECDH key to be exchanged, and indicating how the exchanged key is to be confirmed.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Start Key Exchange (0x11) and the Response_Code_Value field set to Success (0x01).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. The Lower Tester sends ATT_Write_Requests to the ACS Control Point characteristic each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH opcode (0x1B) and an operand with the Key ID used in Step 1, and the AC Client Public Key of the Lower Tester. The exact number of ATT_Write_Requests depends on the payload size.
  5. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange ECDH Response opcode (0x1C) and an operand with the Key ID used in Step 1, and the AC Server Public Key of the IUT.
  6. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  7. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Key Exchange KDF opcode (0x21) and an operand with the Key ID used in Step 1.
  8. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange KDF Response opcode (0x22) and an operand with the Key ID used in Step 1, KDF Salt size and value, and KDF Info size and value.
  9. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  10. The Lower Tester and the IUT exchange the OOB random number as described in the start key exchange request.
  11. The Lower Tester sends two ATT_Write_Requests to the ACS Control Point characteristic each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH Confirmation Code opcode (0x1D) and an operand with the Key ID used in Step 1, and the AC Client Confirmation Code of the Lower Tester.
  12. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange ECDH

Confirmation Code Response opcode (0x1E) and an operand with the Key ID used in Step 1, and the AC Server Confirmation Code of the IUT.

13. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
14. The Lower Tester sends two ATT_Write_Requests to the ACS Control Point characteristic each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH Confirmation Random Number opcode (0x1F) and an operand with the Key ID used in Step 1, and the AC Client Confirmation Random Number of the Lower Tester.
15. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange ECDH Confirmation Random Number Response opcode (0x20) and an operand with the Key ID, and the AC Server Confirmation Random Number of the IUT.
16. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
17. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Key Exchange Response opcode (0x12) and an operand with the Key ID used in Step 1, and the key exchange result.
18. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-32-C [KDF key exchange]

- Test Purpose

  Verify that the IUT can perform a KDF key exchange.

- Reference

  [4] 4.4.3.17.2

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester and the IUT have exchanged the higher-level parent key (e.g., OOB key, ECDH key, KDF key, or manufacturer-specific type) so that the lower-level child key can be derived.
  - The Lower Tester requests the Get Key Descriptor procedure using the ACS Control Point characteristic to acquire the key records (e.g., KDF Key Exchange record, the Nonce_Type field value). If the Nonce_Type field value, for the selected key ID in the security algorithm additional parameters structure, is Sequence Number Different Fixed Parts, then the Lower Tester requests the Set AC Client Nonce Fixed procedure using the ACS Control Point characteristic to have a nonce set.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Start Key Exchange opcode (0x11) and an operand with the Key ID for the KDF to be exchanged, the Selected_Confirmation_Method field set to No Confirmation OOB Method Used, and the Selected_Confirmation_Action field set to 0xFF (i.e., no user action or static confirmation).
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Start Key Exchange (0x11) and the Response_Code_Value field set to Success (0x01).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Key Exchange KDF opcode (0x21) and an operand with the Key ID used in Step 1.
  5. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange KDF Response opcode (0x22) and an operand with the Key ID used in Step 1, KDF Salt size and value, and KDF Info size and value.
  6. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  7. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Key Exchange Response opcode (0x12) and an operand with the Key ID used in Step 1, and the key exchange result.
  8. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-33-C [Set AC Client Nonce Fixed]

- Test Purpose

  Verify that the IUT can perform the Set AC Client Nonce Fixed procedure.

- Reference

  [4] 4.4.3.18

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Key Descriptor procedure using the ACS Control Point characteristic to acquire the key records (e.g., Key_ID and Nonce_Fixed_Size).
  - TSPX_ac_client_nonce_fixed_value in [8] identifies the AC Client Nonce Fixed Value.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Set AC Client Nonce Fixed opcode (0x23) and an operand with the Key ID, and AC Client Nonce Fixed Value as described in the IXIT [8].
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field set to (0x23) followed by the Response_Code_Value field for Success (0x01).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-34-C [Get ATT_MTU]

- Test Purpose

  Verify that the IUT can perform the Get ATT_MTU procedure.

- Reference

  [4] 4.4.3.19

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get ATT_MTU opcode (0xDD) and no operand.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the ATT_MTU Response opcode (0xDE) and an operand with the ATT_MTU size value.
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BV-35-C [Initiate Pairing]

- Test Purpose

  Verify that the IUT can perform the Initiate Pairing procedure.

- Reference

  [4] 4.4.3.20

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Initiate Pairing opcode (0xDF) and no operand.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to (0xDF) followed by the Response_Code_Value field for Success (0x01).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

4.6.6 ACS Control Point – Error Handling

Verify compliant operation of the IUT when an ACS Control Point is written with an invalid operand, or the operand is out of range or the IUT does not support the procedure or other errors specific to the procedure or control point.

ACS/SR/ACSCP/BI-01-C [Opcode not supported]

- Test Purpose

  Verify that the IUT behaves appropriately when it receives an opcode that is not supported for the ACS Control Point characteristic.

- Reference

  [4] 4.4.1, 4.4.5

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

  - The Lower Tester requests the Get ACS Feature procedure using the ACS Control Point characteristic to acquire the IUT’s supported features to determine an opcode that is not supported by the IUT.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of an opcode that is not supported and no operand.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field followed by the Response_Code_Value field for Opcode Not Supported (0x02).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BI-02-C [Set Security Controls Switch with response Invalid Operand]

- Test Purpose

  Verify that the IUT behaves appropriately when it receives a Set Security Controls Switch opcode for the ACS Control Point characteristic with an operand padded with 1s.

- Reference

  [4] 4.4.3.14, 4.4.5

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Set Security Controls Switch opcode (0x16) and an operand with the Switch_State field padded with 1s.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field set to (0x16) followed by the Response_Code_Value field for Invalid Operand (0x03).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BI-03-C [Get Restriction Map Descriptor with response Parameter out of range]

- Test Purpose

  Verify that the IUT behaves appropriately when it receives a Get Restriction Map Descriptor opcode for the ACS Control Point characteristic with a restriction map ID out of range.

- Reference

  [4] 4.4.3.2, 4.4.5

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Restriction Map ID List procedure using the ACS Control Point characteristic to acquire the available restriction map IDs.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get Restriction Map Descriptor opcode (0x02) and an operand with the Restriction_Map_ID field set to a Restriction Map that is not available.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field set to (0x02) followed by the Response_Code_Value field for Parameter Out Of Range (0x05).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BI-04-C [Abort with response Procedure not applicable]

- Test Purpose

  Verify that the IUT behaves appropriately when it receives an Abort opcode for the ACS Control Point characteristic when there is no procedure to abort.

- Reference

  [4] 4.4.3.13, 4.4.5

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Abort opcode (0x15) and no operand.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field set to (0x15) followed by the Response_Code_Value field for Procedure Not Applicable (0x06).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BI-05-C [Get Restriction Map Descriptor with response No records found]

- Test Purpose

  Verify that the IUT behaves appropriately when it receives a Get Restriction Map Descriptor procedure filtered by a Resource Handle that is not available.

- Reference

  [4] 4.4.3.2, 4.4.5

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Restriction Map ID List procedure using the ACS Control Point characteristic to acquire the available restriction map IDs and their mapping to a security configuration ID.
  - The Lower Tester requests the Get Restriction Map Descriptor procedure using the ACS Control Point characteristic to acquire the Resource Handles of protected resources that have a record in the restriction map.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Get Restriction Map Descriptor opcode (0x02) and an operand with the Restriction_Map_ID field set to a value of an available Restriction Map and the Resource_Handle_Filter field set to a Resource Handle value that does not have a record in the corresponding Restriction Map.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field set to (0x02) followed by the Response_Code_Value field for No Records Found (0x08).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BI-06-C [Procedure Already in Progress]

- Test Purpose

  Verify that the IUT responds appropriately when a Client attempts to perform an ACS Control Point procedure before another ACS Control Point procedure is completed.

- Reference

  [4] 4.4.3.1, 4.4.5

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with the Segmentation_Header field and Payload field consisting of the Get All Active Descriptors opcode (0x01) and no operand.
  2. Before the procedure is completed, the Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with the Segmentation_Header field and Payload field containing the Get All Active Descriptors opcode (0x01) and no operand.
  3. The IUT responds with an ATT_Error_Response with Error Code Procedure Already in Progress (0xFE).
  4. Verify that the IUT's response meets the requirements of the service.

- Expected Outcome

  Pass verdict

  The IUT rejects the Write Request from Step 2 and responds with an Attribute Protocol Application Error Code set to Procedure Already in Progress (0xFE).

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BI-07-C [Invalid Public Key]

- Test Purpose

  Verify that the IUT detects and rejects an invalid public key (e.g., public key is not on the curve) during key exchange.

- Reference

  [4] 4.4.3.17.1.1

- Initial Condition
  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get ACS Feature procedure using the ACS Control Point characteristic to acquire the IUT input and output capabilities.
  - The Lower Tester requests the Get Key Descriptor procedure using the ACS Control Point characteristic to acquire the key records (e.g., ECDH Key Exchange record, the Nonce_Type field value). If the Nonce_Type field value, for the selected key ID in the security algorithm additional parameters structure, is Sequence Number Different Fixed Parts, then the Lower Tester requests the Set AC Client Nonce Fixed procedure using the ACS Control Point characteristic to have a nonce set.
- Test Procedure
  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Start Key Exchange opcode (0x11) and an operand with the Key ID, for the ECDH key to be exchanged, and indicating how the exchanged key is to be confirmed.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Start Key Exchange (0x11) and the Response_Code_Value field set to Success (0x01).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. The Lower Tester sends ATT_Write_Requests to the ACS Control Point characteristic each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH opcode (0x1B) and an operand with the Key ID used in Step 1, and an invalid AC Client Public Key. The Lower Tester verifies that this public key is invalid (e.g., not on the curve) before sending it. If the new coordinates happen to be valid, then the generation procedure is repeated.
  5. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to (0x1B) followed by the Response_Code_Value field for Invalid Public Key (0x0A).
  6. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BI-08-C [Invalid key exchange confirmation code]

- Test Purpose

  Verify that the IUT can detect and rejects an invalid confirmation code during key exchange.

- Reference

  [4] 4.4.3.17.1.3

- Initial Condition
  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get ACS Feature procedure using the ACS Control Point characteristic to acquire the IUT input and output capabilities.
  - The Lower Tester requests the Get Key Descriptor procedure using the ACS Control Point characteristic to acquire the key records (e.g., ECDH Key Exchange record, the Nonce_Type field value). If the Nonce_Type field value, for the selected key ID in the security algorithm additional parameters structure, is Sequence Number Different Fixed Parts, then the Lower Tester requests the Set AC Client Nonce Fixed procedure using the ACS Control Point characteristic to have a nonce set.
- Test Procedure
  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Start Key Exchange opcode (0x11) and an operand with the Key ID, for the ECDH key to be exchanged, and indicating how the exchanged key is to be confirmed.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Start Key Exchange (0x11) and the Response_Code_Value field set to Success (0x01).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. The Lower Tester sends ATT_Write_Requests to the ACS Control Point characteristic each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH opcode (0x1B) and an operand with the Key ID used in Step 1, and the AC Client Public Key of the Lower Tester. The exact number of ATT_Write_Requests depends on the payload size.
  5. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange ECDH Response opcode (0x1C) and an operand with the Key ID used in Step 1, and the AC Server Public Key of the IUT.
  6. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  7. The Lower Tester sends ATT_Write_Request(s) to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Key Exchange KDF opcode (0x21) and an operand with the Key ID used in Step 1.
  8. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange KDF Response opcode (0x22) and an operand with the Key ID used in Step 1, KDF Salt size and value, and KDF Info size and value.
  9. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  10. The Lower Tester and the IUT exchange the OOB random number as described in the start key exchange request.
  11. The Lower Tester sends two ATT_Write_Requests to the ACS Control Point characteristic each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH Confirmation Code opcode (0x1D) and an operand with the Key ID used in Step 1, and an invalid AC Client Confirmation Code.
  12. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange ECDH

Confirmation Code Response opcode (0x1E) and an operand with the Key ID used in Step 1, and the AC Server Confirmation Code of the IUT.

13. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
14. The Lower Tester sends two ATT_Write_Requests to the ACS Control Point characteristic, each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH Confirmation Random Number opcode (0x1F) and an operand with the Key ID used in Step 1, and the AC Client Confirmation Random Number of the Lower Tester.
15. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to (0x1F) followed by the Response_Code_Value field for Invalid Key Exchange Confirmation Code (0x09).
16. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

4.6.7 Set AC Client Nonce Fixed – Invalid Behavior

- Test Purpose

  For each selected test case in Table 4.9, verify that the IUT behaves appropriately when it receives a Set AC Client Nonce Fixed procedure with the <Key_ID and AC_Client_Nonce_Fixed_Value> described in Table 4.9.

- Reference

  [4] 4.4.3.18

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Key Descriptor procedure using the ACS Control Point characteristic to acquire the key records (e.g., Key_ID and Nonce_Fixed_Size field values).

- Test Case Configuration

Test Case	Key_ID and AC_Client_Nonce_Fixed_Value	Response
ACS/SR/ACSCP/BI-09-C [Set AC Client Nonce Fixed with response Invalid Operand 1]	TSPX_key_id_value from [8]. AC_Client_Nonce_Fixed_Value: size does not match the Nonce_Fixed_Size field in the security algorithm additional parameters structure.	Invalid Operand (0x03)
ACS/SR/ACSCP/BI-10-C [Set AC Client Nonce Fixed with response Invalid Operand 2]	TSPX_key_id_value from [8]. AC_Client_Nonce_Fixed_Value: value is set equal to another AC_Server_Nonce_Fixed_Value stored on the AC Server.	Invalid Operand (0x03)
ACS/SR/ACSCP/BI-11-C [Set AC Client Nonce Fixed with response Invalid Operand 3]	TSPX_key_id_value from [8]. AC_Client_Nonce_Fixed_Value: value is equal to another AC_Client_Nonce_Fixed_Value stored on the AC Server.	Invalid Operand (0x03)

Table 4.9: Set AC Client Nonce Fixed test cases

- Test Procedure

1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Set AC Client Nonce Fixed opcode (0x23) and an operand with the <Key_ID and AC_Client_Nonce_Fixed_Value> as described in Table 4.9.
2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand representing the Request_Opcode field set to (0x23) followed by the Response_Code_Value field <Response> described in Table 4.9.
3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

Pass verdict

For each selected test case in Table 4.9, the IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BI-12-C [KDF key exchange with response Procedure not applicable]

- Test Purpose

Verify that the IUT can detect a key ID that does not match the key ID provided in the preceding Start Key Exchange procedure, and rejects the Key Exchange KDF procedure.

- Reference

[4] 4.4.3.17.2

- Initial Condition
  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Key Descriptor procedure using the ACS Control Point characteristic to acquire the key records (e.g., KDF Key Exchange record, the Nonce_Type field value). If the Nonce_Type field value, for the selected key ID in the security algorithm additional parameters structure, is Sequence Number Different Fixed Parts, then the Lower Tester requests the Set AC Client Nonce Fixed procedure using the ACS Control Point characteristic to have a nonce set.
- Test Procedure
  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Start Key Exchange opcode (0x11) and an operand with the Key ID, for the KDF to be exchanged, the Selected_Confirmation_Method field set to No Confirmation OOB Method Used, and the Selected_Confirmation_Action field set to 0xFF (i.e., no user action or static confirmation).
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Start Key Exchange (0x11) and the Response_Code_Value field set to Success (0x01).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Key Exchange KDF opcode (0x21) and an operand with a Key ID that is different from the Key ID used in Step 1.
  5. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Key Exchange KDF (0x21) and the Response_Code_Value field set to Procedure Not Applicable (0x06).
  6. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

ACS/SR/ACSCP/BI-13-C [Invalid AC Client confirmation random number]

- Test Purpose

  Verify that the IUT can detect and rejects an invalid AC Client confirmation random number.
- Reference

  [4] 4.4.3.17.1.3

- Initial Condition
  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get ACS Feature procedure using the ACS Control Point characteristic to acquire the IUT’s supported features, protection methods, and OOB capabilities.
  - The Lower Tester requests the Get Key Descriptor procedure using the ACS Control Point characteristic to acquire the key records (e.g., ECDH Key Exchange record, the Nonce_Type field value). If the Nonce_Type field value is Sequence Number Different Fixed Parts, for the selected key ID in the security algorithm additional parameters structure, then the Lower Tester requests the Set AC Client Nonce Fixed procedure using the ACS Control Point characteristic to have a nonce set.
- Test Procedure
  1. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Start Key Exchange opcode (0x11) and an operand with the Key ID, for the ECDH key to be exchanged, and indicating how the exchanged key is to be confirmed.
  2. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to Start Key Exchange (0x11) and the Response_Code_Value field set to Success (0x01).
  3. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  4. The Lower Tester sends ATT_Write_Requests to the ACS Control Point characteristic, each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH opcode (0x1B) and an operand with the Key ID used in Step 1, and the AC Client Public Key of the Lower Tester. The exact number of ATT_Write_Requests depends on the payload size.
  5. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange ECDH Response opcode (0x1C) and an operand with the Key ID used in Step 1, and the AC Server Public Key of the IUT.
  6. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  7. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with a Segmentation_Header field and Payload field consisting of the Key Exchange KDF opcode (0x21) and an operand with the Key ID used in Step 1.
  8. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange KDF Response opcode (0x22) and an operand with the Key ID used in Step 1, KDF Salt size and value, and KDF Info size and value.
  9. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
  10. The Lower Tester and the IUT exchange the OOB random number as described in the start key exchange request.
  11. The Lower Tester sends two ATT_Write_Requests to the ACS Control Point characteristic, each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH Confirmation Code opcode (0x1D) and an operand with the Key ID used in Step 1, and the AC Client Confirmation Code of the Lower Tester.
  12. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Key Exchange ECDH

Confirmation Code Response opcode (0x1E) and an operand with the Key ID used in Step 1, and the AC Server Confirmation Code of the IUT.

13. For each indication sent, verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.
14. The Lower Tester sends two ATT_Write_Requests to the ACS Control Point characteristic, each with a Segmentation_Header field and Payload field to send the Key Exchange ECDH Confirmation Random Number opcode (0x1F) and an operand with the Key ID used in Step 1, and an invalid AC Client Confirmation Random Number for the Lower Tester.
15. The IUT responds with the ACS Control Point characteristic in ATT_Handle_Value_Indication(s) with the Segmentation_Header field and Payload field containing the Response Code opcode (0x00) and an operand with the Request_Opcode field set to (0x1F) followed by the Response_Code_Value field for Invalid Operand (0x03).
16. Verify that the IUT receives an ATT_Handle_Value_Confirmation from the Lower Tester.

- Expected Outcome

  Pass verdict

  The IUT sends all expected indications of the ACS Control Point characteristic with the correct fields.

  The Segmentation_Header field in all requests and responses includes the First Segment bit, Last Segment bit, and Rolling Segment Counter bits with appropriate values.

4.7 General Error Handling

Verify the IUT's error-handling behavior for various scenarios.

ACS/SR/GEH/BI-01-C [Client Characteristic Configuration Descriptor Improperly Configured]

- Test Purpose

  Verify that the IUT responds appropriately when a Client attempts to perform an ACS Control Point procedure with a Client Characteristic Configuration descriptor that is improperly configured.

- Reference

  [4] 4.4.3.16, 4.4.5

- Initial Condition

  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.

- Test Procedure

  1. The Lower Tester sets the Client Characteristic Configuration descriptors of the ACS Control Point characteristics to 0.
  2. The Lower Tester sends an ATT_Write_Request to the ACS Control Point characteristic with the Segmentation_Header field and Payload field containing the Get ACS Feature (0x19) and no operand.
  3. The IUT responds with an ATT_Error_Response with error code Client Characteristic Configuration Descriptor Improperly Configured (0xFD).

- Expected Outcome

  Pass verdict

  The IUT sends an ATT_Error_Response with error code Client Characteristic Configuration Descriptor Improperly Configured (0xFD).

ACS/SR/GEH/BI-02-C [Resource not protected]

- Test Purpose

  Verify that the IUT responds appropriately when a Client attempts to write a request to ACS for a resource that is not protected.

- Reference

  [4] 4.3.2

- Initial Condition

  - Enable the IUT for use with the ACS Control Point, ACS Data Out Notify, and ACS Data Out Indicate characteristics by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Restriction Map Descriptor procedure using the ACS Control Point characteristic to acquire which resources are protected.
  - The IUT and the Lower Tester have established the necessary security.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Data In characteristic with a Segmentation_Header field and Payload field containing a request for a resource that is not protected by ACS.
  2. The IUT responds with an ATT_Error_Response with error code Resource Not Protected (0x81) on the ACS Data In characteristic.

- Expected Outcome

  Pass verdict

  The IUT rejects the Write Request from Step 1 and responds with an Attribute Protocol Application error code set to Resource Not Protected (0x81).

ACS/SR/GEH/BI-03-C [Incorrect security configuration]

- Test Purpose

  Verify that the IUT responds appropriately when a Client attempts to exchange data securely but uses the incorrect security controls for the requested protected resource.

- Reference

  [4] 4.3.2

- Initial Condition

  - Enable the IUT for use with the ACS Control Point, ACS Data Out Notify, and ACS Data Out Indicate characteristics by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Restriction Map Descriptor procedure using the ACS Control Point characteristic to acquire which resources are protected.

  - The Lower Tester requests the Get Information Security Configuration Descriptor procedure using the ACS Control Point characteristic to acquire the mapping between information security configurations and protected resources.
  - The IUT and the Lower Tester have established the necessary security.

- Test Procedure
  1. The Lower Tester sends an ATT_Write_Request to the ACS Data In characteristic with a Segmentation_Header field and Payload field containing a request to a protected resource, but it includes the incorrect information security configuration ID and applicable controls for the requested protected resource.
  2. The IUT responds with an ATT_Error_Response with error code Incorrect Security Configuration (0x82) on the ACS Data In characteristic.

- Expected Outcome

  Pass verdict

  The IUT sends an ATT_Error_Response with error code Incorrect Security Configuration (0x82).

ACS/SR/GEH/BI-04-C [Insufficient Authorization]

- Test Purpose

  Verify that the IUT responds appropriately when a Client attempts to write a request directly to a protected resource.

- Reference

  [4] 3.1, 4.4.1

- Initial Condition
  - Enable the IUT for use with the ACS Control Point characteristic by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Restriction Map Descriptor procedure using the ACS Control Point characteristic to acquire which resources are protected.
  - The Lower Tester requests the Get Information Security Configuration Descriptor procedure using the ACS Control Point characteristic to acquire the mapping between information security configurations and protected resources.

- Test Procedure
  1. The Lower Tester sends an ATT_Write_Request directly to a protected resource (i.e., request not sent to the ACS Data In characteristic).
  2. The IUT responds with an ATT_Error_Response with error code Insufficient Authorization (0x08).

- Expected Outcome

  Pass verdict

  The IUT sends an ATT_Error_Response with error code Insufficient Authorization (0x08).

ACS/SR/GEH/BI-05-C [Invalid Key]

- Test Purpose

  Verify that the IUT responds appropriately when a Client attempts to write to a protected resource with an invalid key.

- Reference

  [4] 4.3.2

- Initial Condition

  - Enable the IUT for use with the ACS Control Point, ACS Data Out Notify, and ACS Data Out Indicate characteristics by performing the preamble described in Section 4.2.3.
  - The Lower Tester requests the Get Restriction Map Descriptor procedure using the ACS Control Point characteristic to acquire which resources are protected.
  - The Lower Tester requests the Get Information Security Configuration Descriptor procedure using the ACS Control Point characteristic to acquire the mapping between information security configurations and protected resources.
  - The IUT and the Lower Tester have established the necessary security.

- Test Procedure

  1. The Lower Tester sends an ATT_Write_Request to the ACS Data In characteristic with a Segmentation_Header field and Payload field containing a request to a protected resource with the correct information security configuration ID and applicable security controls but using a key that has not been exchanged with the IUT.
  2. The IUT responds with an ATT_Error_Response with error code Invalid Key (0x80) on the ACS Data In characteristic.

- Expected Outcome

  Pass verdict

  The IUT sends an ATT_Error_Response with error code Invalid Key (0x80).

ACS/SR/GEH/BI-06-C [Invalid Rolling Segment Counter]

- Test Purpose

  Verify that the IUT responds appropriately when a Client includes an invalid Rolling Segment Counter value in a request.

- Reference

  [4] 4.1.1

- Initial Condition

  - Enable the IUT for use with the ACS Data Out Notify and ACS Data Out Indicate characteristics by performing the preamble described in Section 4.2.3.
  - The IUT and the Lower Tester have established the necessary security.

- Test Procedure

  1. The Lower Tester sends the first of two ATT_Write_Requests to the ACS Data In characteristic with a Segmentation_Header field with a valid Rolling Segment Counter value and Payload field containing a dummy request that needs to be segmented twice.
  2. The IUT responds with an ATT_Write_Response indicating that the ATT_Write_Request was successful.
  3. The Lower Tester sends the second of two ATT_Write_Requests to the ACS Data In characteristic with a Segmentation_Header field with an invalid Rolling Segment Counter value and a Payload field containing the remaining part of the dummy request.
  4. The IUT responds with an ATT_Error_Response with error code Invalid Rolling Segment Counter (0x83) on the ACS Data In characteristic.

- Expected Outcome

  Pass verdict

  The IUT sends an ATT_Error_Response with error code Invalid Rolling Segment Counter (0x83).

4.7.1 Invalid Nonce

- Test Purpose

  For each selected test case in Table 4.10, verify that the IUT responds appropriately when a Client attempts to write a request to ACS with an invalid nonce.

- Reference

  [4] 4.3.2

- Initial Condition

  - Enable the IUT for use with the ACS Control Point, ACS Data Out Notify, and ACS Data Out Indicate characteristics by performing the preamble described in Section 4.2.3.

  - The Lower Tester requests the Get All Active Descriptors procedure using the ACS Control Point characteristic to acquire the current restriction map descriptor, information security controls descriptor, and key descriptor of the IUT.

  - The IUT and the Lower Tester have established the necessary security.

- Test Case Configuration

Test Case
ACS/SR/GEH/BI-07-C [Invalid Nonce with Sequence Number Even-Odd]
ACS/SR/GEH/BI-08-C [Invalid Nonce with Sequence Number Different Fixed Parts]

Table 4.10: Invalid Nonce test cases

- Test Procedure

  1. The Lower Tester issues an ATT_Write_Request to the ACS Data In characteristic with a Segmentation_Header field and Payload field consisting of the value for a protected resource. The Lower Tester applies the required security controls and includes necessary ACS Data In characteristic header fields with an invalid nonce.
  2. The IUT responds with an ATT_Error_Response with Incorrect Security Configuration (0x82).

- Expected Outcome

  Pass verdict

  The IUT sends an ATT_Error_Response with Incorrect Security Configuration (0x82).

5 Test case mapping

The Test Case Mapping Table (TCMT) maps test cases to specific requirements in the ICS. The IUT is tested in all roles for which support is declared in the ICS document.

The columns for the TCMT are defined as follows:

Item: Contains a logical expression based on specific entries from the associated ICS document. Contains a logical expression (using the operators AND, OR, NOT as needed) based on specific entries from the applicable ICS document(s). The entries are in the form of y/x references, where y corresponds to the table number and x corresponds to the feature number as defined in the ICS document for Authorization Control Service [5].

If a test case is mandatory within the respective layer, then the y/x reference is omitted.

Feature: A brief, informal description of the feature being tested.

Test Case(s): The applicable test case identifiers are required for Bluetooth Qualification if the corresponding y/x references defined in the Item column are supported. Further details about the function of the TCMT are elaborated in [2].

For the purpose and structure of the ICS/IXIT, refer to [2].

Item	Feature	Test Case(s)
ACS 2/1 AND ACS 6/1	SDP Record	ACS/SR/SGGIT/SDP/BV-07-C
ACS 6/1	ACS – Service definition	ACS/SR/SGGIT/SER/BV-01-C
ACS 6/2	ACS Status	ACS/SR/SGGIT/CHA/BV-02-C
ACS 6/3	ACS Data In	ACS/SR/SGGIT/CHA/BV-03-C ACS/SR/CW/BV-01-C ACS/SR/GEH/BI-06-C
ACS 3/22 AND ACS 6/3	Invalid Nonce with Sequence Number Even-Odd	ACS/SR/GEH/BI-07-C
ACS 3/23 AND ACS 6/3	Invalid Nonce with Sequence Number Different Fixed Parts	ACS/SR/GEH/BI-08-C
ACS 6/3 AND ACS 8/5	Write long characteristic value to ACS Data In	ACS/SR/CW/BV-02-C
ACS 6/3 AND ACS 6/4	Read characteristic with protected resource	ACS/SR/CR/BV-01-C
ACS 6/3 AND ACS 3/2	Access to Protected Resource	ACS/SR/GEH/BI-02-C ACS/SR/GEH/BI-03-C ACS/SR/GEH/BI-05-C
ACS 6/4	ACS Data Out Notify	ACS/SR/SGGIT/CHA/BV-04-C
ACS 6/5	ACS Data Out Indicate	ACS/SR/SGGIT/CHA/BV-05-C
ACS 6/6	ACS Control Point	ACS/SR/SGGIT/CHA/BV-06-C ACS/SR/ACSCP/BI-01-C ACS/SR/GEH/BI-01-C
ACS 3/2 AND NOT ACS 7/8	Get All Active Descriptors without Key Descriptor	ACS/SR/ACSCP/BV-01-C
ACS 3/2 AND ACS 7/8	Get All Active Descriptors with key descriptor	ACS/SR/ACSCP/BV-02-C

Item	Feature	Test Case(s)
ACS 3/2 AND ACS 6/3 AND ACS 6/5 AND (NOT ACS 7/8)	Get All Active Descriptors without Key Descriptor – Protected	ACS/SR/ACSCP/BV-03-C
ACS 3/2 AND ACS 6/3 AND ACS 6/5 AND ACS 7/8	Get All Active Descriptors with Key Descriptor – Protected	ACS/SR/ACSCP/BV-04-C
ACS 3/2	Unprotected Descriptor	ACS/SR/ACSCP/BV-05-C ACS/SR/ACSCP/BV-06-C ACS/SR/ACSCP/BV-09-C ACS/SR/ACSCP/BV-14-C ACS/SR/ACSCP/BV-15-C ACS/SR/ACSCP/BI-03-C ACS/SR/ACSCP/BI-05-C ACS/SR/ACSCP/BI-06-C ACS/SR/GEH/BI-04-C
ACS 3/2 AND ACS 6/3 AND ACS 6/5	Protected Descriptor	ACS/SR/ACSCP/BV-07-C ACS/SR/ACSCP/BV-08-C
ACS 7/4	Activate Restriction Map	ACS/SR/ACSCP/BV-10-C
ACS 6/3 AND ACS 6/5 AND ACS 7/4	Activate Protected Restriction Map	ACS/SR/ACSCP/BV-11-C
ACS 7/5	Get Resource Handle To UUID Map	ACS/SR/ACSCP/BV-12-C
ACS 7/6	Get Service And Characteristic UUIDs For Characteristic Resource Handle	ACS/SR/ACSCP/BV-13-C
ACS 7/8	Get Key Descriptor	ACS/SR/ACSCP/BV-16-C ACS/SR/ACSCP/BV-17-C
ACS 7/8 AND ACS 3/18 OR ACS 3/19	Get Key Descriptor with key format uncompressed plain	ACS/SR/ACSCP/BV-18-C
ACS 7/8 AND ACS 3/20 OR ACS 3/21	Get Key Descriptor with key format X.509	ACS/SR/ACSCP/BV-19-C
ACS 7/9	Get Current Key List	ACS/SR/ACSCP/BV-20-C
ACS 7/11	Invalidate All Established Security	ACS/SR/ACSCP/BV-21-C
ACS 6/3 AND ACS 6/5 AND ACS 7/11	Invalidate All Established Security for a Protected Resource	ACS/SR/ACSCP/BV-22-C
ACS 7/12	Invalidate Key	ACS/SR/ACSCP/BV-23-C ACS/SR/ACSCP/BV-24-C
ACS 7/13	Abort	ACS/SR/ACSCP/BV-25-C ACS/SR/ACSCP/BI-04-C
ACS 7/14	Set Security Controls Switch	ACS/SR/ACSCP/BV-26-C ACS/SR/ACSCP/BI-02-C
ACS 6/3 AND ACS 6/5 AND ACS 7/14	Set Security Controls Switch for a Protected Resource	ACS/SR/ACSCP/BV-27-C
ACS 7/15	Get Key URI	ACS/SR/ACSCP/BV-28-C
ACS 7/16	Get ACS Feature	ACS/SR/ACSCP/BV-29-C

Item	Feature	Test Case(s)
ACS 3/7 AND ACS 3/8 AND (ACS 5/1 OR ACS 5/2 OR ACS 5/3 OR ACS 5/4 OR ACS 5/5)	ECDH key exchange	ACS/SR/ACSCP/BV-31-C ACS/SR/ACSCP/BI-07-C ACS/SR/ACSCP/BI-08-C ACS/SR/ACSCP/BI-13-C
ACS 3/6 AND ACS 3/7 AND ACS 3/8 AND (ACS 5/1 OR ACS 5/2 OR ACS 5/3 OR ACS 5/4 OR ACS 5/5 OR ACS 5/6)	OOB key exchange	ACS/SR/ACSCP/BV-30-C
ACS 3/8	KDF key exchange	ACS/SR/ACSCP/BV-32-C ACS/SR/ACSCP/BI-12-C
ACS 7/21	Get ATT_MTU	ACS/SR/ACSCP/BV-34-C
ACS 7/22	Initiate Pairing	ACS/SR/ACSCP/BV-35-C
ACS 7/20	Set AC Client Nonce Fixed	ACS/SR/ACSCP/BV-33-C ACS/SR/ACSCP/BI-09-C ACS/SR/ACSCP/BI-10-C ACS/SR/ACSCP/BI-11-C

Table 5.1: Test case mapping

6 ACS Control Point Response Code Test Matrix

The following table summarizes the combination of some of the ACS Control Point opcodes and the Response Code values that are tested and not tested. For the table below, the following key applies:

YES = A test for this combination exists.
NO = A test for this combination does not exist.
N/A = Not a valid combination.

ACS Control Point Response Code	**ACS Contol Point Opcode**				
	Set Security Controls Switch	Abort	Get Restriction Map Descriptor	Key Exchange ECDH	Other
Success	YES	YES	N/A	N/A	N/A
Opcode Not Supported	N/A	N/A	N/A	N/A	YES
Invalid Operand	YES	N/A	NO	NO	N/A
Procedure Not Completed	NO	NO	NO	NO	N/A
Parameter Out Of Range	N/A	N/A	YES	NO	N/A
Procedure Not Applicable	NO	YES	NO	NO	N/A
Abort Unsuccessful	N/A	NO	N/A	N/A	N/A
No Records Found	N/A	N/A	YES	N/A	N/A
Invalid Key Exchange Confirmation Code	N/A	N/A	N/A	YES	N/A
Invalid Public Key	N/A	N/A	N/A	YES	N/A

Table 6.1: ACS Control Point Response Code test coverage

7 ACS Data Error Code Test Matrix

The following table summarizes the combination of some of the ACS Data error codes that are tested and not tested. For the table below, the following key applies:

YES = A test for this combination exists.
NO = A test for this combination does not exist.
N/A = Not a valid combination.

Error Codes	ACS Data characteristic
Invalid Key	YES
Resource not protected	YES
Incorrect security configuration	YES
Invalid Rolling Segment Counter	YES

Table 7.1: ACS Data Error Code test coverage

8 Revision history and acknowledgments

Revision History

Publication Number	Revision Number	Date	Comments
0	p0	2022-09-20	Approved by BTI on 2022-08-31. ACS v1.0 adopted by the BoD on 2022-09-13. Prepared for initial publication.
	p0ed2r00	2023-03-06 – 2023-03-08	TSE 22485 (rating 1): Updated the test procedure and expected outcome in ACS/SR/ACSCP/BV-01-C – -04-C, -10-C – -13-C, -16-C – -35-C and ACS/SR/ACSCP/BI-01-C – -05-C and -07-C – -13-C. Updated the test procedure in ACS/SR/ACSCP/ BV-05-C, -09-C, -14-C, and -15-C. Editorials to align the document with the latest ICS template.
	p0ed2r01	2023-04-13	Incorporated integration review feedback from A. Courtney.
	p0 edition 2	2023-04-14	Approved by BTI on 2023-04-13. Prepared for edition 2 publication.
	p1r00–r01	2024-10-01 – 2024-11-04	TSE 25223 (rating 2): Updated the TCMT entry for KDF key exchange from item ACS 7/23 to item ACS 3/8. TSE 26094 (rating 2): Updated the TCMT entries for TCs ACS/SR/ACSCP/BV-30-C and ACS/SR/ACSCP/BV-31-C to include the new ICS items ACS 5/5 and ACS 5/6. Performed editorial work to align with the current TS template, including updates to the conformance section and the text preceding the TCMT.
1	p1	2025-02-18	Approved by BTI on 2024-12-23. Prepared for TCRL 2025-1 publication.

Acknowledgments

Name	Company
Jörg Brakensiek	Bluetooth SIG, Inc.
Ismail Mohamud	Bluetooth SIG, Inc.
Christoph Fischer	F. Hoffmann-La Roche AG
Nathaniel Hamming	HMT Consulting