ONVIF®

Uplink Device Test Specification

Version 23.06

June 2023

REVISION HISTORY

Vers.	Date	Description
23.06	Jan 04, 2023	Initial version

Table of Contents

1 Introduction

The goal of the ONVIF test specification set is to make it possible to realize fully interoperable IP physical security implementation from different vendors. The set of ONVIF test specification describes the test cases need to verify the [ONVIF Uplink Specification] and [ONVIF Conformance] requirements. It also describes the test framework, test setup, pre-requisites, test policies needed for the execution of the described test cases.

This ONVIF Uplink Test Specification acts as a supplementary document to the [ONVIF Uplink Specification], illustrating test cases need to be executed and passed. And also this specification acts as an input document to the development of test tool which will be used to test the ONVIF device implementation conformance towards ONVIF standard. This test tool is referred as ONVIF Client hereafter.

1.1 Scope

This ONVIF Uplink Test Specification defines and regulates the conformance testing procedure for the ONVIF conformant devices. Conformance testing is meant to be functional black-box testing. The objective of this specification is to provide test cases to test individual requirements of ONVIF devices according to ONVIF Uplink which is defined in [ONVIF Uplink Specification].

The principal intended purposes are:

- Provide self-assessment tool for implementations.
- Provide comprehensive test suite coverage for [ONVIF Network Interface Specs].

This specification **does not** address the following:

- Product use cases and non-functional (performance and regression) testing.
- SOAP Implementation Interoperability test i.e. Web Service Interoperability Basic Profile version 2.0 (WS-I BP 2.0).
- Network protocol implementation Conformance test for HTTP, HTTPS, RTP and RTSP protocol.
- Poor streaming performance test (audio/video distortions, missing audio/video frames, incorrect lib synchronization etc.).

  Wi-Fi Conformance test

The set of ONVIF Test Specification will not cover the complete set of requirements as defined in [ONVIF Uplink Specification]; instead it would cover subset of it. The scope of this specification is to

derive all the normative requirements of [ONVIF Uplink Specification] which are related to ONVIF Uplink and some of the optional requirements.

This ONVIF Uplink Test Specification covers Uplink service which is a functional block of [ONVIF Network Interface Specs]. The following sections give the brief overview of and scope of each functional block.

1.2 Connection

The Connection section covers the test cases needed for initiating an uplink from a device to a service endpoint. Once the connection is established the service endpoint acts as client.

The scope of this specification section is to cover the following functions:

- Setting device uplink configuration.
- Connection establishment and teardown.
- Reconnecting on communication timeout.
- Verification of authentication and user level.
- Responding to requests.
- Streaming of video.

1.3 Capabilities

The Capabilities section covers the test cases needed for getting capabilities from an ONVIF device.

The scope of this specification section is to cover the following functions:

- Getting Uplink service address with GetServices command via Device service
- Getting capabilities with GetServiceCapabilities command
- Getting capabilities with GetServices command via Device service

2 Normative references

- [ONVIF Conformance] ONVIF Conformance Process Specification:

  https://www.onvif.org/profiles/conformance/

- [ONVIF Profile Policy] ONVIF Profile Policy:

  https://www.onvif.org/profiles/

- [ONVIF Network Interface Specs] ONVIF Network Interface Specification documents:

  https://www.onvif.org/profiles/specifications/

- [ONVIF Core Specs] ONVIF Core Specifications:

  https://www.onvif.org/profiles/specifications/

- [ONVIF Uplink Specification] Uplink Specification

  https://www.onvif.org/specs/srv/uplink/ONVIF-Uplink-Spec.pdf

- [ONVIF Base Test] ONVIF Base Device Test Specification:

  https://www.onvif.org/profiles/conformance/device-test/

- [ONVIF RTSP via Media2 Test] Real Time Streaming using Media2 Test Specifcation

  > https://www.onvif.org/wp-content/uploads/2021/06/ONVIF_Real_Time_Streaming_using_Media2_Device_Test_Specification_21.06.pdf

- [ISO/IEC Directives, Part 2] ISO/IEC Directives, Part 2, Annex H:

  http://www.iso.org/directives

- [ISO 16484-5] ISO 16484-5:2014-09 Annex P:

  https://www.iso.org/obp/ui/#!iso:std:63753:en

- [SOAP 1.2, Part 1] W3C SOAP 1.2, Part 1, Messaging Framework:

  http://www.w3.org/TR/soap12-part1/

- [XML-Schema, Part 1] W3C XML Schema Part 1: Structures Second Edition:

  http://www.w3.org/TR/xmlschema-1/

- [XML-Schema, Part 2] W3C XML Schema Part 2: Datatypes Second Edition:

http://www.w3.org/TR/xmlschema-2/

- [WS-Security] "Web Services Security: SOAP Message Security 1.1 (WS-Security 2004)", OASIS Standard, February 2006.:

  http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf

3 Terms and Definitions

3.1 Conventions

The key words "shall", "shall not", "should", "should not", "may", "need not", "can", "cannot" in this specification are to be interpreted as described in [ISO/IEC Directives Part 2].

3.2 Definitions

This section defines terms that are specific to the ONVIF Uplink Service and tests. For the list of applicable general terms and definitions, please see [ONVIF Base Test].

Uplink Doing something in advance to prepare for something else.

Video Source Entity defined by [ONVIF Device I/O Specification]

Video Source Token Token referencing a Device I/O Video Source

3.3 Abbreviations

This section describes abbreviations used in this document.

HTTP Hyper Text Transport Protocol.

AAC Advanced Audio Coding.

URI Uniform Resource Identifier.

WSDL Web Services Description Language.

XML eXtensible Markup Language.

JPEG Joint Photographic Experts Group.

TTL Time To Live.

4 Test Overview

This section provides information about the test setup procedure and required prerequisites, and the test policies that should be followed for test case execution.

4.1 Test Setup

4.1.1 Network Configuration for DUT

The generic test configuration for the execution of test cases defined in this document is as shown below (Figure 1).

Figure 4.1. Test Configuration for DUT

DUT: ONVIF device to be tested. Hereafter, this is referred to as DUT (Device Under Test).

ONVIF Client (Test Tool): Tests are executed by this system and it controls the behavior of the DUT. It handles both expected and unexpected behavior.

Service Endpoint: Cloud connection endpoint according to the ONVIF Uplink Specification to which the DUT can connect.

4.2 Prerequisites

The pre-requisites for executing the test cases described in this Test Specification are:

1. The DUT shall be configured with an IPv4 address.
2. The DUT shall be IP reachable [in the test configuration].
3. The DUT shall be able to be discovered by the Test Tool.
4. The DUT shall be configured with the time, i.e. manual configuration of UTC time and if NTP is supported by the DUT then NTP time shall be synchronized with NTP Server.
5. The DUT time and Test tool time shall be synchronized with each other either manually or by a common NTP server.

4.3 Test Policy

This section describes the test policies specific to the test case execution of each functional block.

The DUT shall adhere to the test policies defined in this section.

4.3.1 Uplink

The test policies specific to the test case execution of all functional blocks:

- DUT shall give the Uplink Service entry point by GetServices command, if DUT supports this service. Otherwise, these test cases will be skipped.
- DUT shall support the following commands:
  - GetUplinks
  - SetUplink
  - DeleteUplink
  - GetDeviceInformation (Device Mgmt Service)
  - GetUsers (Device Mgmt Service)
- A DUT supporting the Media2 service shall support the following commands:
  - GetProfiles (Media2)
  - GetStreamUri (Media2)

Please refer to Section 5.1 for Uplink Test Cases.

4.3.2 Capabilities

The test policies specific to the test case execution of Capabilities functional block:

- DUT shall give the Uplink Service entry point by GetServices command, if DUT supports this service. Otherwise, these test cases will be skipped.
- DUT shall support the following commands:
  - GetServices
  - GetServiceCapabilities
- The following tests are performed
  - Getting capabilities with GetServiceCapabilities command
  - Getting capabilities with GetServices command

Please refer to Section 5.2 for Uplink Test Cases.

5 Uplink Test Cases

5.1 Uplink Connection

5.1.1 CONNECT AND DISCONNECT

Test Case ID: UPLINK-1-1-1

Specification Coverage: GetUplinks (Uplink Specification), SetUplink (Uplink Specification), DeleteUplink (Uplink Specification)

Feature Under Test: Connect, Disconnect

WSDL Reference: uplink.wsdl

Test Purpose: To verify conect and disconnect operation.

Pre-Requisite: Uplink Service is received from the DUT.

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start the DUT.
3. ONVIF Client retrieves Uplink Service capabilities by following the procedure mentioned in Annex A.1 with the following input and output parameters
   - out *cap* - Uplink Service capabilities
4. ONVIF Client removes any existing uplink configurations using the procedure Annex A.2.
5. ONVIF Client starts service endpoint.
6. ONVIF Client calls SetUplink with with parameters
   - RemoteAddress := *IPv4 address and port of service endpoint*
   - UserLevel := *Administrator*
   - Without CertificateID
7. ONVIF Client awaits device connecting to service endpoint.

8. ONVIF Client sends GetDeviceInformation via the service endpoint.
9. ONVIF Client awaits GetDeviceInformationResponse via the service endpoint.
10. ONVIF Client removes any existing uplink configurations using the procedure Annex A.2.
11. ONVIF Client verifies that the device disconnects from the service endpoint.

Test Result:

PASS –

- DUT passes all assertions.

FAIL –

- DUT did not send **SetUplinkResponseC** message(s).
- DUT did not connect to the service endpoint.
- DUT did not send **GetDeviceInformationResponse** message(s).
- DUT did not send **DeleteUplinkResponseC** message(s).

5.2 Uplink Streaming

5.2.1 CONNECT AND STREAM

Test Case ID: UPLINK-2-1-1

Specification Coverage: GetUplinks (Uplink Specification), SetUplink (Uplink Specification), DeleteUplink (Uplink Specification), GetProfiles (Media2 Specification), GetStreamUri (Media2 Specification), RTSP over HTTP (Streaming Specification)

Feature Under Test: Connect, Live Video, Disconnect

WSDL Reference: uplink.wsdl

Test Purpose: To verify live video over the uplink operation.

Pre-Requisite: Uplink Service is received from the DUT and uplink service endpoint is operational. Media2 Service is received from the DUT. H.264 or H.265 encoding is supported by DUT. Real-time streaming is supported by DUT. A media profile with H.264 video encoder configuration is configured on the Device.

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start the DUT.
3. ONVIF Client removes any existing uplink configurations using the procedure Annex A.2.
4. ONVIF Client calls SetUplink with parameters
   - RemoteAddress := *IPv4 address and port of service endpoint*
   - UserLevel := *Administrator*
   - Without CertificateID
5. ONVIF Client awaits device connecting to service endpoint.
6. ONVIF Client sends GetProfiles via the service endpoint without any parameters.
7. ONVIF Client awaits GetProfilesResponse via the service endpoint.
8. ONVIF Client sends GetStreamUri via the service endpoint with parameters
   - Protocol := *RtspOverHttp*
   - Profile := *First profile returned by GetProfilesResponse*
9. ONVIF Client awaits GetStreamUriResponse via the service endpoint.
10. ONVIF Client tries to start and decode media streaming via the uplink service endpoint using the procedure mentioned in Annex A.12 of Real Time Streaming using Media2 Test Specifcation with the following input and output parameters
    - in streamUri - Uri for media streaming
    - in video - media type
    - in H.264 - expected media stream encoding
11. ONVIF Client removes any existing uplink configurations using the procedure Annex A.2.
12. ONVIF Client verifies that the device disconnects from the service endpoint.

Test Result:

PASS –

- DUT passes all assertions.

FAIL –

- DUT did not send any of the required response messages.
- DUT did not connect to the service endpoint.
- DUT did not process the RTSP setup.
- DUT did not send H.264 Video.

5.3 Security

5.3.1 Client Certificate For Uplink Authentication

Test Case ID: UPLINK-3-1-1

Specification Coverage: GetUplinks (UplinkSpecification), SetUplink (Uplink Specification), DeleteUplink (UplinkSpecification)

Feature Under Test: Connect (uplink), Disconnect(uplink), GetDeviceInformation

WSDL Reference: evicemgmt.wsdl,uplink.wsdl, advancedsecurity.wsdl

Test Purpose: To verify TLS based authentication can be performed in connection setup and GetDeviceInformation command can be sent to device.

Pre-Requisite: Uplink Service, DeviceMgmt service and Advance Security service is received from the DUT.

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start the DUT.
3. ONVIF Client invokes **GetServices** message with parameters:
   - IncludeCapability := false
4. The DUT responds with a **GetServicesResponse** message with parameters:
   - Service list =: *listOfServicesWithoutCapabilities*
5. If *listOfServicesWithoutCapabilities* does not contain item with Namespace = "http://www.onvif.org/ver10/uplink/wsdl", FAIL the test and skip other steps.

6. If *listOfServicesWithoutCapabilities* does not contain item with Namespace = "http://www.onvif.org/ver10/advancedsecurity/wsdl", FAIL the test and skip other steps.
7. ONVIF Client chooses one of the available certificate with a valid private key using the procedure in Annex A.3 and get the comman name field.
8. ONVIF Client removes any existing uplink configurations using the procedure Annex A.2
9. ONVIF Client starts service endpoint.
10. ONVIF Client calls SetUplink with with parameters
    - RemoteAddress =: *IPv4 address and port of serviceendpoint*
    - UserLevel =: *Administrator*
    - CertificateID =: *With the certificate Id received in step 7*
11. ONVIF Client awaits device connecting to service endpoint. (Checks if the client certificate Common Name is same as the one selected)
12. ONVIF Client sends **GetDeviceInformation** via the service endpoint.
13. ONVIF Client awaits **GetDeviceInformationResponse** via the service endpoint.
14. ONVIF Client removes any existing uplink configurations using the procedure Annex A.2
15. ONVIF Client verifies that the device disconnects from the service endpoint.

Test Result:

PASS –

- DUT passed all assertions.

FAIL –

- DUT did not send **SetUplinkResponse** message(s).
- DUT does not have any valid certificate id with a privatekey.
- DUT did not connect to the service endpoint.
- DUT did not use the selected certificate for connection.
- DUT did not send **GetDeviceInformationResponsemessage(s)**.
- DUT did not send **DeleteUplinkResponse message(s)**.

5.4 Capabilities

5.4.1 GET SERVICES AND GET UPLINK SERVICE CAPABILITIES CONSISTENCY

Test Case ID: UPLINK-4-1-1

Specification Coverage: Capability exchange (ONVIF Core Specification), Capabilities (Uplink Service Specification)

Feature Under Test: GetServices, GetServiceCapabilities (Uplink)

WSDL Reference: devicemgmt.wsdl, uplink.wsdl

Test Purpose: To verify getting Uplink Service using GetServices request. To verify Get Services and Uplink Service Capabilities consistency.

Pre-Requisite: Uplink Service is received from the DUT.

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start the DUT.
3. ONVIF Client invokes **GetServices** message with parameters:
   - IncludeCapability := false
4. The DUT responds with a **GetServicesResponse** message with parameters:
   - Service list =: *listOfServicesWithoutCapabilities*
5. If *listOfServicesWithoutCapabilities* does not contain item with Namespace = "http://www.onvif.org/ver10/uplink/wsdl", FAIL the test and skip other steps.
6. Set *uplinkServ* := item from *listOfServicesWithoutCapabilities* list with Namespace = "http://www.onvif.org/ver10/uplink/wsdl".
7. If *uplinkServ*.Capabilities is specified, FAIL the test and skip other steps.
8. ONVIF Client invokes **GetServices** message with parameters:
   - IncludeCapability := true

9. The DUT responds with a **GetServicesResponse** message with parameters:
   - Service list =: *listOfServicesWithCapabilities*
10. If *listOfServicesWithCapabilities* does not contain item with Namespace = "http://www.onvif.org/ver10/uplink/wsdl", FAIL the test and skip other steps.
11. Set *uplinkServ* := item from *listOfServicesWithCapabilities* list with Namespace = "http://www.onvif.org/ver10/uplink/wsdl".
12. If *uplinkServ*.Capabilities is not specified, FAIL the test and skip other steps.
13. If *uplinkServ*.Capabilities does not contain valid Capabilities element for Uplink service from "http://www.onvif.org/ver10/uplink/wsdl" namespace, FAIL the test and skip other steps.
14. ONVIF Client invokes **GetServiceCapabilities** (Uplink) message.
15. The DUT responds with **GetServiceCapabilitiesResponse** message with parameters
    - Capabilities =: *cap*
16. If *cap* differs from *uplinkServ*.Capabilities.Capabilities, FAIL the test.

Test Result:

PASS –

- DUT passed all assertions.

FAIL –

- The DUT did not send **GetServicesResponse** message.
- The DUT did not send **GetServiceCapabilitiesResponse** message.

Note: The following fields are compared at step 16:

- MaxUplinks

Annex A Helper Procedures and Additional Notes

A.1 Get Uplink Service Capabilities

Name: HelperGetServiceCapabilities

Procedure Purpose: Helper procedure to get Uplink Service Capabilities from the DUT.

Pre-requisite: Uplink Service is received from the DUT.

Input: None

Returns: The service capabilities (*cap*).

Procedure:

1. ONVIF Client invokes **GetServiceCapabilities** request.
2. The DUT responds with **GetServiceCapabilitiesResponse** message with parameters
   - Capabilities =: *cap*

Procedure Result:

PASS –

- DUT passed all assertions.

FAIL –

- DUT did not send **GetServiceCapabilitiesResponse** message.

A.2 Clear Uplink Service Configuration

Name: HelperGetServiceCapabilities

Procedure Purpose: Helper procedure to remove all uplink configurations from the DUT.

Pre-requisite: Uplink Service is received from the DUT.

Input: None

Returns: None.

Procedure:

1. ONVIF Client invokes **GetUplinks** request.

2. The DUT responds with **GetUplinksResponse** message.
3. For each reported configuration with remoteAddress repeat the following steps:
    a. ONVIF Client invokes **DeleteUplinks** request.
    b. The DUT responds with **DeleteUplinksResponse** message.

Procedure Result:

PASS –

- DUT passed all assertions.

FAIL –

- DUT did not send **DeleteUplinksResponse** message.

A.3 Choose Client Certificate

Name: HelperChooseClientCertificate

Procedure Purpose: Helper procedure to choose one certificate that has private key.

Pre-requisite: Security Configuration Service is received from the DUT.

Input: None

Returns: Certificate ID which has private key and the CN field.

Procedure:

1. ONVIF Client invokes **GetAllCertificates** request.
2. The DUT responds with **GetAllCertificatesResponse** message.
3. For each reported KeyID repeat the following steps:
    a. ONVIF Client invokes **GetPrivateKeyStatus** request.
    b. The DUT responds with **GetPrivateKeyStatusResponse** message.
    c. The DUT Checks if privatekey is available, if available breaks the loop, with corresponding certificateID
4. ONVIF Client extracts the CN field of the selected certificateID, from **GetAllCertificatesResponse**.

Procedure Result:

PASS –

- DUT passed all assertions.

FAIL –

- DUT did not send **GetAllCertificates** message.
- DUT did not send **GetPrivateKeyStatus** message.
- DUT did not have a valid certificate id that has associated private key.