

ONVIF[™]

Device IO Client Test Specification

Version 17.12

December 2017

© 2017 ONVIF, Inc. All rights reserved.

Recipients of this document may copy, distribute, publish, or display this document so long as this copyright notice, license and disclaimer are retained with all copies of the document. No license is granted to modify this document.

THIS DOCUMENT IS PROVIDED "AS IS," AND THE CORPORATION AND ITS MEMBERS AND THEIR AFFILIATES, MAKE NO REPRESENTATIONS OR WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, NON-INFRINGEMENT, OR TITLE; THAT THE CONTENTS OF THIS DOCUMENT ARE SUITABLE FOR ANY PURPOSE; OR THAT THE IMPLEMENTATION OF SUCH CONTENTS WILL NOT INFRINGE ANY PATENTS, COPYRIGHTS, TRADEMARKS OR OTHER RIGHTS.

IN NO EVENT WILL THE CORPORATION OR ITS MEMBERS OR THEIR AFFILIATES BE LIABLE FOR ANY DIRECT, INDIRECT, SPECIAL, INCIDENTAL, PUNITIVE OR CONSEQUENTIAL DAMAGES, ARISING OUT OF OR RELATING TO ANY USE OR DISTRIBUTION OF THIS DOCUMENT, WHETHER OR NOT (1) THE CORPORATION, MEMBERS OR THEIR AFFILIATES HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, OR (2) SUCH DAMAGES WERE REASONABLY FORESEEABLE, AND ARISING OUT OF OR RELATING TO ANY USE OR DISTRIBUTION OF THIS DOCUMENT. THE FOREGOING DISCLAIMER AND LIMITATION ON LIABILITY DO NOT APPLY TO, INVALIDATE, OR LIMIT REPRESENTATIONS AND WARRANTIES MADE BY THE MEMBERS AND THEIR RESPECTIVE AFFILIATES TO THE CORPORATION AND OTHER MEMBERS IN CERTAIN WRITTEN POLICIES OF THE CORPORATION.

REVISION HISTORY

Vers.	Date	Description
17.12	Aug 15, 2017	Requirement level of Profile T of the following features was changed from Mandatory to Cconditional according to #220: Relay Outputs Get Digital Inputs
17.06	Jun 15, 2017	Links in Normative references section were updated.
16.07	Jun 6, 2017	First issue of Device IO Client Test Specification. The following Device IO test cases moved from ONVIF Profile T Client Test Specification accoring to #194: Relay Outputs Using Device IO Get Digital Inputs

Table of Contents

- 1 Introduction 5**
 - 1.1 Scope 5
 - 1.2 Relay Outputs Using Device IO 5
 - 1.3 Get Digital Inputs 6
- 2 Normative references 7**
- 3 Terms and Definitions 8**
 - 3.1 Conventions 8
 - 3.2 Definitions 8
 - 3.3 Abbreviations 8
 - 3.4 Namespaces 9
- 4 Test Overview 10**
 - 4.1 General 10
 - 4.1.1 Feature Level Normative Reference 10
 - 4.1.2 Expected Scenarios Under Test 10
 - 4.1.3 Test Cases 11
 - 4.2 Test Setup 11
 - 4.3 Prerequisites 11
- 5 Relay Outputs Using Device IO Test Cases 13**
 - 5.1 Feature Level Normative Reference: 13
 - 5.2 Expected Scenarios Under Test: 13
 - 5.3 GET RELAY OUTPUTS USING DEVICE IO 13
 - 5.4 SET RELAY OUTPUT STATE USING DEVICE IO 14
- 6 Get Digital Inputs Test Cases 16**
 - 6.1 Feature Level Normative Reference: 16
 - 6.2 Expected Scenarios Under Test: 16
 - 6.3 GET DIGITAL INPUTS 16



1 Introduction

The goal of the ONVIF Test Specification set is to make it possible to realize fully interoperable IP physical security implementations from different vendors. This specification also acts as an input document to the development of a test tool which will be used to test the ONVIF Client implementation conformance towards ONVIF standard. This Client Test Tool analyzes network communications between ONVIF Devices and Clients being tested and determines whether a specific Client is ONVIF conformant (see ONVIF Conformance Process Specification).

This particular document defines test cases required for testing Device IO Service features of a Client application e.g. Relay Outputs, Digital Inputs. It also describes the test framework, test setup, prerequisites, test policies needed for the execution of the described test cases.

1.1 Scope

This ONVIF Device IO Client Test Specification defines and regulates the conformance testing procedure for the ONVIF conformant Clients in the scope of Device IO Service features. Conformance testing is meant to be black-box network traces analysis and verification. The objective of this specification is to provide the test cases to test individual requirements of ONVIF Clients in the scope of Device IO Service features according to ONVIF Profile Specifications.

The principal intended purposes are:

- Provide self-assessment tool for implementations.
- Provide comprehensive test suite coverage for Device IO Service features.

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

- 3rd parties Client use cases
- Non-functional (performance and regression) testing and analysis.
- SOAP Implementation Interoperability test i.e. Web Services Interoperability Basic Profile version 2.0 (WS-I BP2.0).
- Network protocol implementation Conformance test for HTTPS and HTTP protocols.

The following sections cover test cases needed for the verification of relevant features as mentioned in the ONVIF Profile Specifications.

1.2 Relay Outputs Using Device IO

Relay Outputs Using Device IO section specifies Client ability to control of Relay Outputs connected to a device.

1.3 Get Digital Inputs

Get Digital Inputs section specifies Client ability to retrieve of Digital Inputs connected to a device.

2 Normative references

- ONVIF Conformance Process Specification:
<https://www.onvif.org/profiles/conformance/>
- ONVIF Profile Policy:
<https://www.onvif.org/profiles/>
- ONVIF Core Specifications:
<https://www.onvif.org/profiles/specifications/>
- ONVIF Core Client Test Specification:
<https://www.onvif.org/profiles/conformance/client-test/>
- ONVIF Device IO Service Specification:
<https://www.onvif.org/profiles/specifications/>
- ISO/IEC Directives, Part 2, Annex H:
<http://www.iso.org/directives>
- ISO 16484-5:2014-09 Annex P:
<https://www.iso.org/obp/ui/#!iso:std:63753:en>
- WS-BaseNotification:
http://docs.oasis-open.org/wsn/wsn-ws_base_notification-1.3-spec-os.pdf
- W3C SOAP 1.2, Part 1, Messaging Framework:
<http://www.w3.org/TR/soap12-part1/>
- W3C XML Schema Part 1: Structures Second Edition:
<http://www.w3.org/TR/xmlschema-1/>
- W3C XML Schema Part 2: Datatypes Second Edition:
<http://www.w3.org/TR/xmlschema-2/> [http://www.w3.org/TR/xmlschema-2/]

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 describes terms and definitions used in this document.

Profile	See ONVIF Profile Policy.
ONVIF Device	Computer appliance or software program that exposes one or multiple ONVIF Web Services.
ONVIF Client	Computer appliance or software program that uses ONVIF Web Services.
Conversation	A Conversation is all exchanges between two MAC addresses that contains SOAP request and response.
Network	A network is an interconnected group of devices communicating using the Internet protocol.
Network Trace Capture file	Data file created by a network protocol analyzer software (such as Wireshark). Contains network packets data recorded during a live network communications.
SOAP	SOAP is a lightweight protocol intended for exchanging structured information in a decentralized, distributed environment. It uses XML technologies to define an extensible messaging framework providing a message construct that can be exchanged over a variety of underlying protocols.
Client Test Tool	ONVIF Client Test Tool that tests ONVIF Client implementation towards the ONVIF Test Specification set.
Configuration Entity	A network video device media abstract component that produces or consumes a media stream on the network, i.e. video and/or audio stream.
Reference Token	Token provided by the device to uniquely reference an instance of a physicalIO.
Input/Output (I/O)	Currently relay ports and Video/Audio Inputs/Outputs are handled.

3.3 Abbreviations

This section describes abbreviations used in this document.

XML eXtensible Markup Language.

- HTTP** Hyper Text Transport Protocol.
- HTTPS** Hyper Text Transport Protocol over Secure Socket Layer.
- URI** Uniform Resource Identifier.
- WSDL** Web Services Description Language.
- XML** eXtensible Markup Language.

3.4 Namespaces

Prefix and namespaces used in this test specification are listed in Table 1. These prefixes are not part of the standard and an implementation can use any prefix.

Table 3.1. Defined namespaces in this specification

Prefix	Namespace URI	Description
soapenv	http://www.w3.org/2003/05/soap-envelope	Envelope namespace as defined by SOAP 1.2 [SOAP 1.2, Part 1]
xs	http://www.w3.org/2001/XMLSchema	Instance namespace as defined by XS [XML- Schema, Part1] and [XMLSchema,Part 2]
tns1	http://www.onvif.org/ver10/topics	The namespace for the ONVIF topic namespace
tt	http://www.onvif.org/ver10/schema	ONVIF XML schema descriptions
tmd	http://www.onvif.org/ver10/deviceIO/wSDL	The namespace for the WSDL DeviceIO service

4 Test Overview

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

An ONVIF client compliant to Device IO features supports for control of Digital Inputs, and Relay Outputs.

An ONVIF Profile is described by a fixed set of functionalities through a number of services that are provided by the ONVIF standard. A number of services and functionalities are mandatory for each type of ONVIF Profile. An ONVIF Device and ONVIF Client may support any combination of Profiles and other optional services and functionalities.

4.1 General

Test Cases are grouped depending on features. Each Test Cases group provides description of feature requirement level for Profiles, expected scenario under test and related test cases:

- Feature Level Normative Reference
- Expected Scenarios Under Test
- List of Test Cases

4.1.1 Feature Level Normative Reference

Feature Level Normative Reference item contains a feature ID and feature requirement level for the Profiles, which will be used for Profiles conformance.

If Feature Level Normative Reference is defined as Mandatory for some Profile, Client shall pass Expected Scenario Under Test for each Device with this Profile support to claim this Profile Conformance.

If Feature Level Normative Reference is defined as Conditional, Optional for some Profile, Client shall pass Expected Scenario Under Test for at least one Device with this Profile support to claim feature as supported.

4.1.2 Expected Scenarios Under Test

Expected Scenarios Under Test item contains expected scenario under test, conditions when the feature will be defined as supported and as not supported.

4.1.3 Test Cases

Test Case items contain list of test cases which are related to feature. Test cases provide exact procedure of testing feature support conditions.

Each Test Case contains the following parts:

- Test Label - Unique label for each test
- Test Case ID - Unique ID for each test
- Profile Normative References - Normative Reference level for the feature under test is defined in Profile Specification. This reference is informative and will not be used in conformance procedure.
- Feature Under Test - Feature which is under current test. Typically a particular command or an event.
- Test Purpose - The purpose of current test case.
- Pre-Requisite - The pre-requisite defines when the test should be performed. In case if pre-requisite does not match, the test result will be NOT DETECTED.
- Test Procedure - scenario expected to be reflected in network trace file.
- Test Result - Passed and failed criteria of the test case. Depending on these criteria test result will be defined as PASSED or FAILED.
- Validated Feature List - list of features ID related to this test case.

4.2 Test Setup

Collect Network Traces files required by the test cases.

Collect Feature List XML files for Devices detected in the Network Trace files.

Client shall support all mandatory and conditional features listed in the Device Feature List XML file supplied for the Profiles supported by the Client.

For ONVIF compatibility, the ONVIF Client shall follow the requirements of the conformance process. For details please see the latest ONVIF Conformance Process Specification.

4.3 Prerequisites

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

The Device shall be configured with an IPv4 address.

The Device shall be able to be discovered by the Client.

5 Relay Outputs Using Device IO Test Cases

5.1 Feature Level Normative Reference:

Validated Feature: RelayOutputs

Profile T Requirement: Conditional

5.2 Expected Scenarios Under Test:

1. Client connects to Device to control of Relay Outputs connected to a device.
2. Client is considered as supporting Relay Outputs if the following conditions are met:
 - Client is able to retrieve available Relay Outputs using **GetRelayOutputs** operation (Device IO Service) AND
 - Client is able to control Relay Output state using **SetRelayOutputState** operation (Device IO Service).
3. Client is considered as NOT supporting Relay Outputs if ANY of the following is TRUE:
 - No valid response to **GetRelayOutputs** request (Device IO Service) OR
 - No valid response to **SetRelayOutputState** request (Device IO Service).

5.3 GET RELAY OUTPUTS USING DEVICE IO

Test Label: Relay Output - Get Relay Outputs

Test Case ID: RELAYOUTPUTSUSINGDEVICEIO-1

Profile T Normative Reference: Conditional

Feature Under Test: Get Relay Outputs

Test Purpose: To verify that relay outputs provided by Device is received by Client using the **GetRelayOutputs** operation.

Pre-Requisite:

- The Network Trace Capture files contains at least one Conversation between Client and Device with **GetRelayOutputs** operation for Device IO Service present.
- Device supports RelayOutputs for Device IO Service (RelayOutputs).

- Client supports Capabilities feature.
- The Client Test Tool retrieves Device IO Service address from device's response on GetServices or GetCapabilities Client request.

Test Procedure (expected to be reflected in network trace file):

1. Client invokes **GetRelayOutputs** request message to Device IO Service to retrieve relay outputs from the Device.
2. Device responds with code HTTP 200 OK and **GetRelayOutputsResponse** message.

Test Result:**PASS -**

- Client **GetRelayOutputs** request messages to Device IO Service are valid according to XML Schemas listed in [Namespaces](#) AND
- Client **GetRelayOutputs** request to Device IO Service in Test Procedure fulfills the following requirements:
 - [S1] **soapenv:Body** element has child element **tds:GetRelayOutputs** AND
- Device response on the **GetRelayOutputs** request fulfills the following requirements:
 - [S2] It has HTTP 200 response code AND
 - [S3] **soapenv:Body** element has child element **tds:GetRelayOutputsResponse**.

FAIL -

- The Client failed PASS criteria.

Validated Feature List: relay_outputs.get_relay_outputs

5.4 SET RELAY OUTPUT STATE USING DEVICE IO

Test Label: Relay Output - Set Relay Output State

Test Case ID: RELAYOUTPUTSUSINGDEVICEIO-2

Profile T Normative Reference: Conditional

Feature Under Test: Set Relay Output State

Test Purpose: To verify that Client is able to trigger a relay output using the **SetRelayOutputState** operation.

Pre-Requisite:

- Device supports RelayOutputs for Device IO Service (RelayOutputs).
- The Network Trace Capture files contains at least one Conversation between Client and Device with **SetRelayOutputState** operation for Device IO Service present.
- Client supports Capabilities feature.
- The Client Test Tool retrieves Device IO Service address from device's response on GetServices or GetCapabilities Client request.

Test Procedure (expected to be reflected in network trace file):

1. Client invokes **SetRelayOutputState** request message to Device IO Service to trigger a relay output on the Device.
2. Device responds with code HTTP 200 OK and **SetRelayOutputStateResponse** message.

Test Result:**PASS -**

- Client **SetRelayOutputState** request messages to Device IO Service are valid according to XML Schemas listed in [Namespaces](#) AND
- Client **SetRelayOutputState** request to Device IO Service in Test Procedure fulfills the following requirements:
 - [S1] **soapenv:Body** element has child element **tds:SetRelayOutputState** AND
- Device response on the **SetRelayOutputState** request fulfills the following requirements:
 - [S2] It has HTTP 200 response code AND
 - [S3] **soapenv:Body** element has child element **tds:SetRelayOutputStateResponse**.

FAIL -

- The Client failed PASS criteria.

Validated Feature List: relay_outputs.set_relay_output_state

6 Get Digital Inputs Test Cases

6.1 Feature Level Normative Reference:

Validated Feature: DigitalInputs

Profile T Requirement: Conditional

6.2 Expected Scenarios Under Test:

1. Client connects to Device to retrieve of Digital Inputs connected to a device using **GetDigitalInputs** operation.
2. Client subscribes to device messages using **CreatePullPointSubscription** operation to get **tns1:Device/Trigger/DigitalInput** notifications.
3. Client uses Pull Point event mechanism to retrieve notification events from Device.
4. Client is considered as supporting Digital Inputs if the following conditions are met:
 - Client is able to retrieve available Digital Inputs using **GetDigitalInputs** operation AND
 - Client supports EventHandling_Pullpoint feature AND
 - Client is able to monitor the state of the input pins using **tns1:Device/Trigger/DigitalInput** event if device supports DigitalInputs feature.
5. Client is considered as NOT supporting Digital Inputs if ANY of the following is TRUE:
 - No valid response to **GetDigitalInputs** request OR
 - Client does not support EventHandling_Pullpoint feature OR
 - Client does not support **tns1:Device/Trigger/DigitalInput** event.

6.3 GET DIGITAL INPUTS

Test Label: Get Digital Inputs

Test Case ID: GETDIGITALINPUTS-1

Profile T Normative Reference: Conditional

Feature Under Test: Get Digital Inputs

Test Purpose: To verify that digital inputs provided by Device is received by Client using the **GetDigitalInputs** operation.

Pre-Requisite:

- The Network Trace Capture files contains at least one Conversation between Client and Device with **GetDigitalInputs** operation present.
- Device supports Digital Inputs feature (DigitalInputs).

Test Procedure (expected to be reflected in network trace file):

1. Client invokes **GetDigitalInputs** request message to retrieve digital inputs from the Device.
2. Device responds with code HTTP 200 OK and **GetDigitalInputsResponse** message.

Test Result:

PASS -

- Client **GetDigitalInputs** request messages are valid according to XML Schemas listed in [Namespaces](#) AND
- Client **GetDigitalInputs** request in Test Procedure fulfills the following requirements:
 - [S1] **soapenv:Body** element has child element **tmd:GetDigitalInputs** AND
- Device response on the **GetDigitalInputs** request fulfills the following requirements:
 - [S2] It has HTTP 200 response code AND
 - [S3] **soapenv:Body** element has child element **tmd:GetDigitalInputsResponse**.

FAIL -

- The Client failed PASS criteria.

Validated Feature List: digital_inputs.get_digital_inputs