

ONVIF[®]
**Replay Control Device Test
Specification**

Version 19.12

December 2019

© 2008 - 2019 by ONVIF: Open Network Video Interface Forum, 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
12.12	20th/Dec, 2012	First issue of Replay Control Test Specification
13.06	Jun, 2013	Formatting and spelling updates.
13.12	Dec, 2014	The following test cases updated with ID change: PLAYBACK VIDEO STREAMING – PAUSE WITHOUT RANGE PLAYBACK VIDEO STREAMING – PAUSE WITH RANGE PLAYBACK AUDIO STREAMING – PAUSE WITHOUT RANGE PLAYBACK AUDIO STREAMING – PAUSE WITH RANGE PLAYBACK METADATA STREAMING – PAUSE WITHOUT RANGE PLAYBACK METADATA STREAMING – PAUSE WITH RANGE PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – PAUSE WITHOUT RANGE PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – PAUSE WITH RANGE
14.06	Jun, 2014	The following test cases updated with ID change: PLAYBACK VIDEO STREAMING – SEEK PLAYBACK AUDIO STREAMING – SEEK PLAYBACK METADATA STREAMING – SEEK PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – SEEK Annex A.2 Recording Environment Pre-Requisite was updated
17.06	Mar 27, 2017	REPLAY-4-1-1 REPLAY CONFIGURATION was added according to #1342. GETREPLAYURI COMMAND WITH INVALID RECORDING TOKEN test case was moved into 'Get URI' section to synchronize structure with the Device Test Tool.
17.12	Oct 06, 2017	The following Annex was added according to #1285: RTSP Authentication Check
18.06	Jun, 2018	Reformatting document using new template
18.12	Oct 2, 2018	The following were updated in the scope of #1686: Test sequences were removed REPLAY-3-1-12 PLAYBACK VIDEO STREAMING – RATECONTROL (updated to use RTP-Unicast/RTSP/HTTP/TCP transport) REPLAY-3-2-11 PLAYBACK AUDIO STREAMING – RATECONTROL (updated to use RTP-Unicast/RTSP/HTTP/TCP transport) REPLAY-3-3-11 PLAYBACK METADATA STREAMING – RATECONTROL (updated to use RTP-Unicast/RTSP/HTTP/TCP transport) REPLAY-3-4-11 PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – RATECONTROL (updated to use RTP-Unicast/RTSP/HTTP/TCP transport)

Vers.	Date	Description
18.12	Oct 5, 2018	The following were updated in the scope of #1622: REPLAY-3-1-12 PLAYBACK VIDEO STREAMING – RATECONTROL (step 21 with comparison of streaming time removed) REPLAY-3-2-11 PLAYBACK AUDIO STREAMING – RATECONTROL (step 21 with comparison of streaming time removed) REPLAY-3-3-11 PLAYBACK METADATA STREAMING – RATECONTROL (step 21 with comparison of streaming time removed) REPLAY-3-4-11 PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – RATECONTROL (step 21 with comparison of streaming time removed)
18.12	Dec 21, 2018	Switching Hub description in 'Network Configuration for DUT' section was updated according to #1737
19.12	Aug 30, 2019	The following were changed according to #1783: Annex A.5 Check of IP address type in response to RTSP DESCRIBE (added) All test cases from the following sections are updated to use Annex A.5: Playback Control – Video Streaming Playback Control – Audio Streaming Playback Control – Metadata Streaming Playback Control – Video, Audio and Metadata Streaming

TABLE OF CONTENTS

1	Introduction	6
1.1	Scope.....	6
2	Terms and Definitions	9
2.1	Conventions.....	9
2.2	Definitions	9
2.3	Abbreviations	9
3	Test Overview	11
3.1	Test Setup.....	11
3.2	Prerequisites	12
3.3	Test Policy.....	12
4	Replay Control Test Cases	15
4.1	Capabilities	15
4.2	Get URI.....	17
4.3	Playback Control – Video Streaming.....	17
4.4	Playback Control – Audio Streaming.....	47
4.5	Playback Control – Metadata Streaming.....	68
4.6	Playback Control – Video, Audio and Metadata Streaming	96
4.7	Replay Configuration	120
Annex A	122
A.1	Invalid RTP header extension.....	122
A.2	Recording Environment Pre-Requisite	122
A.3	Get Service Capabilities	122
A.4	RTSP Authentication Check	123
A.5	Check of IP address type in response to RTSP DESCRIBE	123

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 Network Interface Specs] and [ONVIF Conformance] requirements. Also the test cases are to be basic inputs for some Profile specification requirements. It also describes the test framework, test setup, pre-requisites, test policies needed for the execution of the described test cases.

This ONVIF Replay Control Test Specification acts as a supplementary document to the [ONVIF Network Interface Specs], illustrating test cases need to be executed and passed. 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 Replay Control 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 the test cases to test individual requirements of ONVIF devices according to ONVIF core services which are defined in [ONVIF Network Interface Specs].

The principal intended purposes are:

1. To provide self-assessment tool for implementations.
2. To provide comprehensive test suite coverage for [ONVIF Network Interface Specs].

This specification does not address the following.

1. Product use cases and non-functional (performance and regression) testing.
2. SOAP Implementation Interoperability test i.e. Web Services Interoperability Basic Profile version 2.0 (WS-I BP2.0).
3. Network protocol implementation Conformance test for HTTPS, HTTP, RTP and RTSP protocols.
4. Wi-Fi Conformance test

The set of ONVIF Test Specification will not cover the complete set of requirements as defined in [ONVIF Network Interface Specs]; instead it would cover its subset.

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

1.1.1 Capabilities

Capabilities test cases are covered for verification to get Replay Control Service capabilities. It means that GetServices and GetServiceCapabilities commands are covered by this test case.

1.1.2 Playback Control – Video Streaming

Playback Control – Video Streaming covers the test cases needed for the verification of video replay streaming features as mentioned in [ONVIF Network Interface Specs]. Playback Control – Video Streaming section defines different video streaming options based on RTP for video playback streams. Stream control is done using RTSP protocol.

The scope of this specification is to cover the following playback streaming options for video streams:

- RTSP control requests including ONVIF RTSP Headers extensions
- RTP Unicast over UDP
- RTP over RTSP over TCP
- RTP over RTSP over HTTP over TCP
- ONVIF RTP Header Extensions

1.1.3 Playback Control – Audio Streaming

Playback Control – Audio Streaming covers the test cases needed for the verification of audio replay streaming features as mentioned in [ONVIF Network Interface Specs]. Playback Control – Audio Streaming section defines different audio streaming options based on RTP for audio playback streams. Stream control is done using RTSP protocol.

The scope of this specification is to cover the following playback streaming options for audio streams:

- RTSP control requests including ONVIF RTSP Headers extensions
- RTP Unicast over UDP
- RTP over RTSP over TCP
- RTP over RTSP over HTTP over TCP
- ONVIF RTP Header Extensions

1.1.4 Playback Control – Metadata Streaming

Playback Control – Metadata Streaming covers the test cases needed for the verification of metadata replay streaming features as mentioned in [ONVIF Network Interface Specs]. Playback

Control – Metadata Streaming section defines different audio streaming options based on RTP for metadata playback streams. Stream control is done using RTSP protocol.

The scope of this specification is to cover the following playback streaming options for metadata streams:

- RTSP control requests including ONVIF RTSP Headers extensions
- RTP Unicast over UDP
- RTP over RTSP over TCP
- RTP over RTSP over HTTP over TCP
- ONVIF RTP Header Extensions

1.1.5 Playback Control – Video, Audio and Metadata Streaming

Playback Control – Video, Audio and Metadata Streaming covers the test cases needed for the verification of replay streaming features for video, Audio and metadata in combination as mentioned in [ONVIF Network Interface Specs]. Playback Control – Video, Audio and Metadata Streaming section defines different video, Audio and metadata streaming options based on RTP for playback streams. Stream control is done using RTSP protocol.

The scope of this specification to cover the following playback streaming options for media streams.

- RTSP control requests including ONVIF RTSP Headers extensions
- RTP Unicast over UDP
- RTP over RTSP over TCP
- RTP over RTSP over HTTP over TCP
- ONVIF RTP Header Extensions

2 Terms and Definitions

2.1 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].

2.2 Definitions

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

Term Definition	Definition
Track	An individual data channel consisting of video, audio, or metadata. This definition is consistent with the definition of track in [RFC 2326]
Metadata	All streaming data except video and audio, including video analytics results, PTZ position data and other metadata (such as textual data from POS applications).
Recording	A container for a set of audio, video and metadata tracks. A recording can hold one or more tracks. A track is viewed as an infinite timeline that holds data at certain times.

2.3 Abbreviations

This section describes abbreviations used in this document.

Abbreviation	Normal
DUT	Device Under Test
HTTP	Hyper Text Transport Protocol
RTCP	RTP Control Protocol
RTSP	Real Time Streaming Protocol
RTP	Real-time Transport Protocol
SDP	Session Description Protocol
TCP	Transport Control Protocol
UTC	Coordinated Universal Time
UDP	User Datagram Protocol
URI	Uniform Resource Identifier
WSDL	Web Services Description Language
WS-I BP 2.0	Web Services Interoperability Basic Profile version 2.0

Abbreviation	Normal
XML	eXtensible Markup Language

3 Test Overview

This section describes the test setup and prerequisites needed, and the test policies that should be followed for test case execution.

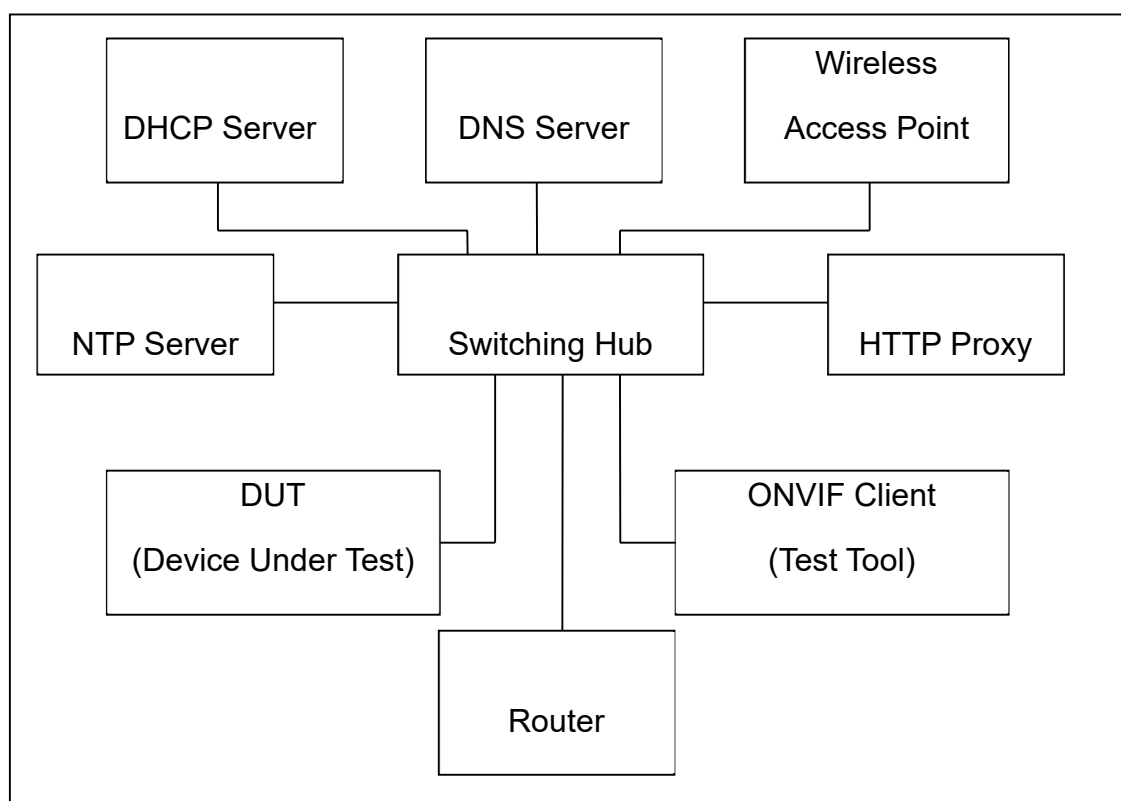
3.1 Test Setup

3.1.1 Network Configuration for device under test

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

Based on the individual test case requirements, some of the entities in the below setup may not be needed for the execution of those corresponding test cases.

Figure 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.

HTTP Proxy: provides facilitation in case of RTP and RTSP tunneling over HTTP.

Wireless Access Point: provides wireless connectivity to the devices that support wireless connection.

DNS Server: provides DNS related information to the connected devices.

DHCP Server: provides IPv4 Address to the connected devices.

NTP Server: provides time synchronization between ONVIF Client and DUT.

Switching Hub: provides network connectivity among all the test equipment in the test environment. All devices should be connected to the Switching Hub. When running multiple test instances in parallel on the same network, the Switching Hub should be configured to use filtering in order to avoid multicast traffic being flooded to all ports, because this may affect test stability.

Router: provides router advertisements for IPv6 configuration.

3.2 Prerequisites

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

The DUT shall be configured with an IPv4 address.

The DUT shall be IP reachable [in the test configuration].

The DUT shall be available for being discovered by the ONVIF Client Test Tool.

The DUT shall be configured with the time i.e. manual configuration of UTC time and if NTP is supported by DUT then NTP time shall be synchronized with NTP Server.

The DUT time and client Test tool time shall be synchronized with each other either manually or by common NTP server.

3.3 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.

3.3.1 Capabilities

The device under test shall demonstrate replay control service capability both in GetServices and GetServiceCapabilities responses. A DUT that does not display replay control service capability constitutes failure of test procedure.

Please refer to Section 4.1 for Capabilities Test Cases.

3.3.2 Playback Control – Video Streaming

Replay streaming test case execution would need the recording with video data. So device shall have at least one recording with video data. ONVIF Client shall explicitly specify the recording token for them.

DUT shall give the Replay Control Service entry point by Services command.

Please refer to Section 4.2 for Playback Control Video Streaming Test Cases.

Please refer to Annex A.2 for Recording Environment Pre-Requisite.

3.3.3 Playback Control – Audio Streaming

Replay streaming test case execution would need the recording with audio data. So device shall have at least one recording with audio data, if Audio Recording is supported. ONVIF Client shall explicitly specify the recording token for it.

DUT shall give the Replay Control Service entry point by Services command.

Please refer to Section 4.3.12 for Playback Control Audio Streaming Test Cases.

Please refer to Annex A.2 for Recording Environment Pre-Requisite.

3.3.4 Playback Control – Metadata Streaming

Replay streaming test case execution would need the recording with metadata data. So device shall have at least one recording with metadata data, if Metadata Recording is supported. ONVIF Client shall explicitly specify the recording token for it.

DUT shall give the Replay Control Service entry point by Services command.

Please refer to Section 4.5 for Playback Control Metadata Streaming Test Cases.

Please refer to Annex A.2 for Recording Environment Pre-Requisite.

3.3.5 Playback Control – Video, Audio and Metadata Streaming

Replay streaming test case execution would need the recording with both video data, audio data (if Audio Recording is supported), and metadata data (if Metadata Recording is supported). So device shall have at least one recording with both video data, audio data (if Audio Recording is supported), and metadata data (if Metadata Recording is supported). ONVIF Client shall explicitly

specify the recording token for it.

DUT shall give the Replay Control Service entry point by Services command.

Please refer to Section 4.5.11 for Playback Control Video, Audio and Metadata Streaming Test Cases.

Please refer to Annex A.2 for Recording Environment Pre-Requisite.

3.3.6 Authentication method selection as a testing framework

According to later version of [ONVIF Core Specs ONVIF client is required to support both HTTP digest and WS-UsernameToken functionality as authentication functionality. Therefore, ONVIF Client (ONVIF Device Test Tool in this context) as a testing framework shall properly select authentication method between the two based on the response from DUT toward specific request. The following is the deterministic procedure on which authentication method is to be selected.

Procedure:

1. ONVIF Client invokes a specific command which is under testing without any user credentials (no WS-UsernameToken, no HTTP digest authentication header).
2. If DUT returns a correct response, then ONVIF Client determines that DUT does not require any user authentication toward the command according to the configured security policy.
3. If DUT returns HTTP 401 Unauthorized error along with WWW-Authentication: Digest header, then ONVIF Client determines that DUT supports HTTP digest authentication. ONVIF Client shall provide the proper level of user credential to continue the test procedure.
4. If DUT returns SOAP fault (Sender/NotAuthorized) message, then ONVIF Client determines that WS-UsernameToken is supported by DUT. ONVIF Client shall provide the proper level of user credential to continue the test procedure.

4 Replay Control Test Cases

4.1 Capabilities

4.1.1 REPLAY SERVICE CAPABILITIES

Test Label: Replay Service Capabilities Verification.

Test Case ID: REPLAY-1-1-1

ONVIF Core Specification Coverage: Capability exchange

Command under test: GetServiceCapabilities (for Replay Service)

WSDL Reference: replay.wsdl

Test Purpose: To verify Replay Service Capabilities of the DUT.

Pre-Requisite: Replay Service was received from the DUT.

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start the DUT.
3. ONVIF Client will invoke GetServiceCapabilitiesRequest message to retrieve replay service capabilities of the DUT.
4. Verify the GetServiceCapabilitiesResponse from the DUT.

Test Result:

PASS –

The DUT passed all assertions.

FAIL –

The DUT did not send valid GetServiceCapabilitiesResponse.

4.1.2 GET SERVICES AND GET REPLAY SERVICE CAPABILITIES CONSISTENCY

Test Label: Get Services and Replay Service Capabilities Consistency Verification.

Test Case ID: REPLAY-1-1-2

ONVIF Core Specification Coverage: Capability exchange

Command under test: GetServices, GetServiceCapabilities (for Replay Service)

WSDL Reference: devicemgmt.wsdl, replay.wsdl

Test Purpose: To verify Get Services and Replay Service Capabilities consistency.

Pre-Requisite: None.

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start the DUT.
3. ONVIF Client will invoke GetServicesRequest message (**IncludeCapability = true**) to retrieve all services of the DUT with service capabilities.
4. Verify the GetServicesResponse message from the DUT.
5. ONVIF Client will invoke GetServiceCapabilitiesRequest message to retrieve Replay service capabilities of the DUT.
6. Verify the GetServiceCapabilitiesResponse message from the DUT.

Test Result:

PASS –

The DUT passed all assertions.

FAIL –

The DUT did not send a valid GetServicesResponse message.

The DUT did not send a valid GetServiceCapabilitiesResponse message.

The DUT sent different Capabilities in GetServicesResponse message and in GetServiceCapabilitiesResponse message.

NOTE: Service will be defined as Replay service if it has Namespace element that is equal to "http://www.onvif.org/ver10/replay/wsdl".

4.2 Get URI

4.2.1 GETREPLAYURI COMMAND WITH INVALID RECORDING TOKEN

Test Label: Replay Service GetReplayUri Command Validation (Invalid Token).

Test Case ID: REPLAY-1-2-1

ONVIF Core Specification Coverage: GetReplayUri

Command under test: GetReplayUri

WSDL Reference: replay.wsdl

Test Purpose: To check GetReplayUri command with invalid RecordingToken.

Pre-Requisite: None.

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. ONVIF Client will invoke GetReplayUriRequest message with invalid RecordingToken (RecordingToken = [**Nonexistent Token**]).
4. Verify the DUT generates a SOAP 1.2 fault message (**InvalidArgVal /InvalidToken**).

Test Result:

PASS –

The DUT passed all assertions.

FAIL –

The DUT did not send the SOAP 1.2 fault message (InvalidArgVal/InvalidToken).

The DUT did not send correct SOAP 1.2 fault message.

NOTE: Other faults than specified in the test are acceptable, though the specified are preferable.

4.3 Playback Control – Video Streaming

4.3.1 PLAYBACK VIDEO STREAMING – MEDIA CONTROL

Test Label: DUT RTSP Control Messages Verification for Replay Video.

Test Case ID: REPLAY-3-1-1

ONVIF Core Specification Coverage: RTP data transfer via UDP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify RTSP control messages of the DUT for replay video streaming.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP OPTIONS control request to get the RTSP methods supported by DUT.
9. The DUT sends 200 OK Response and list of supported RTSP methods.
10. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
11. The DUT sends 200 OK Response and SDP message (*sdp*).
12. Set *responseHeader* := header of response to DESCRIBE.
13. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
14. ONVIF Client validates the session description information in the SDP message.
15. ONVIF Client verifies tag with TrackReference in SDP message for each video track.
16. ONVIF Client will invoke RTSP SETUP control request with "onvif-replay" feature tag (**"Require: onvif-replay"**) to create a RTSP Session.
17. The DUT sends 200 OK Response and Stream Information details.

18. ONVIF Client verifies “Transport”, “Session” and “timeout” header fields in the SETUP response message.
19. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-**”) control request to initiate the video streaming.
20. The DUT sends 200 OK Response and RTP protocol information.
21. ONVIF Client verifies “Session”, “RTP-Info”, “seq”, “uri” and “rtptime” header fields in the PLAY response message.
22. The DUT transfers video streams over UDP.
23. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
24. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
25. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send a valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send mandatory RTSP commands DESCRIBE SETUP, PLAY, TEARDOWN, and OPTIONS in OPTIONS response.

The DUT sent SDP message with one or more video track without the TrackReference in corresponding format (a:x-onvif-track:<TrackReference>).

The DUT did not send correct video stream information in the SDP message.

The DUT did not send RTSP 200 OK response for RTSP OPTIONS, DESCRIBE, SETUP and PLAY requests.

The DUT did not send mandatory headers or fields in the SETUP response message.

The DUT did not send mandatory headers or fields in the PLAY response message.

RTSP Session is terminated by the DUT during video streaming.

The DUT did not send valid RTP header including replay header extension for video stream.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for video track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for video track will be used as T2.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.2 PLAYBACK VIDEO STREAMING – RTP-Unicast/UDP

Test Label: Replay Viewing DUT video streaming using RTP-Unicast/UDP transport.

Test Case ID: REPLAY-3-1-2

ONVIF Core Specification Coverage: RTP data transfer via UDP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify playback video streaming based on RTP/UDP Unicast Transport.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token Token1 of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type

12. ONVIF Client will invoke RTSP SETUP (“**Require: onvif-replay**”) request with transport parameter as RTP/UDP.
13. The DUT sends 200 OK message and the video stream information.
14. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-**”) control request to initiate the video streaming.
15. The DUT sends 200 OK message and starts video streaming.
16. The DUT sends RTP video stream to ONVIF Client over UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
18. ONVIF Client validates RTCP packets if The DUT sends theirs.
19. ONVIF Client will invoke RTSP TEARDOWN control request at the end of video streaming to terminate the RTSP session.
20. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send a valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during video streaming.

The DUT did not send valid RTP header and RTP header extension for video stream.

The DUT sent an incorrect RTCP report.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for video track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for video track will be used as T2.

NOTE: RTCP report will be validated only if it is received from the DUT.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.3 PLAYBACK VIDEO STREAMING – RTP-Unicast/RTSP/HTTP/TCP

Test Label: Replay Viewing DUT video streaming using HTTP transport.

Test Case ID: REPLAY-3-1-3

ONVIF Core Specification Coverage: RTP/RTSP/HTTP/TCP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify playback video streaming based on HTTP Transport.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "HTTP").
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP ("**Require: onvif-replay**") request with transport parameter as **RTP/TCP**.
13. The DUT sends 200 OK message and the video stream information.
14. ONVIF Client will invoke RTSP PLAY ("**Require: onvif-replay**", "**Range: clock = T1-**") control request to initiate the video streaming.

15. The DUT sends 200 OK message and starts video streaming.
16. The DUT sends RTP video stream to ONVIF Client over HTTP.
17. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
18. ONVIF Client validates RTCP packets if The DUT sends theirs.
19. ONVIF Client will invoke RTSP TEARDOWN control request at the end of video streaming to terminate the RTSP session.
20. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send a valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during video streaming.

The DUT did not send valid RTP header and RTP header extension for video stream.

The DUT sent an incorrect RTCP report.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for video track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for video track will be used as T2.

NOTE: RTCP report will be validated only if it is received from the DUT.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.4 PLAYBACK VIDEO STREAMING – RTP/RTSP/TCP

Test Label: Replay Viewing DUT video streaming using RTP/RTSP/TCP transport.

Test Case ID: REPLAY-3-1-4

ONVIF Core Specification Coverage: RTP/RTSP/TCP, RTP, RTCP, Stream control, RTSP,

Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify playback video streaming based on RTP/RTSP/TCP using RTSP tunnel.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "RTSP").
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP ("**Require: onvif-replay**") request with transport parameter as **RTP/TCP**.
13. The DUT sends 200 OK message and the video stream information.
14. ONVIF Client will invoke RTSP PLAY ("**Require: onvif-replay**", "**Range: clock = T1-**") control request to initiate the video streaming.
15. The DUT sends 200 OK message and starts video streaming.
16. The DUT sends RTP video stream to ONVIF Client over TCP.
17. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
18. ONVIF Client validates RTCP packets if The DUT sends theirs.
19. ONVIF Client will invoke RTSP TEARDOWN control request at the end of video streaming to terminate the RTSP session.

20. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:

PASS –

The DUT passed all assertions.

FAIL –

The DUT did not send a valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during video streaming.

The DUT did not send valid RTP header and RTP header extension for video stream.

The DUT sent an incorrect RTCP report.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for video track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for video track will be used as T2.

NOTE: RTCP report will be validated only if it is received from the DUT.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.5 REVERSE PLAYBACK VIDEO STREAMING – RTP-Unicast/UDP

Test Label: Reverse Replay Viewing DUT video streaming using RTP-Unicast/UDP transport.

Test Case ID: REPLAY-3-1-5

ONVIF Core Specification Coverage: RTP data transfer via UDP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify reverse playback video streaming based on RTP/UDP Unicast Transport.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP").
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP ("**Require: onvif-replay**") request with transport parameter as **RTP/UDP**.
13. The DUT sends 200 OK message and the video stream information.
14. ONVIF Client will invoke RTSP PLAY ("**Require: onvif-replay**", "**Range: clock = T2-**", "**Scale: -1.0**") control request to initiate the video streaming.
15. The DUT sends 200 OK message and starts video streaming.
16. The DUT sends RTP video stream to ONVIF Client over UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
18. ONVIF Client verifies Packet Transmission Order (GOPs are sent in reverse order, but packets within a GOP are sent in forward order).
19. ONVIF Client validates the RTP sequence numbers. Verify that the RTP sequence numbers of packets are increment monotonically in the order of delivery, not in the intended order of playback.
20. ONVIF Client checks Timestamps (the RTP timestamps shell indicate times at which each frame should be rendered at the client).
21. ONVIF Client validates RTCP packets if The DUT sends theirs.

22. ONVIF Client will invoke RTSP TEARDOWN control request at the end of video streaming to terminate the RTSP session.
23. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send s valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send s valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during video streaming.

The DUT did not send valid RTP header and RTP header extension for video stream.

The DUT sent GOPs or packets within a GOP in wrong order.

The DUT sent RTP packets with invalid sequence number.

The DUT sent RTP packets with invalid timestamp.

The DUT sent incorrect RTCP report.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for video track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for video track will be used as T2.

NOTE: RTCP report will be validated only if it is received from the DUT.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.6 REVERSE PLAYBACK VIDEO STREAMING – RTP- Unicast/RTSP/HTTP/TCP

Test Label: Reverse Replay Viewing DUT video streaming using HTTP transport.

Test Case ID: REPLAY-3-1-6

ONVIF Core Specification Coverage: RTP/RTSP/HTTP/TCP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify reverse playback video streaming based on HTTP Transport.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "HTTP").
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP ("**Require: onvif-replay**") request with transport parameter as **RTP/TCP**.
13. The DUT sends 200 OK message and the video stream information.
14. ONVIF Client will invoke RTSP PLAY ("**Require: onvif-replay**", "**Range: clock = T2-**", "**Scale: -1.0**") control request to initiate the video streaming.
15. The DUT sends 200 OK message and starts video streaming.
16. The DUT sends RTP video stream to ONVIF Client over HTTP.
17. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.

18. ONVIF Client verifies Packet Transmission Order (GOPs are sent in reverse order, but packets within a GOP are sent in forward order).
19. ONVIF Client validates the RTP sequence numbers. Verify that the RTP sequence numbers of packets are increment monotonically in the order of delivery, not in the intended order of playback.
20. ONVIF Client checks Timestamps (the RTP timestamps shall indicate times at which each frame should be rendered at the client).
21. ONVIF Client validates RTCP packets if The DUT sends theirs.
22. ONVIF Client will invoke RTSP TEARDOWN control request at the end of video streaming to terminate the RTSP session.
23. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send a valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TEARDOWN requests.

RTSP Session was terminated by the DUT during video streaming.

The DUT did not send a valid RTP header and RTP header extension for video stream.

The DUT sent GOPs or packets within a GOP in wrong order.

The DUT sent RTP packets with invalid sequence number.

The DUT sent RTP packets with invalid timestamp.

The DUT sent an incorrect RTCP report.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for video track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for video track will be used as T2.

NOTE: RTCP report will be validated only if it is received from the DUT.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.7 REVERSE PLAYBACK VIDEO STREAMING – RTP/RTSP/TCP

Test Label: Reverse Replay Viewing DUT video streaming using RTP/RTSP/TCP transport.

Test Case ID: REPLAY-3-1-7

ONVIF Core Specification Coverage: RTP/RTSP/TCP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify reverse playback video streaming based on RTP/RTSP/TCP using RTSP tunnel.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "RTSP").
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type

12. ONVIF Client will invoke RTSP SETUP (“**Require: onvif-replay**”) request with transport parameter as **RTP/TCP**.
13. The DUT sends 200 OK message and the video stream information.
14. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T2-**”, “**Scale: -1.0**”) control request to initiate the video streaming.
15. The DUT sends 200 OK message and starts video streaming.
16. The DUT sends RTP video stream to ONVIF Client over TCP.
17. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
18. ONVIF Client verifies Packet Transmission Order (GOPs are sent in reverse order, but packets within a GOP are sent in forward order).
19. ONVIF Client validates the RTP sequence numbers. Verify that the RTP sequence numbers of packets are increment monotonically in the order of delivery, not in the intended order of playback.
20. ONVIF Client checks Timestamps (the RTP timestamps shell indicate times at which each frame should be rendered at the client).
21. ONVIF Client validates RTCP packets if The DUT sends theirs.
22. ONVIF Client will invoke RTSP TEARDOWN control request at the end of video streaming to terminate the RTSP session.
23. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during video streaming.

DUT sent GOPs or packets within a GOP in wrong order.

DUT sent RTP packets with invalid sequence number.

DUT sent RTP packets with invalid timestamp.

DUT did not send valid RTP header and RTP header extension for video stream.

DUT sent incorrect RTCP report.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for video track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for video track will be used as T2.

NOTE: RTCP report will be validated only if it is received from the DUT.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.8 PLAYBACK VIDEO STREAMING – PLAY WITH RANGE

Test Label: Replay Video Streaming from Specified Range.

Test Case ID: REPLAY-3-1-10

ONVIF Core Specification Coverage: Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify stop of replay video streaming based on RTP/UDP Unicast Transport after specified time Range was reached.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP").
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.

9. The DUT sends 200 OK Response and SDP message (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP control request with “onvif-replay” feature tag (“**Require: onvif-replay**”) to create a RTSP Session.
13. The DUT sends 200 OK Response and Stream Information details.
14. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-T1 + delta**”) control request to initiate the video streaming.
15. The DUT sends 200 OK Response and RTP protocol information.
16. The DUT transfers video streams over RTP/UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
18. The DUT stops transferring of video stream after the specified time **T1 + delta** was reached.
19. ONVIF Client verifies that there are no RTP received after specified time **[T1 + delta]**.
20. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
21. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send GetReplayUriResponse message.

The DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP OPTIONS, DESCRIBE, SETUP and PLAY requests.

The DUT did not stop video streaming after specified time.

The DUT did not send valid RTP header in video stream (RTP sequence number not in the orders of delivery, timestamps correspond to playback timing considering inaccuracy).

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for video track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for video track will be used as T2.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.9 PLAYBACK VIDEO STREAMING - I-FRAMES

Test Label: Replay I-Frames Viewing DUT Video Streaming.

Test Case ID: REPLAY-3-1-11

ONVIF Core Specification Coverage: Frames header field.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify I-frames RTSP playback.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2). Recording contains H.264 or MPEG-4 encoding.

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with video with H.264 or MPEG-4 encoding from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (RecordingToken = Token1, Stream = "RTP-Unicast", Transport. Protocol = "UDP").
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.
10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type

11. ONVIF Client will invoke RTSP SETUP request (“**Require: onvif-replay**”) with transport parameter as **RTP/UDP**.
12. The DUT sends 200 OK message and the video stream information.
13. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = [start of recording]-**”, “**Frames: intra**”) control request to initiate the video streaming with intra frame only.
14. The DUT sends 200 OK message and starts video streaming.
15. The DUT sends RTP video stream to ONVIF Client over UDP.
16. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
17. ONVIF Client verifies that The DUT sends I-Frames only.
18. ONVIF Client will invoke RTSP TEARDOWN control request at the end of video streaming to terminate the RTSP session.
19. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send a valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during video streaming.

The DUT did not accept a Frames header with a value of Intra.

The DUT did not send valid RTP header for video stream (RTP sequence number not in the order of delivery, timestamps correspond to playback timing considering inaccuracy).

The DUT did not send at least one I-frame.

The DUT sent other frames than I-frames.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for video track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for video track will be used as T2.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.10 PLAYBACK VIDEO STREAMING – RATECONTROL

Test Label: Compare Replay Video Streaming with different Rate-Control.

Test Case ID: REPLAY-3-1-12

ONVIF Core Specification Coverage: Rate-Control header field.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify applying of Rate-Control value for playback video streaming.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "HTTP").
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.
10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
11. ONVIF Client will invoke RTSP SETUP request ("**Require: onvif-replay**") with transport parameter as **RTP/AVP/TCP**.
12. The DUT sends 200 OK message and the video stream information.

13. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-T1+Delta**”, “**Rate-Control: yes**”) control request to initiate the video streaming.
14. The DUT sends 200 OK message and starts video streaming.
15. The DUT sends RTP video stream to ONVIF Client over HTTP.
16. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
17. ONVIF Client waits while replay streaming is played and calculates play duration **D1**.
18. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-T1+Delta**”, “**Rate-Control: no**”) control request to initiate the video streaming.
19. The DUT sends 200 OK message and starts video streaming.
20. The DUT sends RTP video stream to ONVIF Client over HTTP.
21. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
22. ONVIF Client waits while replay streaming is played and calculates play duration **D2**.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send a valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY requests.

RTSP Session was terminated by the DUT during video streaming.

The DUT did not accept Rate-Control header for PLAY command.

The DUT did not send valid RTP header for video stream with rate control (RTP sequence number not in the order of delivery, timestamps correspond to playback timing considering inaccuracy).

The DUT did not send valid RTP header for video stream without rate control (RTP sequence number not in the order of delivery, timestamps correspond to original sampling times of recorded frames considering inaccuracy).

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.11 PLAYBACK VIDEO STREAMING – IMMEDIATE HEADER

Test Label: Compare Replay Video streaming with Immediate header.

Test Case ID: REPLAY-3-1-13

ONVIF Core Specification Coverage: Immediate header field

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To compare RTSP playback with Immediate header.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP").
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.
10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
11. ONVIF Client will invoke RTSP SETUP request ("**Require: onvif-replay**") with transport parameter as **RTP/UDP**.
12. The DUT sends 200 OK message and the video stream information.
13. ONVIF Client will invoke RTSP PLAY ("**Require: onvif-replay**", "**Range: clockT1 -**") control request to initiate the video streaming.
14. The DUT sends 200 OK message and starts video streaming.
15. The DUT sends RTP video stream to ONVIF Client over UDP.

16. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
17. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock [T1+delta]**”, “**Immediate: yes**”) control request to initiate the second video streaming from other time.
18. The DUT cancelling existing first PLAY command and immediately start playing from the **T1+delta**.
19. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation. ONVIF Client verifies that first packet sent from the new location have the D (discontinuity) bit set in its RTP extension header.
20. ONVIF Client will invoke RTSP TEARDOWN control request at the end of video streaming to terminate the RTSP session.
21. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send a valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during video streaming.

The DUT did not accept an Immediate header for PLAY command.

The DUT did not send valid RTP header for video stream.

The DUT did not cancel the first PLAY command after PLAY command with “Immediate: yes”.

The DUT did not immediately start playing from the new location after PLAY command with “Immediate: yes”.

The DUT did not send valid RTP header from the new location (first packet did not have the D (discontinuity) bit set in its RTP extension header).

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.12 PLAYBACK VIDEO STREAMING – PAUSE WITHOUT RANGE

Test Label: Replay Viewing DUT video streaming using with Pause (no Range).

Test Case ID: REPLAY-3-1-15

ONVIF Core Specification Coverage: Playback, Pause.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify pause without Range of replay video streaming based on RTP/UDP Unicast Transport.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP").
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
9. The DUT sends 200 OK Response and SDP message (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP control request with "onvif-replay" feature tag (**"Require: onvif-replay"**) to create a RTSP Session with transport parameter as „RTP/UDP“.
13. The DUT sends 200 OK Response and Stream Information details.
14. ONVIF Client will invoke RTSP PLAY (**"Require: onvif-replay"**, **"Range: clock = T1-"**) control request to initiate the video streaming.
15. The DUT sends 200 OK Response and RTP protocol information.
16. The DUT transfers video streams over RTP/UDP.

17. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
18. ONVIF Client will invoke RTSP PAUSE (**no Range header**) control request to pause the video streaming.
19. The DUT sends 200 OK Response.
20. ONVIF Client verifies “CSeq” header field and optional “Date” header field if it is present in the PAUSE response message.
21. The DUT pauses transferring of video streams and DUT sets the pause point to the current normal play time.
22. ONVIF Client verifies that there are no RTP received after this.
23. ONVIF Client will wait 2s after pause of replay streaming and will invoke RTSP PLAY (**“Require: onvif-replay”, no Range**) control request to resume the video streaming.
24. The DUT sends 200 OK Response and RTP protocol information.
25. The DUT resumes video streams over RTP/UDP from the pause point.
26. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
27. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send a valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PAUSE and PLAY requests.

The DUT did not send mandatory header or field in the PAUSE response message (“CSeq”).

The DUT did not accept a Range header with a value of clock for PAUSE request.

The DUT did not pause streaming after PAUSE request.

The DUT did not resume streaming after PAUSE request from pause point.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for video track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for video track will be used as T2.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: On the step 20 (verification that no packets were sent out with a timestamp later than the Date-field): If Date-field is not supplied, the ONVIF Client will follow to the following logic:

If after receiving reply to the "PAUSE" command the ONVIF Client do not receive any RTP packet during 3 sec then step is considered as passed. Otherwise, the ONVIF Client will check that DUT do not send RTP packets within 3 sec after the last packet, which had been received in the first 3 sec timespan.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.13 PLAYBACK VIDEO STREAMING – PAUSE WITH RANGE

Test Label: Replay Viewing DUT video streaming using with Pause (with Range).

Test Case ID: REPLAY-3-1-16

ONVIF Core Specification Coverage: Playback, Pause.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify pause with Range of replay video streaming based on RTP/UDP Unicast Transport.

Pre-Requirement: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (RecordingToken = Token1, Stream = "RTP-Unicast", Transport. Protocol = "UDP").
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
9. The DUT sends 200 OK Response and SDP message (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters

- in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP control request with “onvif-replay” feature tag (“**Require: onvif-replay**”) to create a RTSP Session.
 13. The DUT sends 200 OK Response and Stream Information details.
 14. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-**”) control request to initiate the video streaming.
 15. The DUT sends 200 OK Response and RTP protocol information.
 16. The DUT transfers video streams over RTP/UDP.
 17. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation.
 18. ONVIF Client will invoke RTSP PAUSE (“**Range: clock = T1 + delta**”) control request to pause the video streaming.
 19. The DUT sends 200 OK Response.
 20. ONVIF Client verifies “CSeq” header field and optional “Date” header field if it is present in the PAUSE response message.
 21. The DUT pauses transferring of video streams after the specified time.
 22. ONVIF Client verifies that there are no RTP received after specified time **T1 + delta**.
 23. ONVIF Client will wait 2 s after pause of replay streaming and will invoke RTSP PLAY (“**Require: onvif-replay**”, **no Range**) control request to resume the video streaming.
 24. The DUT sends 200 OK Response and RTP protocol information.
 25. The DUT resumes video streams over RTP/UDP from the pause point.
 26. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
 27. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send a valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PAUSE and PLAY requests.

The DUT did not send mandatory headers or fields in the PAUSE response message (“CSeq”, “Date”).

The DUT did not accept a Range header with a value of clock for PAUSE request.

The DUT did not pause streaming after PAUSE request at specified point.

The DUT did not resume streaming after PAUSE request from posed point.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for video track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for video track will be used as T2.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.3.14 PLAYBACK VIDEO STREAMING – SEEK

Test Label: Verify Seek Scenario for DUT Video Streaming.

Test Case ID: REPLAY-3-1-17

ONVIF Core Specification Coverage: Immediate header field

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify Seek scenario for DUT video streaming.

Pre-Requisite: Recording with video track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start the DUT.
3. Get token **Token1** of recording which contains at least one track with video from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP").
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.
10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE

- in *sdp* - SDP information
 - in IPv4 – IP type
11. ONVIF Client will invoke RTSP SETUP request (“**Require: onvif-replay**”) with transport parameter as **RTP/UDP**.
 12. The DUT sends 200 OK message and the video stream information.
 13. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-**”, “**Rate-Control: no**”) control request to initiate the video streaming.
 14. The DUT sends 200 OK message and starts video streaming.
 15. The DUT sends RTP video stream to ONVIF Client over RTSP control connection.
 16. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation. ONVIF Client compares NTP time of the first frames with start time of the recording considering inaccuracy (inaccuracy value=2 s).
 17. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = MiddleT1-**”, “**Rate-Control: no**”, “**Immediate: yes**”, where MiddleT1 is the time point into the recording time range) control request to initiate the video streaming from other time.
 18. The DUT cancels previous PLAY command and immediately start playing from the **MiddleT1**.
 19. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation. ONVIF Client compares NTP time of the first frame with MiddleT1 considering inaccuracy (inaccuracy value=2 s).
 20. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = [MiddleT1-delta]-**”, “**Rate-Control: no**”, “**Immediate: yes**”) control request to initiate the video streaming from other time.
 21. The DUT cancels previous PLAY command and immediately start playing from the **MiddleT1-delta**.
 22. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation. ONVIF Client compares NTP time of the first frame with MiddleT1-delta considering inaccuracy (inaccuracy value=2 s).
 23. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = [MiddleT1+delta]-**”, “**Rate-Control: no**”, “**Immediate: yes**”) control request to initiate the video streaming from other time.
 24. The DUT cancels previous PLAY command and immediately start playing from the **MiddleT1+delta**.
 25. ONVIF Client verifies RTP header and RTP header extensions for video stream and render it after the validation. ONVIF Client compares NTP time of the first frame with MiddleT1+delta considering inaccuracy (inaccuracy value=2 s).
 26. ONVIF Client will invoke RTSP TEARDOWN control request at the end of video streaming to terminate the RTSP session.
 27. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send a valid **GetRecordingInformationResponse** message.

The DUT did not send **GetReplayUriResponse** message.

The DUT did not send a valid **GetReplayUriResponse** message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TEARDOWN requests.

RTSP Session was terminated by the DUT during video streaming.

The DUT did not accept an Immediate header for PLAY command.

The DUT did not send valid RTP header for video stream.

The DUT did not cancel the first PLAY command after PLAY command with “Immediate: yes”.

The DUT did not immediately start playing from the new location after PLAY command with “Immediate: yes”.

The DUT sent the first frame after PLAY response with NTP time different from the requested range at step 13 or step 16 or step 19 or step 22.

NOTE: The test tool will select MiddleT1, MiddleT1-delta and MiddleT1+delta values for PLAYS's range from the first minute of the recording.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.4 Playback Control – Audio Streaming

4.4.1 PLAYBACK AUDIO STREAMING – MEDIA CONTROL

Test Label: DUT RTSP Control Messages Verification for Replay Audio.

Test Case ID: REPLAY-3-2-1

ONVIF Core Specification Coverage: RTP data transfer via UDP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify RTSP control messages of the DUT for replay audio streaming.

Pre-Requisite: Recording with audio track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with audio from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP OPTIONS control request to get the RTSP methods supported by DUT.
9. The DUT sends 200 OK Response and list of supported RTSP methods.
10. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
11. The DUT sends 200 OK Response and SDP message (*sdp*).
12. Set *responseHeader* := header of response to DESCRIBE.
13. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information

- in IPv4 – IP type
14. ONVIF Client validates the session description information in the SDP message.
 15. ONVIF Client verifies tag with TrackReference in SDP message for each audio track.
 16. ONVIF Client will invoke RTSP SETUP control request with “onvif-replay” feature tag (“**Require: onvif-replay**”) to create a RTSP Session.
 17. The DUT sends 200 OK Response and Stream Information details.
 18. ONVIF Client verifies “Transport”, “Session” and “timeout” header fields in the SETUP response message.
 19. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-**”) control request to initiate the audio streaming.
 20. The DUT sends 200 OK Response and RTP protocol information.
 21. ONVIF Client verifies “Session”, “RTP-Info”, “seq”, “uri” and “rtpptime” header fields in the PLAY response message.
 22. The DUT transfers audio streams over UDP.
 23. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation.
 24. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
 25. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send mandatory RTSP commands DESCRIBE SETUP, PLAY, TEARDOWN, and OPTIONS in OPTIONS response.

The DUT sends SDP message with one or more audio track without the TrackReference in corresponding format (a:x-onvif-track:<TrackReference>).

DUT did not send correct audio stream information in the SDP message.

DUT did not send RTSP 200 OK response for RTSP OPTIONS, DESCRIBE, SETUP and PLAY requests.

DUT did not send mandatory headers or fields in the SETUP response message.

DUT did not send mandatory headers or fields in the PLAY response message.

RTSP Session was terminated by the DUT during audio streaming.

DUT did not send valid RTP header including replay header extension for audio stream.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for audio track will be used as **T1**.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for audio track will be used as **T2**.

NOTE: See A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.4.2 PLAYBACK AUDIO STREAMING – RTP-Unicast/UDP

Test Label: Replay Viewing DUT audio streaming using RTP-Unicast/UDP transport.

Test Case ID: REPLAY-3-2-2

ONVIF Core Specification Coverage: RTP data transfer via UDP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify playback audio streaming based on RTP/UDP Unicast Transport.

Pre-Requisite: Recording with audio track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with audio from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message
(RecordingInformation.RecordingToken = Token1,
RecordingInformation.EarliestRecording = T1,
RecordingInformation.LatestRecording = T2).
6. ONVIF Client will invoke GetReplayUriRequest (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP").
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).

10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP (“**Require: onvif-replay**”) request with transport parameter as **RTP/UDP**.
13. The DUT sends 200 OK message and the audio stream information.
14. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-**”) control request to initiate the audio streaming.
15. The DUT sends 200 OK message and starts audio streaming.
16. The DUT sends RTP audio stream to ONVIF Client over UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation.
18. ONVIF Client validates RTCP packets if The DUT sends theirs.
19. ONVIF Client will invoke RTSP TEARDOWN control request at the end of audio streaming to terminate the RTSP session.
20. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during audio streaming.

DUT did not send valid RTP header and RTP header extension for audio stream.

DUT sent an incorrect RTCP report.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for audio track will be used as **T1**.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for audio track will be used as **T2**.

NOTE: RTCP report will be validated only if it is received from the DUT.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.4.3 PLAYBACK AUDIO STREAMING – RTP-Unicast/RTSP/HTTP/TCP

Test Label: Replay Viewing DUT audio streaming using HTTP transport.

Test Case ID: REPLAY-3-2-3

ONVIF Core Specification Coverage: RTP/RTSP/HTTP/TCP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify playback audio streaming based on HTTP Transport.

Pre-Requirement: Recording with audio track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with audio from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "HTTP"**).
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type

12. ONVIF Client will invoke RTSP SETUP (“**Require: onvif-replay**”) request with transport parameter as **RTP/TCP**.
13. The DUT sends 200 OK message and the audio stream information.
14. ONVIF Client will invoke RTSP PLAY (“Require: onvif-replay”, “Range: clock = T1-”) control request to initiate the audio streaming.
15. The DUT sends 200 OK message and starts audio streaming.
16. The DUT sends RTP audio stream to ONVIF Client over HTTP.
17. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation.
18. ONVIF Client validates RTCP packets if The DUT sends theirs.
19. ONVIF Client will invoke RTSP TEARDOWN control request at the end of audio streaming to terminate the RTSP session.
20. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during audio streaming.

DUT did not send a valid RTP header and RTP header extension for audio stream.

DUT sent an incorrect RTCP report.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for audio track will be used as **T1**.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for audio track will be used as **T2**.

NOTE: RTCP report will be validated only if it is received from the DUT.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.4.4 PLAYBACK AUDIO STREAMING – RTP/RTSP/TCP

Test Label: Replay Viewing DUT audio streaming using RTP/RTSP/TCP transport.

Test Case ID: REPLAY-3-2-4

ONVIF Core Specification Coverage: RTP/RTSP/TCP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify playback audio streaming based on RTP/RTSP/TCP using RTSP tunnel.

Pre-Requisite: Recording with audio track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with audio from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "RTSP"**).
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP ("**Require: onvif-replay**") request with transport parameter as **RTP/TCP**.
13. The DUT sends 200 OK message and the audio stream information.
14. ONVIF Client will invoke RTSP PLAY ("Require: onvif-replay", "Range: clock = T1-") control request to initiate the audio streaming.

15. The DUT sends 200 OK message and starts audio streaming.
16. The DUT sends RTP audio stream to ONVIF Client over TCP.
17. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation.
18. ONVIF Client validates RTCP packets if The DUT sends theirs.
19. ONVIF Client will invoke RTSP TEARDOWN control request at the end of audio streaming to terminate the RTSP session.
20. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during audio streaming.

DUT did not send valid RTP header and RTP header extension for audio stream.

DUT sent an incorrect RTCP report.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for audio track will be used as **T1**.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for audio track will be used as **T2**.

NOTE: RTCP report will be validated only if it is received from the DUT.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.4.5 PLAYBACK AUDIO STREAMING – PLAY WITH RANGE

Test Label: Replay Audio Streaming from Specified Range.

Test Case ID: REPLAY-3-2-10

ONVIF Core Specification Coverage: Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify stop of replay audio streaming based on RTP/UDP Unicast Transport after the specified time Range was reached.

Pre-Requisite: Recording with audio track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with audio from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
9. The DUT sends 200 OK Response and SDP message (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP control request with "onvif-replay" feature tag (**"Require: onvif-replay"**) to create a RTSP Session.
13. The DUT sends 200 OK Response and Stream Information details.
14. ONVIF Client will invoke RTSP PLAY (**"Require: onvif-replay", "Range: clock = T1-T1 + delta"**) control request to initiate the audio streaming.
15. The DUT sends 200 OK Response and RTP protocol information.
16. DUT transfers audio streams over RTP/UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation.
18. DUT stops transferring of audio stream after the specified time **T1 + delta** was reached.
19. ONVIF Client verifies that there are no RTP received after specified time [T1 + delta].
20. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.

21. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:

PASS –

DUT passed all assertions.

FAIL –

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP OPTIONS, DESCRIBE, SETUP and PLAY requests.

DUT did not stop audio streaming after specified time.

DUT did not send a valid RTP header in audio stream (RTP sequence number not in the orders of delivery, timestamps correspond to playback timing considering inaccuracy).

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for audio track will be used as **T1**.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for audio track will be used as **T2**.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.4.6 PLAYBACK AUDIO STREAMING – RATECONTROL

Test Label: Compare Replay Audio Streaming with different Rate-Control.

Test Case ID: REPLAY-3-2-11

ONVIF Core Specification Coverage: Rate-Control header field.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify applying of Rate-Control value for playback audio streaming.

Pre-Requisite: Recording with audio track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with audio from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "HTTP"**).
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.
10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
11. ONVIF Client will invoke RTSP SETUP request ("**Require: onvif-replay**") with transport parameter as **RTP/AVP/TCP**.
12. The DUT sends 200 OK message and the audio stream information.
13. ONVIF Client will invoke RTSP PLAY ("**Require: onvif-replay**", "**Range: clock = T1-T1+Delta**", "**Rate-Control: yes**") control request to initiate the audio streaming.
14. The DUT sends 200 OK message and starts audio streaming.
15. The DUT sends RTP audio stream to ONVIF Client over HTTP.
16. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation.
17. ONVIF Client waits while replay streaming is played and calculates play duration **D1**.
18. ONVIF Client will invoke RTSP PLAY ("**Require: onvif-replay**", "**Range: clock = T1-T1+Delta**", "**Rate-Control: no**") control request to initiate the audio streaming.
19. The DUT sends 200 OK message and starts audio streaming.
20. The DUT sends RTP audio stream to ONVIF Client over HTTP.
21. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation.
22. ONVIF Client waits while replay streaming is played and calculates play duration **D2**.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY requests.

RTSP Session was terminated by the DUT during audio streaming.

DUT did not accept Rate-Control header for PLAY command.

DUT did not send a valid RTP header for audio stream with rate control (RTP sequence numbers are not in the order of delivery, timestamps correspond to playback timing considering inaccuracy).

DUT did not send a valid RTP header for audio stream without rate control (RTP sequence numbers are not in the order of delivery, timestamps correspond to original sampling times of recorded frames considering inaccuracy).

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.4.7 PLAYBACK AUDIO STREAMING – IMMEDIATE HEADER

Test Label: Compare Replay Audio streaming with Immediate header.

Test Case ID: REPLAY-3-2-12

ONVIF Core Specification Coverage: Immediate header field

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To compare RTSP playback with Immediate header.

Pre-Requisite: Recording with audio track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with audio from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.

5. Verify the GetRecordingInformationResponse message (RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2).
6. ONVIF Client will invoke GetReplayUriRequest message (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP").
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.
10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
11. ONVIF Client will invoke RTSP SETUP request ("Require: onvif-replay") with transport parameter as RTP/UDP.
12. The DUT sends 200 OK message and the audio stream information.
13. ONVIF Client will invoke RTSP PLAY ("Require: onvif-replay", "Range: clockT1 -") control request to initiate the audio streaming.
14. The DUT sends 200 OK message and starts audio streaming.
15. The DUT sends RTP audio stream to ONVIF Client over UDP.
16. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation.
17. ONVIF Client will invoke RTSP PLAY ("Require: onvif-replay", "Range: clock [T1+delta]-", "Immediate: yes") control request to initiate the second audio streaming from other time.
18. DUT cancelling existing first PLAY command and immediately start playing from the T1+delta.
19. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation. ONVIF Client verifies that first packet sent from the new location have the D (discontinuity) bit set in its RTP extension header.
20. ONVIF Client will invoke RTSP TEARDOWN control request at the end of audio streaming to terminate the RTSP session.
21. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:

PASS –

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during audio streaming.

DUT did not accept an Immediate header for PLAY command.

DUT did not send valid RTP header for audio stream.

DUT did not cancel the first PLAY command after PLAY command with “Immediate: yes”.

DUT did not immediately start playing from the new location after PLAY command with “Immediate: yes”.

DUT did not send a valid RTP header from the new location (first packet did not have the D (discontinuity) bit set in its RTP extension header).

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.4.8 PLAYBACK AUDIO STREAMING – PAUSE WITHOUT RANGE

Test Label: Replay Viewing DUT audio streaming using with Pause (no Range).

Test Case ID: REPLAY-3-2-14

ONVIF Core Specification Coverage: Playback, Pause.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify pause without Range of replay audio streaming based on RTP/UDP Unicast Transport.

Pre-Requisite: Recording with audio track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with audio from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = “Token1”**) to retrieve information about Recording.

5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
9. The DUT sends 200 OK Response and SDP message (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP control request with "onvif-replay" feature tag (**"Require: onvif-replay"**) to create a RTSP Session with transport parameter as **„RTP/UDP“**.
13. The DUT sends 200 OK Response and Stream Information details.
14. ONVIF Client will invoke RTSP PLAY ("Require: onvif-replay", "Range: clock = T1-") control request to initiate the audio streaming.
15. The DUT sends 200 OK Response and RTP protocol information.
16. The DUT transfers audio streams over RTP/UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation.
18. ONVIF Client will invoke RTSP PAUSE (**no Range header**) control request to pause the audio streaming.
19. The DUT sends 200 OK Response.
20. ONVIF Client verifies "CSeq" header field and optional "Date" header field if it is present in the PAUSE response message.
21. The DUT pauses transferring of audio streams and DUT sets the pause point to the current normal play time.
22. ONVIF Client verifies that there are no RTP received after this.
23. ONVIF Client will wait 2s after pause of replay streaming and will invoke RTSP PLAY (**"Require: onvif-replay", no Range**) control request to resume the audio streaming.
24. The DUT sends 200 OK Response and RTP protocol information.
25. The DUT resumes audio streams over RTP/UDP from the pause point.
26. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
27. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:

PASS –

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PAUSE and PLAY requests.

DUT did not send mandatory header or field in the PAUSE response message ("CSeq").

DUT did not accept a Range header with a value of clock for PAUSE request.

DUT did not pause streaming after PAUSE request.

DUT did not resume streaming after PAUSE request from pause point.

Note: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for audio track will be used as **T1**.

Note: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for audio track will be used as **T2**.

Note: On the step 20 (verification that no packets were sent out with a timestamp later than the Date-field): If Date-field is not supplied, the ONVIF Client will follow to the following logic:

If after receiving reply to the "PAUSE" command the ONVIF Client do not receive any RTP packet during 3 sec then step is considered as passed. Otherwise, the ONVIF Client will check that DUT do not send RTP packets within 3 sec after the last packet, which had been received in the first 3 sec timespan.

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.4.9 PLAYBACK AUDIO STREAMING – PAUSE WITH RANGE

Test Label: Replay Viewing DUT audio streaming using with Pause (with Range).

Test Case ID: REPLAY-3-2-15

ONVIF Core Specification Coverage: Playback, Pause.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify pause with Range of replay audio streaming based on RTP/UDP Unicast Transport.

Pre-Requisite: Recording with audio track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with audio from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
9. The DUT sends 200 OK Response and SDP message (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP control request with "onvif-replay" feature tag (**"Require: onvif-replay"**) to create a RTSP Session.
13. The DUT sends 200 OK Response and Stream Information details.
14. ONVIF Client will invoke RTSP PLAY (**"Require: onvif-replay", "Range: clock = T1-"**) control request to initiate the audio streaming.
15. The DUT sends 200 OK Response and RTP protocol information.
16. DUT transfers audio streams over RTP/UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation.

18. ONVIF Client will invoke RTSP PAUSE (“**Range: clock = T1 + delta**”) control request to pause the audio streaming.
19. The DUT sends 200 OK Response.
20. ONVIF Client verifies “CSeq” header field and optional “Date” header field if it is present in the PAUSE response message.
21. DUT pauses transferring of audio streams after the specified time.
22. ONVIF Client verifies that there are no RTP received after specified time **T1 + delta**.
23. ONVIF Client will wait 2 s after pause of replay streaming and will invoke RTSP PLAY (“**Require: onvif-replay**”, **no Range**) control request to resume the audio streaming.
24. The DUT sends 200 OK Response and RTP protocol information.
25. DUT resumes audio streams over RTP/UDP from the pause point.
26. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
27. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PAUSE and PLAY requests.

DUT did not send mandatory header or field in the PAUSE response message (“CSeq”).

DUT did not accept a Range header with a value of clock for PAUSE request.

DUT did not pause streaming after PAUSE request at specified point.

DUT did not resume streaming after PAUSE request from posed point.

Note: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for audio track will be used as **T1**.

Note: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for audio track will be used as **T2**.

Note: See Annex A.1 for Invalid RTP header definition.

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.4.10 PLAYBACK AUDIO STREAMING – SEEK

Test Label: Verify Seek Scenario for DUT Audio Streaming.

Test Case ID: REPLAY-3-2-16

ONVIF Core Specification Coverage: Immediate header field

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify Seek scenario for DUT audio streaming.

Pre-Requisite: Recording with audio track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start the DUT.
3. Get token **Token1** of recording which contains at least one track with audio from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.
10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
11. ONVIF Client will invoke RTSP SETUP request ("**Require: onvif-replay**") with transport parameter as **RTP/UDP**.
12. The DUT sends 200 OK message and the audio stream information.

13. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-**”, “**Rate-Control: no**”) control request to initiate the audio streaming.
14. The DUT sends 200 OK message and starts audio streaming.
15. The DUT sends RTP audio stream to ONVIF Client over RTSP control connection.
16. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation. ONVIF Client compares NTP time of the first frames with start time of the recording considering inaccuracy (inaccuracy value=2 s).
17. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = MiddleT1-**”, “**Rate-Control: no**”, “**Immediate: yes**”, where MiddleT1 is the time point into the recording time range) control request to initiate the audio streaming from other time.
18. The DUT cancels previous PLAY command and immediately start playing from the **MiddleT1**.
19. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation. ONVIF Client compares NTP time of the first frame with MiddleT1 considering inaccuracy (inaccuracy value=2 s).
20. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = [MiddleT1 - delta]-**”, “**Rate-Control: no**”, “**Immediate: yes**”) control request to initiate the audio streaming from other time.
21. The DUT cancels previous PLAY command and immediately start playing from the **MiddleT1-delta**.
22. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation. ONVIF Client compares NTP time of the first frame with MiddleT1-delta considering inaccuracy (inaccuracy value=2 s).
23. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = [MiddleT1 + delta]-**”, “**Rate-Control: no**”, “**Immediate: yes**”) control request to initiate the audio streaming from other time.
24. The DUT cancels previous PLAY command and immediately start playing from the **MiddleT1 + delta**.
25. ONVIF Client verifies RTP header and RTP header extensions for audio stream and render it after the validation. ONVIF Client compares NTP time of the first frame with MiddleT1+delta considering inaccuracy (inaccuracy value=2 s).
26. ONVIF Client will invoke RTSP TEARDOWN control request at the end of audio streaming to terminate the RTSP session.
27. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send a valid **GetRecordingInformationResponse** message.

The DUT did not send **GetReplayUriResponse** message.

The DUT did not send a valid **GetReplayUriResponse** message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TEARDOWN requests.

RTSP Session was terminated by the DUT during audio streaming.

The DUT did not accept an Immediate header for PLAY command.

The DUT did not send a valid RTP header for audio stream.

The DUT did not cancel the first PLAY command after PLAY command with "Immediate: yes".

The DUT sent the first frame after PLAY response with NTP time different from the requested range at step 13 or step 16 or step 19 or step 22.

Note: The test tool will select MiddleT1, MiddleT1-delta and MiddleT1+delta values for PLAYS's range from the first minute of the recording.

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5 Playback Control – Metadata Streaming

4.5.1 PLAYBACK METADATA STREAMING – MEDIA CONTROL

Test Label: DUT RTSP Control Messages Verification for Replay Metadata.

Test Case ID: REPLAY-3-3-1

ONVIF Core Specification Coverage: RTP data transfer via UDP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify RTSP control messages of the DUT for replay metadata streaming.

Pre-Requisite: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP OPTIONS control request to get the RTSP methods supported by DUT.
9. The DUT sends 200 OK Response and list of supported RTSP methods.
10. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
11. The DUT sends 200 OK Response and SDP message (*sdp*).
12. Set *responseHeader* := header of response to DESCRIBE.
13. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE

- in *sdp* - SDP information
 - in IPv4 – IP type
14. ONVIF Client validates the session description information in the SDP message.
 15. ONVIF Client verifies tag with TrackReference in SDP message for each metadata track.
 16. ONVIF Client will invoke RTSP SETUP control request with “onvif-replay” feature tag (“**Require: onvif-replay**”) to create a RTSP Session.
 17. The DUT sends 200 OK Response and Stream Information details.
 18. ONVIF Client verifies “Transport”, “Session” and “timeout” header fields in the SETUP response message.
 19. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-**”) control request to initiate the metadata streaming.
 20. The DUT sends 200 OK Response and RTP protocol information.
 21. ONVIF Client verifies “Session”, “RTP-Info”, “seq”, “uri” and “rtptime” header fields in the PLAY response message.
 22. The DUT transfers metadata streams over UDP.
 23. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
 24. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
 25. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send mandatory RTSP commands DESCRIBE SETUP, PLAY, TEARDOWN, and OPTIONS in OPTIONS response.

The DUT sends SDP message with one or more metadata track without the TrackReference in corresponding format (a:x-onvif-track:<TrackReference>).

DUT did not send correct metadata stream information in the SDP message.

DUT did not send RTSP 200 OK response for RTSP OPTIONS, DESCRIBE, SETUP and PLAY requests.

DUT did not send mandatory headers or fields in the SETUP response message.

DUT did not send mandatory headers or fields in the PLAY response message.

RTSP Session was terminated by the DUT during metadata streaming.

DUT did not send a valid RTP header including replay header extension for metadata stream.

Note: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for metadata track will be used as **T1**.

Note: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for metadata track will be used as **T2**.

Note: See A.1 for Invalid RTP header definition.

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5.2 PLAYBACK METADATA STREAMING – RTP-Unicast/UDP

Test Label: Replay Viewing DUT metadata streaming using RTP-Unicast/UDP transport.

Test Case ID: REPLAY-3-3-2

ONVIF Core Specification Coverage: RTP data transfer via UDP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify playback metadata streaming based on RTP/UDP Unicast Transport.

Pre-Requisite: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.

5. Verify the `GetRecordingInformationResponse` message
(`RecordingInformation.RecordingToken = Token1`,
`RecordingInformation.EarliestRecording = T1`,
`RecordingInformation.LatestRecording = T2`).
6. ONVIF Client will invoke `GetReplayUriRequest` (`RecordingToken = Token1`, `Stream = "RTP-Unicast"`, `Transport.Protocol = "UDP"`).
7. Verify the `GetReplayUriResponse` from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP ("**Require: onvif-replay**") request with transport parameter as **RTP/UDP**.
13. The DUT sends 200 OK message and the metadata stream information.
14. ONVIF Client will invoke RTSP PLAY ("**Require: onvif-replay**", "**Range: clock = T1-**") control request to initiate the metadata streaming.
15. The DUT sends 200 OK message and starts metadata streaming.
16. The DUT sends RTP metadata stream to ONVIF Client over UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
18. ONVIF Client validates RTCP packets if The DUT sends theirs.
19. ONVIF Client will invoke RTSP TEARDOWN control request at the end of metadata streaming to terminate the RTSP session.
20. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid `GetRecordingInformationResponse` message.

DUT did not send `GetReplayUriResponse` message.

DUT did not send a valid `GetReplayUriResponse` message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during metadata streaming.

DUT did not send a valid RTP header and RTP header extension for metadata stream.

DUT sent an incorrect RTCP report.

Note: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for metadata track will be used as **T1**.

Note: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for metadata track will be used as **T2**.

Note: RTCP report will be validated only if it is received from the DUT.

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5.3 PLAYBACK METADATA STREAMING – RTP- Unicast/RTSP/HTTP/TCP

Test Label: Replay Viewing DUT metadata streaming using HTTP transport.

Test Case ID: REPLAY-3-3-3

ONVIF Core Specification Coverage: RTP/RTSP/HTTP/TCP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify playback metadata streaming based on HTTP Transport.

Pre-Requisite: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.

5. Verify the `GetRecordingInformationResponse` message
**(RecordingInformation.RecordingToken = Token1,
RecordingInformation.EarliestRecording = T1,
RecordingInformation.LatestRecording = T2)**.
6. ONVIF Client will invoke `GetReplayUriRequest` (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "HTTP"**).
7. Verify the `GetReplayUriResponse` from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP (**"Require: onvif-replay"**) request with transport parameter as **RTP/TCP**.
13. The DUT sends 200 OK message and the metadata stream information.
14. ONVIF Client will invoke RTSP PLAY (**"Require: onvif-replay", "Range: clock = T1-"**) control request to initiate the metadata streaming.
15. The DUT sends 200 OK message and starts metadata streaming.
16. The DUT sends RTP metadata stream to ONVIF Client over HTTP.
17. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
18. ONVIF Client validates RTCP packets if The DUT sends theirs.
19. ONVIF Client will invoke RTSP TEARDOWN control request at the end of metadata streaming to terminate the RTSP session.
20. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid `GetRecordingInformationResponse` message.

DUT did not send `GetReplayUriResponse` message.

DUT did not send a valid `GetReplayUriResponse` message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during metadata streaming.

DUT did not send a valid RTP header and RTP header extension for metadata stream.

DUT sent an incorrect RTCP report.

Note: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for metadata track will be used as **T1**.

Note: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for metadata track will be used as **T2**.

Note: RTCP report will be validated only if it is received from the DUT.

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5.4 PLAYBACK METADATA STREAMING – RTP/RTSP/TCP

Test Label: Replay Viewing DUT metadata streaming using RTP/RTSP/TCP transport.

Test Case ID: REPLAY-3-3-4

ONVIF Core Specification Coverage: RTP/RTSP/TCP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify playback metadata streaming based on RTP/RTSP/TCP using RTSP tunnel.

Pre-Requisite: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
1. Start a DUT.
2. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
2. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.

3. Verify the GetRecordingInformationResponse message (RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2).
4. ONVIF Client will invoke GetReplayUriRequest (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "RTSP").
5. Verify the GetReplayUriResponse from the DUT.
6. ONVIF Client will invoke RTSP DESCRIBE request.
7. The DUT sends 200 OK message and SDP information (*sdp*).
8. Set *responseHeader* := header of response to DESCRIBE.
9. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
10. ONVIF Client will invoke RTSP SETUP ("Require: onvif-replay") request with transport parameter as RTP/TCP.
11. The DUT sends 200 OK message and the metadata stream information.
12. ONVIF Client will invoke RTSP PLAY ("Require: onvif-replay", "Range: clock = T1-") control request to initiate the metadata streaming.
13. The DUT sends 200 OK message and starts metadata streaming.
14. The DUT sends RTP metadata stream to ONVIF Client over TCP.
15. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
16. ONVIF Client validates RTCP packets if The DUT sends theirs.
17. ONVIF Client will invoke RTSP TEARDOWN control request at the end of metadata streaming to terminate the RTSP session.
18. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:

PASS –

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during metadata streaming.

DUT did not send a valid RTP header and RTP header extension for metadata stream.

DUT sent an incorrect RTCP report.

Note: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for metadata track will be used as **T1**.

Note: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for metadata track will be used as **T2**.

Note: RTCP report will be validated only if it is received from the DUT.

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5.5 REVERSE PLAYBACK METADATA STREAMING – RTP- Unicast/UDP

Test Label: Reverse Replay Viewing DUT metadata streaming using RTP-Unicast/UDP transport.

Test Case ID: REPLAY-3-3-5

ONVIF Core Specification Coverage: RTP data transfer via UDP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify reverse playback metadata streaming based on RTP/UDP Unicast Transport.

Pre-Requisite: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.

5. Verify the GetRecordingInformationResponse message (RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2).
6. ONVIF Client will invoke GetReplayUriRequest (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP").
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP ("Require: onvif-replay") request with transport parameter as RTP/UDP.
13. The DUT sends 200 OK message and the metadata stream information.
14. ONVIF Client will invoke RTSP PLAY ("Require: onvif-replay", "Range: clock = T2-", "Scale: -1.0") control request to initiate the metadata streaming.
15. The DUT sends 200 OK message and starts metadata streaming.
16. The DUT sends RTP metadata stream to ONVIF Client over UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
18. ONVIF Client verifies Packet Transmission Order (GOPs are sent in reverse order, but packets within a GOP are sent in forward order).
19. ONVIF Client validates the RTP sequence numbers. Verify that the RTP sequence numbers of packets are increment monotonically in the order of delivery, not in the intended order of playback.
20. ONVIF Client checks Timestamps (the RTP timestamps shall indicate times at which each frame should be rendered at the client).
21. ONVIF Client validates RTCP packets if The DUT sends theirs.
22. ONVIF Client will invoke RTSP TEARDOWN control request at the end of metadata streaming to terminate the RTSP session.
23. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:

PASS –

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during metadata streaming.

DUT did not send a valid RTP header and RTP header extension for metadata stream.

DUT sent GOPs or packets within a GOP in wrong order.

DUT sent RTP packets with invalid sequence number.

DUT sent RTP packets with invalid timestamp.

DUT sent an incorrect RTCP report.

Note: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for metadata track will be used as **T1**.

Note: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for metadata track will be used as **T2**.

Note: RTCP report will be validated only if it is received from the DUT.

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5.6 REVERSE PLAYBACK METADATA STREAMING – RTP- Unicast/RTSP/HTTP/TCP

Test Label: Reverse Replay Viewing DUT metadata streaming using HTTP transport.

Test Case ID: REPLAY-3-3-6

ONVIF Core Specification Coverage: RTP/RTSP/HTTP/TCP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify reverse playback metadata streaming based on HTTP Transport.

Pre-Requirement: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "HTTP"**).
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP (**"Require: onvif-replay"**) request with transport parameter as **RTP/TCP**.
13. The DUT sends 200 OK message and the metadata stream information.
14. ONVIF Client will invoke RTSP PLAY (**"Require: onvif-replay", "Range: clock = T2-", "Scale: -1.0"**) control request to initiate the metadata streaming.
15. The DUT sends 200 OK message and starts metadata streaming.
16. The DUT sends RTP metadata stream to ONVIF Client over HTTP.
17. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
18. ONVIF Client verifies Packet Transmission Order (GOPs are sent in reverse order, but packets within a GOP are sent in forward order).
19. ONVIF Client validates the RTP sequence numbers. Verify that the RTP sequence numbers of packets are increment monotonically in the order of delivery, not in the intended order of playback.
20. ONVIF Client checks Timestamps (the RTP timestamps shall indicate times at which each frame should be rendered at the client).
21. ONVIF Client validates RTCP packets if The DUT sends theirs.

22. ONVIF Client will invoke RTSP TEARDOWN control request at the end of metadata streaming to terminate the RTSP session.
23. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during metadata streaming.

DUT did not send a valid RTP header and RTP header extension for metadata stream.

DUT sent GOPs or packets within a GOP in wrong order.

DUT sent RTP packets with invalid sequence number.

DUT sent RTP packets with invalid timestamp.

DUT sent an incorrect RTCP report.

Note: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for metadata track will be used as **T1**.

Note: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for metadata track will be used as **T2**.

Note: RTCP report will be validated only if it is received from the DUT.

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5.7 REVERSE PLAYBACK METADATA STREAMING –

RTP/RTSP/TCP

Test Label: Reverse Replay Viewing DUT metadata streaming using RTP/RTSP/TCP transport.

Test Case ID: REPLAY-3-3-7

ONVIF Core Specification Coverage: RTP/RTSP/TCP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify reverse playback metadata streaming based on RTP/RTSP/TCP using RTSP tunnel.

Pre-Requisite: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "RTSP"**).
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type

12. ONVIF Client will invoke RTSP SETUP (“**Require: onvif-replay**”) request with transport parameter as **RTP/TCP**.
13. The DUT sends 200 OK message and the metadata stream information.
14. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T2-**”, “**Scale: -1.0**”) control request to initiate the metadata streaming.
15. The DUT sends 200 OK message and starts metadata streaming.
16. The DUT sends RTP metadata stream to ONVIF Client over TCP.
17. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
18. ONVIF Client verifies Packet Transmission Order (GOPs are sent in reverse order, but packets within a GOP are sent in forward order).
19. ONVIF Client validates the RTP sequence numbers. Verify that the RTP sequence numbers of packets are increment monotonically in the order of delivery, not in the intended order of playback.
20. ONVIF Client checks Timestamps (the RTP timestamps shall indicate times at which each frame should be rendered at the client).
21. ONVIF Client validates RTCP packets if The DUT sends theirs.
22. ONVIF Client will invoke RTSP TEARDOWN control request at the end of metadata streaming to terminate the RTSP session.
23. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send s valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send s valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during metadata streaming.

DUT sent GOPs or packets within a GOP in wrong order.

DUT sent RTP packets with invalid sequence number.

DUT sent RTP packets with invalid timestamp.

DUT did not send s valid RTP header and RTP header extension for metadata stream.

DUT sent incorrect RTCP report.

Note: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for metadata track will be used as **T1**.

Note: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for metadata track will be used as **T2**.

Note: RTCP report will be validated only if it is received from the DUT.

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5.8 PLAYBACK METADATA STREAMING – PLAY WITH RANGE

Test Label: Replay metadata Streaming from Specified Range.

Test Case ID: REPLAY-3-3-10

ONVIF Core Specification Coverage: Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify stop of replay metadata streaming based on RTP/UDP Unicast Transport after the specified time Range was reached.

Pre-Requisite: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport. Protocol = "UDP"**).

7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
9. The DUT sends 200 OK Response and SDP message (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP control request with “onvif-replay” feature tag (“**Require: onvif-replay**”) to create a RTSP Session.
13. The DUT sends 200 OK Response and Stream Information details.
14. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-T1 + delta**”) control request to initiate the metadata streaming.
15. The DUT sends 200 OK Response and RTP protocol information.
16. DUT transfers metadata streams over RTP/UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
18. DUT stops transferring of metadata stream after the specified time **T1 + delta** was reached.
19. ONVIF Client verifies that there are no RTP received after specified time [**T1 + delta**].
20. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
21. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP OPTIONS, DESCRIBE, SETUP and PLAY requests.

DUT did not stop metadata streaming after specified time.

DUT did not send valid RTP header in metadata stream (RTP sequence number not in the orders of delivery, timestamps correspond to playback timing considering inaccuracy).

Note: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for metadata

track will be used as **T1**.

Note: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for metadata track will be used as **T2**.

Note: See Annex A.1 for Invalid RTP header definition.

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5.9 PLAYBACK METADATA STREAMING – RATECONTROL

Test Label: Compare Replay metadata Streaming with different Rate-Control.

Test Case ID: REPLAY-3-3-11

ONVIF Core Specification Coverage: Rate-Control header field.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify applying of Rate-Control value for playback metadata streaming.

Pre-Requisite: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "HTTP"**).
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.

10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
11. ONVIF Client will invoke RTSP SETUP request (“**Require: onvif-replay**”) with transport parameter as **RTP/AVP/TCP**.
12. The DUT sends 200 OK message and the metadata stream information.
13. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-T1+Delta**”, “**Rate-Control: yes**”) control request to initiate the metadata streaming.
14. The DUT sends 200 OK message and starts metadata streaming.
15. The DUT sends RTP metadata stream to ONVIF Client over HTTP.
16. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
17. ONVIF Client waits while replay streaming is played and calculates play duration **D1**.
18. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-T1+Delta**”, “**Rate-Control: no**”) control request to initiate the metadata streaming.
19. The DUT sends 200 OK message and starts metadata streaming.
20. The DUT sends RTP metadata stream to ONVIF Client over HTTP.
21. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
22. ONVIF Client waits while replay streaming is played and calculates play duration **D2**.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY requests.

RTSP Session was terminated by the DUT during metadata streaming.

DUT did not accept Rate-Control header for PLAY command.

DUT did not send a valid RTP header for metadata stream with rate control (RTP sequence numbers are not in the order of delivery, timestamps correspond to playback timing considering inaccuracy).

DUT did not send a valid RTP header for metadata stream without rate control (RTP

sequence numbers are not in the order of delivery, timestamps correspond to original sampling times of recorded frames considering inaccuracy).

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5.10 PLAYBACK METADATA STREAMING – IMMEDIATE HEADER

Test Label: Compare Replay metadata streaming with Immediate header.

Test Case ID: REPLAY-3-3-12

ONVIF Core Specification Coverage: Immediate header field

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To compare RTSP playback with Immediate header.

Pre-Requisite: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport. Protocol = "UDP"**).
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.
10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type

11. ONVIF Client will invoke RTSP SETUP request (“**Require: onvif-replay**”) with transport parameter as **RTP/UDP**.
12. The DUT sends 200 OK message and the metadata stream information.
13. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clockT1 -**”) control request to initiate the metadata streaming.
14. The DUT sends 200 OK message and starts metadata streaming.
15. The DUT sends RTP metadata stream to ONVIF Client over UDP.
16. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
17. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock [T1+delta]-**”, “**Immediate: yes**”) control request to initiate the second metadata streaming from other time.
18. DUT cancelling existing first PLAY command and immediately start playing from the **T1+delta**.
19. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation. ONVIF Client verifies that first packet sent from the new location have the D (discontinuity) bit set in its RTP extension header.
20. ONVIF Client will invoke RTSP TEARDOWN control request at the end of metadata streaming to terminate the RTSP session.
21. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during metadata streaming.

DUT did not accept an Immediate header for PLAY command.

DUT did not send a valid RTP header for metadata stream.

DUT did not cancel the first PLAY command after PLAY command with “Immediate: yes”.

DUT did not immediately start playing from the new location after PLAY command with “Immediate: yes”.

DUT did not send a valid RTP header from the new location (first packet did not have the D (discontinuity) bit set in its RTP extension header).

Note: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5.11 PLAYBACK METADATA STREAMING – PAUSE WITHOUT RANGE

Test Label: Replay Viewing DUT metadata streaming using with Pause (no Range).

Test Case ID: REPLAY-3-3-14

ONVIF Core Specification Coverage: Playback, Pause.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify pause without Range of replay metadata streaming based on RTP/UDP Unicast Transport.

Pre-Requisite: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
9. The DUT sends 200 OK Response and SDP message (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.

11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP control request with “onvif-replay” feature tag (“**Require: onvif-replay**”) to create a RTSP Session with transport parameter as „RTP/UDP”.
13. The DUT sends 200 OK Response and Stream Information details.
14. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-**”) control request to initiate the metadata streaming.
15. The DUT sends 200 OK Response and RTP protocol information.
16. DUT transfers metadata streams over RTP/UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
18. ONVIF Client will invoke RTSP PAUSE (**no Range header**) control request to pause the metadata streaming.
19. The DUT sends 200 OK Response.
20. ONVIF Client verifies “CSeq” header field and optional “Date” header field if it is present in the PAUSE response message.
21. DUT pauses transferring of metadata streams and DUT sets the pause point to the current normal play time.
22. ONVIF Client verifies that there are no RTP received after this.
23. ONVIF Client will wait 2s after pause of replay streaming and will invoke RTSP PLAY (“**Require: onvif-replay**”, **no Range**) control request to resume the metadata streaming.
24. The DUT sends 200 OK Response and RTP protocol information.
25. DUT resumes metadata streams over RTP/UDP from the pause point.
26. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
27. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PAUSE and PLAY

requests.

DUT did not send mandatory header or field in the PAUSE response message ("CSeq").

DUT did not accept a Range header with a value of clock for PAUSE request.

DUT did not pause streaming after PAUSE request.

DUT did not resume streaming after PAUSE request from pause point.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for metadata track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for metadata track will be used as T2.

NOTE: On the step 20 (verification that no packets were sent out with a timestamp later than the Date-field): If Date-field is not supplied, the ONVIF Client will follow to the following logic:

If after receiving reply to the "PAUSE" command the ONVIF Client do not receive any RTP packet during 3 sec then stepis considered as passed. Otherwise, the ONVIF Client will check that DUT do not send RTP packets within 3 sec after the last packet, which had been received in the first 3 sec timespan.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5.12 PLAYBACK METADATA STREAMING – PAUSE WITH RANGE

Test Label: Replay Viewing DUT metadata streaming using with Pause (with Range).

Test Case ID: REPLAY-3-3-15

ONVIF Core Specification Coverage: Playback, Pause.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify pause with Range of replay metadata streaming based on RTP/UDP Unicast Transport.

Pre-Requisite: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.

3. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
9. The DUT sends 200 OK Response and SDP message (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP control request with "onvif-replay" feature tag (**"Require: onvif-replay"**) to create a RTSP Session.
13. The DUT sends 200 OK Response and Stream Information details.
14. ONVIF Client will invoke RTSP PLAY (**"Require: onvif-replay", "Range: clock = T1-"**) control request to initiate the metadata streaming.
15. The DUT sends 200 OK Response and RTP protocol information.
16. DUT transfers metadata streams over RTP/UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation.
18. ONVIF Client will invoke RTSP PAUSE (**"Range: clock = T1 + delta"**) control request to pause the metadata streaming.
19. The DUT sends 200 OK Response.
20. ONVIF Client verifies "CSeq" header field and optional "Date" header field if it is present in the PAUSE response message.
21. DUT pauses transferring of metadata streams after the specified time.
22. ONVIF Client verifies that there are no RTP received after specified time **T1 + delta**.
23. ONVIF Client will wait 2 s after pause of replay streaming and will invoke RTSP PLAY (**"Require: onvif-replay", no Range**) control request to resume the metadata streaming.
24. The DUT sends 200 OK Response and RTP protocol information.
25. DUT resumes metadata streams over RTP/UDP from the pause point.
26. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
27. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PAUSE and PLAY requests.

DUT did not send mandatory header or field in the PAUSE response message (“CSeq”).

DUT did not accept a Range header with a value of clock for PAUSE request.

DUT did not pause streaming after PAUSE request at specified point.

DUT did not resume streaming after PAUSE request from posed point.

NOTE: If RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for metadata track will be used as T1.

NOTE: If RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for metadata track will be used as T2.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.5.13 PLAYBACK METADATA STREAMING – SEEK

Test Label: Verify Seek Scenario for DUT metadata Streaming.

Test Case ID: REPLAY-3-3-16

ONVIF Core Specification Coverage: Immediate header field

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify Seek scenario for DUT metadata streaming.

Pre-Requisite: Recording with metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start the DUT.
3. Get token **Token1** of recording which contains at least one track with metadata from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.
10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
11. ONVIF Client will invoke RTSP SETUP request ("**Require: onvif-replay**") with transport parameter as **RTP/UDP**.
12. The DUT sends 200 OK message and the metadata stream information.
13. ONVIF Client will invoke RTSP PLAY ("**Require: onvif-replay**", "**Range: clock = T1-**", "**Rate-Control: no**") control request to initiate the metadata streaming.
14. The DUT sends 200 OK message and starts metadata streaming.
15. The DUT sends RTP metadata stream to ONVIF Client over RTSP control connection.

16. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation. ONVIF Client checks that NTP time of the first frames are within [T1, T2] interval.
17. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = MiddleT1-**”, “**Rate-Control: no**”, “**Immediate: yes**”, where MiddleT1 is the time point into the recording time range) control request to initiate the metadata streaming from other time.
18. The DUT cancels previous PLAY command and immediately start playing from the **MiddleT1**.
19. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation. ONVIF Client checks that NTP time of the first frames are within [MiddleT1, T2] interval.
20. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = [MiddleT1 - delta]-**”, “**Rate-Control: no**”, “**Immediate: yes**”) control request to initiate the metadata streaming from other time.
21. The DUT cancels previous PLAY command and immediately start playing from the **MiddleT1-delta**.
22. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation. ONVIF Client checks that NTP time of the first frames are within [MiddleT1-delta, T2] interval.
23. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = [MiddleT1+delta]-**”, “**Rate-Control: no**”, “**Immediate: yes**”) control request to initiate the metadata streaming from other time.
24. The DUT cancels previous PLAY command and immediately start playing from the **MiddleT1+delta**.
25. ONVIF Client verifies RTP header and RTP header extensions for metadata stream and renders it after the validation. ONVIF Client checks that NTP time of the first frames are within [MiddleT1+delta, T2] interval.
26. ONVIF Client will invoke RTSP TEARDOWN control request at the end of metadata streaming to terminate the RTSP session.
27. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send a valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

The DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TEARDOWN requests.

RTSP Session was terminated by the DUT during metadata streaming.

The DUT did not accept an Immediate header for PLAY command.

The DUT did not send a valid RTP header for metadata stream.

The DUT did not cancel the first PLAY command after PLAY command with “Immediate: yes”.

The DUT did not immediately start playing from the new location after PLAY command with “Immediate: yes”.

The DUT sent the first frame after PLAY response with NTP time value not within requested interval at step 13 or step 16 or step 19 or step 22.

NOTE: The test tool will select MiddleT1, MiddleT1-delta and MiddleT1+delta values for PLAYS's range from the first minute of the recording.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.6 Playback Control – Video, Audio and Metadata Streaming

4.6.1 PLAYBACK VIDEO, AUDIO AND METADATA STREAMING –

MEDIA CONTROL

Test Label: DUT RTSP Control Messages Verification for Replay Video, Audio and Metadata.

Test Case ID: REPLAY-3-4-1

ONVIF Core Specification Coverage: RTP data transfer via UDP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify RTSP control messages of the DUT for replay Video, Audio and Metadata streaming.

Pre-Requisite: Recording with both video, audio (if Audio Recording is supported), and metadata track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) from the DUT.

4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP OPTIONS control request to get the RTSP methods supported by DUT.
9. The DUT sends 200 OK Response and list of supported RTSP methods.
10. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
11. The DUT sends 200 OK Response and SDP message (*sdp*).
12. Set *responseHeader* := header of response to DESCRIBE.
13. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
14. ONVIF Client validates the session description information in the SDP message.
15. ONVIF Client verifies tag with TrackReference in SDP message for each Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track.
16. ONVIF Client will invoke RTSP SETUP control request with "onvif-replay" feature tag (**"Require: onvif-replay"**) to create a RTSP Session.
17. The DUT sends 200 OK Response and Stream Information details.
18. ONVIF Client verifies "Transport", "Session" and "timeout" header fields in the SETUP response message.
19. ONVIF Client will invoke RTSP PLAY (**"Require: onvif-replay", "Range: clock = T1-"**) control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.
20. The DUT sends 200 OK Response and RTP protocol information.
21. ONVIF Client verifies "Session", "RTP-Info", "seq", "uri" and "rtptime" header fields in the PLAY response message.
22. DUT transfers Video, Audio (if Audio Recording supported), and Metadata (if Metadata Recording is supported) streams over UDP.
23. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream and render it after the validation.
24. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
25. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send mandatory RTSP commands DESCRIBE SETUP, PLAY, TEARDOWN, and OPTIONS in OPTIONS response.

The DUT sends SDP message with one or more Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track without the TrackReference in corresponding format (a:x-onvif-track:<TrackReference>).

DUT did not send correct Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream information in the SDP message.

DUT did not send RTSP 200 OK response for RTSP OPTIONS, DESCRIBE, SETUP and PLAY requests.

DUT did not send mandatory headers or fields in the SETUP response message.

DUT did not send mandatory headers or fields in the PLAY response message.

RTSP Session was terminated by the DUT during Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.

DUT did not send a valid RTP header including replay header extension for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream.

NOTE: RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then minimum value of Track.DataFrom for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T1.

NOTE: RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then maximum value of Track.DataTo for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T2.

NOTE: See A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.6.2 PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – RTP-

Unicast/UDP

Test Label: Replay Viewing DUT Video, Audio and Metadata streaming using RTP-Unicast/UDP transport.

Test Case ID: REPLAY-3-4-2

ONVIF Core Specification Coverage: RTP data transfer via UDP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify playback Video, Audio and Metadata streaming based on RTP/UDP Unicast Transport.

Pre-Requisite: Recording with both video, audio (if Audio Recording is supported), and metadata (if Metadata Recording is supported) track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type

12. ONVIF Client will invoke RTSP SETUP (“**Require: onvif-replay**”) request with transport parameter as **RTP/UDP**.
13. The DUT sends 200 OK message and the Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream information.
14. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-**”) control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.
15. The DUT sends 200 OK message and starts Video (if is supported), Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.
16. The DUT sends RTP Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream to ONVIF Client over UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream and render it after the validation.
18. ONVIF Client validates RTCP packets if The DUT sends theirs.
19. ONVIF Client will invoke RTSP TEARDOWN control request at the end of Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming to terminate the RTSP session.
20. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.

DUT did not send a valid RTP header and RTP header extension for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream.

DUT sent an incorrect RTCP report.

NOTE: RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then minimum value of Track.DataFrom for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T1.

NOTE: RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then maximum value of Track.DataTo for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T2.

NOTE: RTCP report will be validated only if it is received from the DUT.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.6.3 PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – RTP- Unicast/RTSP/HTTP/TCP

Test Label: Replay Viewing DUT Video, Audio and Metadata streaming using HTTP transport.

Test Case ID: REPLAY-3-4-3

ONVIF Core Specification Coverage: RTP/RTSP/HTTP/TCP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify playback Video, Audio and Metadata streaming based on HTTP Transport.

Pre-Requisite: Recording with both video, audio (if Audio Recording is supported), and metadata (if Metadata Recording is supported) track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message
(RecordingInformation.RecordingToken = Token1,
RecordingInformation.EarliestRecording = T1,
RecordingInformation.LatestRecording = T2).
6. ONVIF Client will invoke GetReplayUriRequest (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "HTTP"**).
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.

9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP (“**Require: onvif-replay**”) request with transport parameter as **RTP/TCP**.
13. The DUT sends 200 OK message and the Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream information.
14. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-***”) control request to initiate the Video, Audio and Metadata streaming.
15. The DUT sends 200 OK message and starts Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.
16. The DUT sends RTP Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream to ONVIF Client over HTTP.
17. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream and render it after the validation.
18. ONVIF Client validates RTCP packets if The DUT sends theirs.
19. ONVIF Client will invoke RTSP TEARDOWN control request at the end of Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming to terminate the RTSP session.
20. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send s valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send s valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.

DUT did not send s valid RTP header and RTP header extension for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream.

DUT sent an incorrect RTCP report.

NOTE: RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then minimum value of Track.DataFrom for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T1.

NOTE: RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then maximum value of Track.DataTo for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T2.

NOTE: RTCP report will be validated only if it is received from the DUT.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.6.4 PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – RTP/RTSP/TCP

Test Label: Replay Viewing DUT Video, Audio and Metadata streaming using RTP/RTSP/TCP transport.

Test Case ID: REPLAY-3-4-4

ONVIF Core Specification Coverage: RTP/RTSP/TCP, RTP, RTCP, Stream control, RTSP, Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify playback Video, Audio and Metadata streaming based on RTP/RTSP/TCP using RTSP tunnel.

Pre-Requisite: Recording with both video, audio (if Audio Recording is supported), and metadata (if Metadata Recording is supported) track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.

5. Verify the GetRecordingInformationResponse message (RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2).
6. ONVIF Client will invoke GetReplayUriRequest (RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "RTSP").
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE request.
9. The DUT sends 200 OK message and SDP information (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP ("Require: onvif-replay") request with transport parameter as RTP/TCP.
13. The DUT sends 200 OK message and the Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream information.
14. ONVIF Client will invoke RTSP PLAY ("Require: onvif-replay", "Range: clock = T1-") control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.
15. The DUT sends 200 OK message and starts Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.
16. The DUT sends RTP Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream to ONVIF Client over TCP.
17. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream and render it after the validation.
18. ONVIF Client validates RTCP packets if The DUT sends theirs.
19. ONVIF Client will invoke RTSP TEARDOWN control request at the end of Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming to terminate the RTSP session.
20. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.

DUT did not send a valid RTP header and RTP header extension for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream.

DUT sent an incorrect RTCP report.

NOTE: RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then minimum value of Track.DataFrom for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T1.

NOTE: RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then maximum value of Track.DataTo for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T2.

NOTE: RTCP report will be validated only if it is received from the DUT.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.6.5 PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – PLAY WITH RANGE

Test Label: Replay Video, Audio and Metadata Streaming from Specified Range.

Test Case ID: REPLAY-3-4-10

ONVIF Core Specification Coverage: Playback.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify stop of replay Video, Audio and Metadata streaming based on RTP/UDP Unicast Transport after the specified time Range was reached.

Pre-Requisite: Recording with both video, audio (if Audio Recording is supported), and metadata (if Metadata Recording is supported) track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.

3. Get token **Token1** of recording which contains at least one track with Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
9. The DUT sends 200 OK Response and SDP message (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP control request with "onvif-replay" feature tag (**"Require: onvif-replay"**) to create a RTSP Session.
13. The DUT sends 200 OK Response and Stream Information details.
14. ONVIF Client will invoke RTSP PLAY (**"Require: onvif-replay", "Range: clock = T1-T1 + delta"**) control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.
15. The DUT sends 200 OK Response and RTP protocol information.
16. DUT transfers Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streams over RTP/UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream and render it after the validation.
18. DUT stops transferring of Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream after the specified time **T1 + delta** was reached.
19. ONVIF Client verifies that there are no RTP received after the specified time **[T1 + delta]**.
20. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
21. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP OPTIONS, DESCRIBE, SETUP and PLAY requests.

DUT did not stop Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming after the specified time.

DUT did not send valid RTP header in Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream (RTP sequence number not in the orders of delivery, timestamps correspond to playback timing considering inaccuracy).

NOTE: RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then minimum value of Track.DataFrom for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T1.

NOTE: RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then maximum value of Track.DataTo for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T2.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.6.6 PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – RATECONTROL

Test Label: Compare Replay Video, Audio and Metadata Streaming with different Rate-Control.

Test Case ID: REPLAY-3-4-11

ONVIF Core Specification Coverage: Rate-Control header field.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify applying of Rate-Control value for playback Video, Audio and Metadata streaming.

Pre-Requisite: Recording with both video, audio (if Audio Recording is supported), and metadata (if Metadata Recording is supported) track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "HTTP"**).
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.
10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
11. ONVIF Client will invoke RTSP SETUP request ("**Require: onvif-replay**") with transport parameter as **RTP/AVP/TCP**.
12. The DUT sends 200 OK message and the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream information.
13. ONVIF Client will invoke RTSP PLAY ("**Require: onvif-replay**", "**Range: clock = T1-T1+Delta**", "**Rate-Control: yes**") control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.
14. The DUT sends 200 OK message and starts Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.
15. The DUT sends RTP Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream to ONVIF Client over HTTP.
16. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream and render it after the validation.
17. ONVIF Client waits while replay streaming is played and calculates play duration **D1**.
18. ONVIF Client will invoke RTSP PLAY ("**Require: onvif-replay**", "**Range: clock = T1-T1+Delta**", "**Rate-Control: no**") control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.
19. The DUT sends 200 OK message and starts Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.
20. The DUT sends RTP Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream to ONVIF Client over HTTP.

21. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream and render it after the validation.
22. ONVIF Client waits while replay streaming is played and calculates play duration **D2**.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY requests.

RTSP Session is terminated by the DUT during Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.

DUT did not accept Rate-Control header for PLAY command.

DUT did not send a valid RTP header for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream with rate control (RTP sequence number not in the order of delivery, timestamps correspond to playback timing considering inaccuracy).

DUT did not send valid RTP header for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream without rate control (RTP sequence number not in the order of delivery, timestamps correspond to original sampling times of recorded frames considering inaccuracy).

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.6.7 PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – IMMEDIATE HEADER

Test Label: Compare Replay Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming with Immediate header.

Test Case ID: REPLAY-3-4-12

ONVIF Core Specification Coverage: Immediate header field

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To compare RTSP playback with Immediate header.

Pre-Requisite: Recording with both video, audio (if Audio Recording is supported), and metadata (if Metadata Recording is supported) track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.
10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - o in *responseHeader* - header of response to DESCRIBE
 - o in *sdp* - SDP information
 - o in IPv4 – IP type
11. ONVIF Client will invoke RTSP SETUP request ("**Require: onvif-replay**") with transport parameter as **RTP/UDP**.
12. The DUT sends 200 OK message and the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream information.
13. ONVIF Client will invoke RTSP PLAY ("**Require: onvif-replay**", "**Range: clockT1 -**") control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.
14. The DUT sends 200 OK message and starts Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.
15. The DUT sends RTP Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream to ONVIF Client over UDP.

16. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream and render it after the validation.
17. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock [T1 + delta]**”, “**Immediate: yes**”) control request to initiate the second Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming from other time.
18. DUT cancelling existing first PLAY command and immediately start playing from the **T1+delta**.
19. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream and render it after the validation. ONVIF Client verifies that first packet sent from the new location have the D (discontinuity) bit set in its RTP extension header.
20. ONVIF Client will invoke RTSP TEARDOWN control request at the end of Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming to terminate the RTSP session.
21. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TERDOWN requests.

RTSP Session was terminated by the DUT during Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.

DUT did not accept an Immediate header for PLAY command.

DUT did not send s valid RTP header for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream.

DUT did not cancel the first PLAY command after PLAY command with “Immediate: yes”.

DUT did not immediately start playing from the new location after PLAY command with “Immediate: yes”.

DUT did not send valid RTP header from the new location (first packet did not have the D (discontinuity) bit set in its RTP extension header).

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.6.8 PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – PAUSE WITHOUT RANGE

Test Label: Replay Viewing DUT Video, Audio and Metadata streaming using with Pause (no Range).

Test Case ID: REPLAY-3-4-14

ONVIF Core Specification Coverage: Playback, Pause.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify pause without Range of replay Video, Audio and Metadata streaming based on RTP/UDP Unicast Transport.

Pre-Requirement: Recording with both video, audio (if Audio Recording is supported), and metadata (if Metadata Recording is supported) track with data exists for a DUT (see AnnexA.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT.
8. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
9. The DUT sends 200 OK Response and SDP message (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information

- in IPv4 – IP type
- 12. ONVIF Client will invoke RTSP SETUP control request with “onvif-replay” feature tag (“**Require: onvif-replay**”) to create a RTSP Session with transport parameter as „RTP/UDP”.
- 13. The DUT sends 200 OK Response and Stream Information details.
- 14. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-**”) control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.
- 15. The DUT sends 200 OK Response and RTP protocol information.
- 16. DUT transfers Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streams over RTP/UDP.
- 17. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) stream and renders it after the validation.
- 18. ONVIF Client will invoke RTSP PAUSE (**no Range header**) control request to pause the Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.
- 19. The DUT sends 200 OK Response.
- 20. ONVIF Client verifies “CSeq” header field and optional “Date” header field if it is present in the PAUSE response message.
- 21. DUT pauses transferring of Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streams and DUT sets the pause point to the current normal play time.
- 22. ONVIF Client verifies that there are no RTP received after this.
- 23. ONVIF Client will wait 2s after pause of replay streaming and will invoke RTSP PLAY (“**Require: onvif-replay**”, **no Range**) control request to resume the Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streaming.
- 24. The DUT sends 200 OK Response and RTP protocol information.
- 25. DUT resumes Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) streams over RTP/UDP from the pause point.
- 26. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
- 27. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PAUSE and PLAY requests.

DUT did not send mandatory header or field in the PAUSE response message ("CSeq").

DUT did not accept a Range header with a value of clock for PAUSE request.

DUT did not pause streaming after PAUSE request.

DUT did not resume streaming after PAUSE request from pause point.

NOTE: RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then minimum value of Track.DataFrom for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T1.

NOTE: RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then maximum value of Track.DataTo for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T2.

NOTE: On the step 20 (verification that no packets were sent out with a timestamp later than the Date-field): If Date-field is not supplied, the ONVIF Client will follow to the following logic:

If after receiving reply to the "PAUSE" command the ONVIF Client do not receive any RTP packet during 3 sec then stepis considered as passed. Otherwise, the ONVIF Client will check that DUT do not send RTP packets within 3 sec after the last packet, which had been received in the first 3 sec timespan.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.6.9 PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – PAUSE WITH RANGE

Test Label: Replay Viewing DUT Video, Audio and Metadata streaming using with Pause (with Range).

Test Case ID: REPLAY-3-4-15

ONVIF Core Specification Coverage: Playback, Pause.

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify pause with Range of replay Video, Audio and Metadata streaming based on RTP/UDP Unicast Transport.

Pre-Requisite: Recording with both video, audio (if Audio Recording is supported), and metadata (if Metadata Recording is supported) track with data exists for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. Get token **Token1** of recording which contains at least one track with Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. Verify the GetReplayUriResponse from the DUT
8. ONVIF Client will invoke RTSP DESCRIBE control request to retrieve the media description information.
9. The DUT sends 200 OK Response and SDP message (*sdp*).
10. Set *responseHeader* := header of response to DESCRIBE.
11. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
12. ONVIF Client will invoke RTSP SETUP control request with "onvif-replay" feature tag (**"Require: onvif-replay"**) to create a RTSP Session.
13. The DUT sends 200 OK Response and Stream Information details.
14. ONVIF Client will invoke RTSP PLAY (**"Require: onvif-replay", "Range: clock = T1-"**) control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.
15. The DUT sends 200 OK Response and RTP protocol information.
16. DUT transfers Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streams over RTP/UDP.
17. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream and render it after the validation.
18. ONVIF Client will invoke RTSP PAUSE (**"Range: clock = T1 + delta"**) control request to pause the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.
19. The DUT sends 200 OK Response.
20. ONVIF Client verifies "CSeq" header field and optional "Date" header field if it is present in the PAUSE response message.
21. DUT pauses transferring of Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streams after the specified time.

22. ONVIF Client verifies that there are no RTP received after specified time **T1 + delta**.
23. ONVIF Client will wait 2 s after pause of replay streaming and will invoke RTSP PLAY (**“Require: onvif-replay”, no Range**) control request to resume the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.
24. The DUT sends 200 OK Response and RTP protocol information.
25. DUT resumes Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streams over RTP/UDP from the pause point.
26. ONVIF Client will invoke RTSP TEARDOWN control request to terminate the RTSP session at the end of the streaming.
27. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

DUT passed all assertions.

FAIL –

DUT did not send a valid GetRecordingInformationResponse message.

DUT did not send GetReplayUriResponse message.

DUT did not send a valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PAUSE and PLAY requests.

DUT did not send mandatory header or field in the PAUSE response message (“CSeq”).

DUT did not accept a Range header with a value of clock for PAUSE request.

DUT did not pause streaming after PAUSE request at specified point.

DUT did not resume streaming after PAUSE request from posed point.

NOTE: RecordingInformation.EarliestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataFrom for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T1.

NOTE: RecordingInformation.LatestRecording is not included in GetRecordingInformationResponse message, then value of Track.DataTo for Video, Audio (if Audio Recording is supported), and Metadata (if Metadata Recording is supported) track will be used as T2.

NOTE: See Annex A.1 for Invalid RTP header definition.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.6.10 PLAYBACK VIDEO, AUDIO AND METADATA STREAMING – SEEK

Test Label: Verify Seek Scenario for DUT Video, Audio and Metadata Streaming.

Test Case ID: REPLAY-3-4-16

ONVIF Core Specification Coverage: Immediate header field

Command under test: None

WSDL Reference: replay.wsdl

Test Purpose: To verify Seek scenario for DUT Video, Audio and Metadata streaming.

Pre-Requisite: Recording with all three: video, audio (if Audio Recording is supported), and metadata (if Metadata Recording is supported) tracks with data exist for a DUT (see Annex A.2).

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start the DUT.
3. Get token **Token1** of recording which contains at least one track with Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) from the DUT.
4. ONVIF Client will invoke GetRecordingInformationRequest message (**RecordingToken = "Token1"**) to retrieve information about Recording.
5. Verify the GetRecordingInformationResponse message (**RecordingInformation.RecordingToken = Token1, RecordingInformation.EarliestRecording = T1, RecordingInformation.LatestRecording = T2**).
6. ONVIF Client will invoke GetReplayUriRequest message (**RecordingToken = Token1, Stream = "RTP-Unicast", Transport.Protocol = "UDP"**).
7. ONVIF Client will invoke RTSP DESCRIBE request.
8. The DUT sends 200 OK message and SDP information (*sdp*).
9. Set *responseHeader* := header of response to DESCRIBE.
10. ONVIF Client checks types of IP addresses returned in response to DESCRIBE by following the procedure mentioned in Annex A.5 with the following input parameters
 - in *responseHeader* - header of response to DESCRIBE
 - in *sdp* - SDP information
 - in IPv4 – IP type
11. ONVIF Client will invoke RTSP SETUP request ("**Require: onvif-replay**") with transport parameter as **RTP/UDP**.
12. The DUT sends 200 OK message and the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream information.

13. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = T1-**”, “**Rate-Control: no**”) control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.
14. The DUT sends 200 OK message and starts Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.
15. The DUT sends RTP Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream to ONVIF Client over RTSP control connection.
16. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream and render it after the validation. ONVIF Client compares NTP time of the first frames with start time of the recording considering inaccuracy (inaccuracy value=2 s).
17. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = MiddleT1-**”, “**Rate-Control: no**”, “**Immediate: yes**”, where MiddleT1 is the time point into the recording time range) control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming from other time.
18. The DUT cancels previous PLAY command and immediately start playing from the **MiddleT1**.
19. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream and render it after the validation. ONVIF Client compares NTP time of the first frame with MiddleT1 considering inaccuracy (inaccuracy value=2 s).
20. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = [MiddleT1 – delta]-**”, “**Rate-Control: no**”, “**Immediate: yes**”) control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming from other time.
21. The DUT cancels previous PLAY command and immediately start playing from the **MiddleT1-delta**.
22. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream and render it after the validation. ONVIF Client compares NTP time of the first frame with MiddleT1-delta considering inaccuracy (inaccuracy value=2 s).
23. ONVIF Client will invoke RTSP PLAY (“**Require: onvif-replay**”, “**Range: clock = [MiddleT1 + delta]-**”, “**Rate-Control: no**”, “**Immediate: yes**”) control request to initiate the Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming from other time.
24. The DUT cancels previous PLAY command and immediately start playing from the **MiddleT1+delta**.
25. ONVIF Client verifies RTP header and RTP header extensions for Video, Audio (if Audio Recording is unsupported), and Metadata (if Audio Recording is supported) stream and render it after the validation. ONVIF Client compares NTP time of the first frame with MiddleT1+delta considering inaccuracy (inaccuracy value=2 s).

26. ONVIF Client will invoke RTSP TEARDOWN control request at the end of Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming to terminate the RTSP session.
27. The DUT sends 200 OK Response and terminates the RTSP Session.

Test Result:**PASS –**

The DUT passed all assertions.

FAIL –

The DUT did not send valid GetRecordingInformationResponse message.

The DUT did not send GetReplayUriResponse message.

The DUT did not send valid GetReplayUriResponse message with valid RTSP URI.

DUT did not send RTSP 200 OK response for RTSP DESCRIBE, SETUP, PLAY and TEARDOWN requests.

RTSP Session was terminated by the DUT during Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) streaming.

The DUT did not accept an immediate header for PLAY command.

The DUT did not send valid RTP header for Video, Audio (if Audio Recording is supported), and Metadata (if Audio Recording is supported) stream.

The DUT did not cancel the first PLAY command after PLAY command with "Immediate: yes".

The DUT sent the first frame after PLAY response with NTP time different from the requested range at step 13 or step 16 or step 19 or step 22.

NOTE: The test tool will select MiddleT1, MiddleT1-delta and MiddleT1+delta values for PLAYS's range from the first minute of the recording.

NOTE: ONVIF Client checks authentication type for all RTSP requests by following the procedure mentioned in A.4.

4.7 Replay Configuration

4.7.1 REPLAY CONFIGURATION

Test Case ID: REPLAY-4-1-1

Specification Coverage: ReplayConfiguration

Feature under test: GetReplayConfiguration, SetReplayConfiguration

WSDL Reference: replay.wsdl

Test Purpose: To check GetReplayConfiguration and SetReplayConfiguration commands.

Pre-Requisite: Replay Service is received from the DUT. GetServices is supported by the DUT.

Test Configuration: ONVIF Client and DUT.

Test Procedure:

1. Start an ONVIF Client.
2. Start a DUT.
3. ONVIF Client gets the service capabilities (out *cap*) by following the procedure mentioned in Annex A.3
4. ONVIF Client invokes **GetReplayConfiguration** request message.
5. The DUT responds with **GetReplayConfigurationResponse** with parameters

Configuration =: *initialConf*

6. If *cap.SessionTimeoutRange* skipped set *sessionTimeout* := PT60S.
7. If *cap.SessionTimeoutRange* is not skipped and *initialConf.SessionTimeout* is not equal to the first value from *cap.SessionTimeoutRange*, set *sessionTimeout* to the first value from *cap.SessionTimeoutRange*. Otherwise set *sessionTimeout* to the last value from *cap.SessionTimeoutRange* list.
8. ONVIF Client invokes **SetReplayConfiguration** request message with parameters

Configuration.SessionTimeout := *sessionTimeout* value

9. ONVIF Client invokes **GetReplayConfiguration** request message.
10. The DUT responds with **GetReplayConfigurationResponse** with parameters

Configuration =: *conf*

11. If *conf.SessionTimeout* value is not equal to *sessionTimeout* value, FAIL the test and go to step 12.
12. ONVIF Client restores replay configuration changed at step 8.

Test Result:

PASS –

The DUT passed all assertions.

FAIL –

The DUT did not send GetReplayConfigurationResponse message.

The DUT did not send SetReplayConfigurationResponse message.

Annex A

This section describes the meaning of the following definitions. These definitions are used in the test case description.

A.1 Invalid RTP header extension

An RTP header extension which is not formed according to the header field format defined in the Streaming Specification document Section 6.2 is considered as invalid RTP header.

RTP header will be assumed as invalid in the following cases:

1. Sequence number is not in order of delivery.

RTP header with replay header extension will be assumed as invalid in the following cases:

1. RTP header extension ID is not equal to 0xABAC.
2. RTP header does not have one of the following fields: NTP timestamp, C, E, D, mbz, CSeq (e.g. RTP replay extension does not have proper length).
3. NTP timestamp is not monotonically increasing in case of normal playback or decreasing in case of reverse playback.

A.2 Recording Environment Pre-Requisite

The following pre-perquisite shall be filled before executing tests for Replay Service:

1. At least one recording shall be present at the DUT and specified for test execution
2. The recording shall be stopped before executing tests
3. The recording shall contain at least one track each for video, audio (if Audio Recording is supported) and Metadata (if Metadata Recording is supported)
4. The recording shall contain data at least in the video track, audio track (if Audio Recording is supported), and metadata track (if Metadata Recording is supported) and configuration of each track shall be consistent for the duration of the recording.
5. The recording contains at least one gap
6. The recording span (including gap) should be not more than 3 minutes
7. Each recording event shall be at least one minute for the recording
8. GOPs of recording should be renewed every 2 seconds.

A.3 Get Service Capabilities

Name: HelperGetServiceCapabilities

Procedure Purpose: Helper procedure to get service capabilities.

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

Input: None

Returns: The service capabilities (*cap*).

Procedure:

1. ONVIF Client invokes **GetServiceCapabilities**.
2. The DUT responds with a **GetServiceCapabilitiesResponse** message with parameters

Capabilities =: *cap*

Procedure Result:

PASS –

The DUT passed all assertions.

FAIL –

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

A.4 RTSP Authentication Check

Name: HelperRTSPAuthenticationCheck

Procedure Purpose: Helper procedure to check that DUT uses Digest authentication type for RTSP.

Pre-requisite: Real-time Streaming is supported by DUT.

Input: None.

Returns: None.

Procedure:

1. If DUT sends RTSP 401 Unauthorized message to any RTSP request:
 - 1.1. If RTSP 401 Unauthorized message does not have **WWW-Authenticate: Digest** header, FAIL the test and skip other steps.

Procedure Result:

PASS –

DUT passes all assertions.

FAIL –

DUT did not send response message to RTSP request.

A.5 Check of IP address type in response to RTSP DESCRIBE

Name: HelperCheckIPType

Procedure Purpose: Helper procedure to check IP addresses types returned by DUT in response to RTSP DESCRIBE.

Pre-requisite: None.

Input: Header of response to DESCRIBE (*responseHeader*). SDP (*sdp*). IP type (*ipType*).

Returns: None.

Procedure:

1. For each **Content-Base** field in *responseHeader* (*contentBase*) that has absolute IP value:
 - 1.1. If *contentBase* IP value does not correspond to *ipType*, FAIL the test and skip other steps (see [RFC2326] for details).
2. For each **Content-Location** field in *responseHeader* (*contentLocation*) that has absolute IP value:
 - 2.1. If *contentLocation* IP value does not correspond to *ipType*, FAIL the test and skip other steps (see [RFC2326] for details).
3. For each **"a=control"** attribute in *sdp* (*aControl*) that has absolute IP value:
 - 3.1. If *aControl* IP value does not correspond to *ipType*, FAIL the test and skip other steps (see [RFC2326] for details).
4. If *ipType* = IPv4:
 - 4.1. If *sdp* contains at least one origin field ("o=") with **addrtype** != "IP4", FAIL the test and skip other steps (see [RFC4566] for details).
 - 4.2. If *sdp* contains at least one origin field ("o=") with IP type of **unicast-address** sub-field != IPv4 type, FAIL the test and skip other steps (see [RFC4566] for details).
 - 4.3. If *sdp* contains at least one connection data field ("c=") with **addrtype** != "IP4", FAIL the test and skip other steps (see [RFC4566] for details).
 - 4.4. If *sdp* contains at least one connection data field ("c=") with IP type of **connection address** sub-field != IPv4 type, FAIL the test and skip other steps (see [RFC4566] for details).
5. If *ipType* = IPv6:
 - 5.1. If *sdp* contains at least one origin field ("o=") with **addrtype** != "IP6", FAIL the test and skip other steps (see [RFC4566] for details).
 - 5.2. If *sdp* contains at least one origin field ("o=") with IP type of **unicast-address** sub-field != IPv6 type, FAIL the test and skip other steps (see [RFC4566] for details).
 - 5.3. If *sdp* contains at least one connection data field ("c=") with **addrtype** != "IP6", FAIL the test and skip other steps (see [RFC4566] for details).
 - 5.4. If *sdp* contains at least one connection data field ("c=") with IP type of **connection address** sub-field != IPv6 type, FAIL the test and skip other steps (see [RFC4566] for details).

Procedure Result:

PASS –

DUT passes all assertions.

FAIL –

None.