ONVIF®

Device Feature Discovery Specification

Version 20.06

June 2020

REVISION HISTORY

Vers.	Date	Description
11.12	Dec 22, 2011	First issue
12.06	Jun 22, 2012	Update for a new version of the ONVIF Device Test Tool. No changes made for Feature Discovery.
12.12	Dec 20, 2012	Update for a new version of the ONVIF Device Test Tool. Recording Control Service, Recording Search Service, Replay Control Service, Receiver Service specific features were added.
13.06	Jun, 2013	Update for a new version of the ONVIF Device Test Tool. Access Control Service, Door Control Service specific features were added.
13.12	Dec, 2013	Advanced Security features were added. tns1:RecordingConfig/DeleteTrackData feature was added into section Recording Control Service support and section Recording Control Service support items.
14.06	Jun, 2014	'Metadata Recording support' item was removed. Recording Control features support (GetServices) was added in section Recording Control Service support Item
14.12	Dec, 2014	IrCutfilterConfiguration function support was added into section Imaging Service support Item and into section Imaging Service support Item. Scope was updated. Passphrase Management feature support was added into section Advanced Security Service Support Item. Feature definition for Profile Q was added: section Monitoring Events support Item was added, section Monitoring Events support Item was added, section Device Management Service Capabilities and section Device Service Capabilities configuration functionality in Device Management Service items were updated.
15.06	Jun, 2015	CRLs Management, Certification path validation policies Management, TLS WWW client auth extended key usage extension, TLS client authentication features support was added into section Advanced Security Service support. Credential Service features support was added into section Credential Service support and section Credential Service support. Access Rules Service features support was added into section Access Rules Service support and section Access Rules Service support. Schedule Service features support was added into section Schedule Service support and section Schedule Service support.
16.06	Jan 27, 2016	The section Media2 Service – general has been added.
16.06	Mar 15, 2016	The conditions in tables have been updated.

16.07	July 7, 2016	Changed version number, added Media2 Profile Configuration for PTZ Control
16.09	Sep, 2016	TLS1.0, TLS1.1, TLS1.2 features added
16.10	Oct, 2016	Media2 Service features support item was added
17.01	Jan, 2017	Minor changes: typos were fixed
17.01	Jan, 2017	TLS1.0, TLS1.1, TLS1.2 features removed
17.06	Feb, 2017	The following items were updated: Section Media2 Service – general Section PTZ Service support Section PTZ Service support The following item was added: Section Media2 Service support Scope list updated.
17.06	Mar 06, 2017	Tampering Events support in Imaging Service Item was added into section Imaging Service support and into section Imaging Service support. Scope list updated.
17.06	Mar 07, 2017	Analytics feature added into section Media2 Service – general
17.06	Apr 18, 2017	Table Media2 Service features support was updated. Section Analytics Service support was updated. Rule Engine and Rule Options features for Analytics Service were added into 5.5.10 Analytics Service support section and into section Analytics Service support section.
17.06	Apr 27, 2017	Table Credential features support (GetServices) updated with adding pt:ExemptFromAuthentication feature according to #1385 ticket.
17.06	May 03, 2017	TLS1.0, TLS1.1, TLS1.2 features added
17.06	May 22, 2017	Motion Alarm support in Imaging Service Item was added into section Imaging Service support and into section Imaging Service support. Scope list updated.
17.06	Jun 2, 2017	Table Media2 Service features support was updated according to #1382: acceptance of 'All' value in ConfigurationsSupported removed.
17.12	Jul 12, 2017	Serial Ports support in Device IO Service Item was added into section Device IO Service. Scope list updated.
17.12	Jul 13, 2017	Digital Input Options support in Device IO Service Item was added into section Device IO Service. Scope list updated.
17.12	Jul 20, 2017	Section Provisioning Service support was added.

		Section Provisioning Service support was added. Scope list updated.
17.12	Jul 24, 2017	JPEG, H.264, and MPEG4 features were added into Recording Control features support (GetServices) and Recording Control features support (GetCapabilities) according to #1377. Scope list updated.
17.12	Aug 29, 2017	The document formating were updated.
17.12	Aug 29, 2017	Section Thermal Service support was added. Section Thermal Service support was added. Scope list updated.
17.12	Oct 03, 2017	Section Auxiliary Commands features support was added. Section Auxiliary Commands support was added. Scope list updated. Section Device Management Service Capabilities was updated.
17.12	Oct 12, 2017	RTP/RTSP/HTTPS feature support was added in Media2 Service – general. Scope list updated.
17.12	Oct 16, 2017	Motion Region Detector Rule feature support was added in section Analytics Service support and in 5.6.12 Analytics Service support according to #1185. Scope list updated.
17.12	Oct 26, 2017	Focus Control function support was added in section Imaging Service support and in section Imaging Service support according to #1450. OSD Types function support was added in section Media2 Service – general according to #1450. Scope list updated.
17.12	Nov 22, 2017	Auxiliary Commands Features Support section was removed. The features discovery from this section was moved to section Device Management Service Capabilities.
18.06	Jan 23, 2018	The following were updated in the scope of #1567: Scope (new feature added) Discovery Procedure (GetServices)\Device IO Features Support (updated with Note) Discovery Procedure (GetServices)\Device IO Relay Output Options Support (added) Discovery Procedure (GetCapabilities)\Device IO Service (updated with note) Discovery Procedure (GetCapabilities)\Relay Outputs Support (updated with note)

		Discovery Procedure (GetCapabilities)\Device IO Relay Output Options Support (added)
18.06	Feb 13, 2018	The following were updated in the scope of #1535: Scope (the description of which features will be defined per each node was added) Feature Support Criteria (updated with entity for features) PTZ Service Features Support (GetService) (reordered) PTZ Service Support (GetServices) (links updated) Various Functions Support in PTZ Service (GetServices) (removed) PTZ Nodes Features Support (GetServices) (added) Fixed/Configurable Home Position Support (GetServices) (removed) Fixed/Configurable Home Position Support for PTZ Node (GetServices) (added) PTZ Service Support (GetServices) (links updated) PTZ Service Features Support (GetCapabilities) (reordered) Various Functions Support in PTZ Service (GetCapabilities) (removed) PTZ Nodes Features Support (GetCapabilities) (added) Fixed/Configurable Home Position Support (GetCapabilities) (removed) Fixed/Configurable Home Position Support for PTZ Node (GetCapabilities) (added) Annex A.2 Media2 Service - Media Profile Configuration for PTZ Control (removed) Annex A.8 Media Profile Configuration for PTZ Control (added) Annex A.9 Media Profile Configuration with Video Source Configuration (added)
18.06	Feb 13, 2018	The following were updated in the scope of #1581: Media2 Service – OSD Types Support (GetServices) table updated.
18.06	Apr 17, 2018	Section 'Event Service - Message Content Filter support' was added into 'Discovery Procedure (GetServices)' section according to #1340. Section 'Event Service - Message Content Filter support' was added into 'Discovery Procedure (GetCapabilities)' section section according to #1340. Scope list updated.
18.06	May 24, 2018	Section 'Event Service - ONVIF Message Content Filter Dialect Support' was added into 'Discovery Procedure (GetServices)' section according to #1618.

		Section 'Event Service - ONVIF Message Content Filter Dialect Support' was added into 'Discovery Procedure (GetCapabilities)' section section according to #1618. Scope list updated.
18.06	Jun 21, 2018	Reformatting document using new template
18.12	Aug 16, 2018	The following was done according to #1690: HTTPS Support was added into Discovery Procedure (GetServices) section. HTTPS Support was added into Discovery Procedure (GetCapabilities) section. RTP/RTSP/HTTPS feature support was updated. Scope list updated.
18.06 SR1	Aug 27, 2018	The following were updated in the scope of #1711: Scope (Device IO Service\Relay Outputs\Bistable and Device IO Service\Relay Outputs\Monostable was added) Discovery Procedure (GetServices): Device IO Service (updated with format and links to new section) Discovery Procedure (GetServices): Device IO Features Support (updated with format and links to new section) Discovery Procedure (GetServices): Device IO Relay Output Options Support (updated with format and links to new section) Discovery Procedure (GetServices): Device IO Relay Outputs Features Support (added) Discovery Procedure (GetCapabilities): Device IO Service (updated with format and links to new section) Discovery Procedure (GetCapabilities): Device IO Relay Output Support (updated with format and links to new section) Discovery Procedure (GetCapabilities): Device IO Relay Output Options Support (updated with format and links to new section) Discovery Procedure (GetCapabilities): Device IO Relay Outputs Features Support (added)
18.12	Aug 31, 2018	The following were updated in the scope of #1716: Discovery Procedure (GetServices): Imaging Service Support (format updated) Discovery Procedure (GetServices): IrCutfilterConfiguration Feature Support (replaced with new) Discovery Procedure (GetCapabilities): Imaging Service Support (format updated) Discovery Procedure (GetCapabilities): IrCutfilterConfiguration Feature Support (replaced with new)
18.12	Aug 31, 2018	The following were updated in the scope of #1726:

		Discovery Procedure (GetServices): Recording Search Service Support (updated to correspond to implementation) Discovery Procedure (GetCapabilities): Recording Search Service Support (updated to correspond to implementation)
18.12	Oct 1, 2018	The following were updated in the scope of #1599: Scope (Advanced Security Service\Keystore features support\No Private Key Sharing was added) Discovery Procedure (GetServices): Advanced Security Service Support (updated with format) Discovery Procedure (GetServices): Advanced Security Features Support (updated with format, definition of new feature No Private Key Sharing was added)
18.12	Oct 2, 2018	The following were updated in the scope of #1663: Annex A.1 Selection/Creation of Media Profile That Contains PTZ Configuration (reformatted, step 5 added) Annex A.11 Configure Empty Media Profile (added)
18.12	Oct 8, 2018	The following were updated in the scope of #1677: Scope (AccessControl/Denied/CredentialNotFound feature added) Discovery Procedure (GetServices): Access Control Service Support (description format updated) Discovery Procedure (GetServices): Access Control Service Support \Access Control Events Support (tns1:AccessControl/Denied/ CredentialNotFound added) Annex A.3 Get Complete Access Point Info List (description format updated) Annex A.4 Get Complete Area Info List (description format updated)
18.12	Oct 8, 2018	The following were updated in the scope of #1639 (for pre-configuration procedure): Introduction (updated with pre-configuration description) Scope (updated with pre-configuration description) Pre-Configuration Procedure (added)
18.12	Oct 8, 2018	The following were updated in the scope of #1670: Feature Discovery Procedure (Client Supplied Token feature was added) Discovery Procedure (GetServices): Schedule Service Support (description format updated) Discovery Procedure (GetServices): Schedule Service Support \Schedule Features Support (Client Supplied Token feature was added)
18.12	Oct 12, 2018	The following were updated in the scope of #1746:

		Discovery Procedure (GetServices): Device IO Relay Outputs Features Support (DelayTime value in SetRelayOutputSettings request for Monostable mode updated, Note added) Discovery Procedure (GetCapabilities): Device IO Relay Outputs Features Support (DelayTime value in SetRelayOutputSettings request for Monostable mode updated, Note added)
18.12	Oct 12, 2018	The following were updated in the scope of #1672: Feature Discovery Procedure (Client Supplied Token feature was added) Discovery Procedure (GetServices): Credential Service Support (description format updated) Discovery Procedure (GetServices): Credential Service Support \Credential Features Support (Client Supplied Token feature was added)
18.12	Oct 16, 2018	The following were updated in the scope of #1671: Feature Discovery Procedure (Client Supplied Token feature was added) Discovery Procedure (GetServices): Access Rules Service Support (description format updated) Discovery Procedure (GetServices): Access Rules Service Support \Access Rules Features Support (Client Supplied Token feature was added)
18.12	Nov 14, 2018	The following were updated in the scope of #1653: Scope (Advanced Security Service was renamed to Security Configuration Service) Normative references ("[ONVIF Advanced Security Test] ONVIF Advanced Security Test Specification:" replaced with "[ONVIF Security Configuration Test] ONVIF Security Configuration Device Test Specification:") Discovery Procedure (GetServices): HTTPS Support ("Advanced Security Service" was replaced with "Security Configuration Service" in several places) Discovery Procedure (GetServices): Advanced Security Service Support section was renamed to Security Configuration Service Support Discovery Procedure (GetServices): Security Configuration Service Support ("Advanced Security Service" was replaced with "Security Configuration Service" in many places) Discovery Procedure (GetCapabilities): Security Configuration Service Support section was renamed to Security Configuration Service Support Discovery Procedure (GetCapabilities): Security Configuration Service Support ("Advanced Security Service" was replaced with "Security Configuration Service" in many places)
19.06	Apr 10, 2019	The following were updated in the scope of #1764: Discovery Procedure (GetServices): Media2 Service – Audio Encoding Support ("MPEG4-GENERIC" value added for AAC)

		Discovery Procedure (GetServices): Media2 Service – Audio Decoding Support ("MPEG4-GENERIC" value added for AAC)
19.06	May 06, 2019	The following were updated in the scope of #1799: Scope (updated with Enabled TLS Versions feature) Discovery Procedure (GetServices): Security Configuration Features Support (updated with Enabled TLS Versions feature)
19.12	Apr 25, 2019	The following were updated in the scope of #1833: Feature Discovery Procedure (Analytics Modules and Analytics Module Options features were added) Discovery Procedure (GetServices): Analytics Features Support (Analytics Modules and Analytics Module Options features were added) Discovery Procedure (GetCapabilities): Analytics Service Support (Analytics Modules and Analytics Module Options features were added)
19.12	Jul 31, 2019	The following were updated in the scope of #1675: Scope (Door Management feature was added) Discovery Procedure (GetServices): Door Control Service (Door Control Features Support new section was added)
19.12	Aug 05, 2019	The following were updated in the scope of #1668: Scope (Client Supplied Token feature was added) Discovery Procedure (GetServices): Door Control Service (Client Supplied Token feature was added)
19.12	Sep 11, 2019	The following were updated in the scope of #1669 and #1676: Scope (Client Supplied Token (Area, Access Point) feature was added) Scope (Access Point Management feature was added) Scope (Area Management feature was added) Discovery Procedure (GetServices): 5.5.17.4 Access Control Service Features Support section was added into 5.5.17 Access Control Service Support
19.12	Sep 25, 2019	The following were updated in the scope of #1675: Door Control Features Support section (Door Entity was added section) Door Entity support was renamed to Door Entity Features Support Door Entity Features Support section (Door Entity was removed from section, pre-requisite was added) Door Control Events Support section (pre-requisite was added)
19.12	Sep 25, 2019	The following were updated in the scope of #1894: Scope (Metadata feature was added into Media2 service)

		Discovery Procedure (GetServices): Media2 Service – General updated (Metadata feature added into Table 5.18 Media2 Service Features Support (GetServices))
19.12	Oct 17, 2019	The following were updated in the scope of #1831: Scope (Supported Metadata feature was added under Analytics Modules) Discovery Procedure (GetServices): 5.5.11.1 Analytics Features Support (Supported Metadata feature added into Table 5.42 Analytics Features Support (GetServices)) Discovery Procedure (GetCapabilities): 5.6.13 Analytics Service Support (Supported Metadata feature added into Table 5.101 Analytics Service (GetCapabilities))
19.12	Oct 21, 2019	The following were updated in the scope of #1837: Scope (Object Classification feature was added under Analytics Modules/Metadata Types) Discovery Procedure (GetServices): 5.5.11.3 Analytics Service - Metadata Types section with Object Classification feature added Discovery Procedure (GetCapabilities): 5.6.13 Analytics Service - Metadata Types (added)
20.06	Jan 10, 2019	The following were updated in the scope of #1880 and #1882: Scope (Whitelist and Blacklist features were added) Discovery Procedure (GetServices): Credential Features Support (Whitelist and Blacklist features were added)
20.06	Jan 13, 2020	The following were updated in the scope of #1923: Scope (AccessControl/AccessGranted/Identifier and AccessControl/Denied/Identifier features were added) Discovery Procedure (GetServices): Access Control Events Support (AccessControl/AccessGranted/Identifier and AccessControl/Denied/Identifier features were added)
20.06	Jan 15, 2020	The following were updated in the scope of #1841: Scope (Analytics Service/Analytics Modules/Supported Metadata/Object Classification/Vehicle Info feature was added) Discovery Procedure (GetServices): Analytics Service - Supported Metadata (Vehicle Info feature was added)
20.06	Jan 15, 2020	The following were updated in the scope of #1878: Scope (Access Control Service\Access Point Entity\Identifier Access feature was added) Discovery Procedure (GetServices): Access Point Entity Support and Access Point Features Support (Identifier Access feature was added)
20.06	Jan 16, 2020	The following were updated in the scope of #1845: Scope (Analytics Service/Analytics Modules/Supported Metadata/Geo Location feature was added)

		Discovery Procedure (GetServices): Analytics Service - Supported Metadata (Geo Location feature was added)
20.06	Feb 10, 2020	The following were updated in the scope of #1820: Scope (Analytics Service/Analytics Modules/Supported Metadata/ Human Face feature was added) Discovery Procedure (GetServices): Analytics Service - Supported Metadata (Human Face feature was added) The following were updated in the scope of #1843: Scope (Analytics Service/Analytics Modules/Supported Metadata/ License Plate feature was added) Discovery Procedure (GetServices): Analytics Service - Supported Metadata (License Plate feature was added)
20.06	Feb 12, 2020	The following were updated in the scope of #1995: Scope (Event Service /Event Broker feature with subfeatures was added) 5.5.7.1 Event service features (Event Broker, mqtt, mqtts, ws, and wss features were added) 5.6.9.1 Event service features (Event Broker, mqtt, mqtts, ws, and wss features were added)
20.06	Mar 11, 2020	The following were updated in the scope of #1901: Scope (Media2 /Media2 Events was added) 5.5.6.6 Media2 Events Support section (added)
20.06	Mar 17, 2020	The following were updated in the scope of #1835: Discovery Procedure (GetServices): Analytics Features Support (Image Sending, Embedded Image Sending Type, Local Storage Image Sending Type, Remote Storage Image Sending Type features were added) Discovery Procedure (GetCapabilities): Analytics Service Support (Image Sending, Embedded Image Sending Type, Local Storage Image Sending Type, Remote Storage Image Sending Type features were added) Scope (Image Sending, Embedded Image Sending Type, Local Storage Image Sending Type, Remote Storage Image Sending Type features were added)
20.06	Mar 17, 2020	The following were updated in the scope of #1847 and #1849: Discovery Procedure (GetServices): Motion Region Detector Rule section was rename to Supported Rules section. Discovery Procedure (GetCapabilities): Motion Region Detector Rule section was rename to Supported Rules section. Discovery Procedure (GetServices): Supported Rules (Face Recognition Rule, License Plate Recognition Rule features were added). Scope (Face Recognition Rule, License Plate Recognition Rule features were added)

20.06	Apr 16, 2020	The following were updated in the scope of #2037: Discovery Procedure (GetServices): Analytics Service - Supported Metadata (Human Face feature was updated) Discovery Procedure (GetServices): Analytics Service - Supported Metadata (Human feature was replaced wit Human Body feature) Discovery Procedure (GetServices): Analytics Service - Supported Metadata (License Plate feature was replaced with License Plate Info feature) Discovery Procedure (GetServices): Analytics Service - Supported Metadata (Vehicle feature was replaced with Vehicle Info feature)
20.06	May 12, 2020	The following were updated in the scope of #1999: Feature Discovery Procedure (Pull-Point Notification added) Discovery Procedure (GetServices): Event Service section updated (reformatted, Event Service definition changed, Event Service - Pull-Point Notification Support section added) Discovery Procedure (GetCapabilities): Event Service section updated (reformatted, Event Service definition changed, Event Service - Pull-Point Notification Support section added)

Table of Contents

1 Introduction

ONVIF Test Specification ([ONVIF Test]) defines/describes test cases need to verify according to [ONVIF Network Interface Specs], [ONVIF Conformance] in conjunction with a certain Profile Specification requirements. However, requirement on which test cases need to be executed and passed is out of the scope of [ONVIF Test]. Such requirements have to be described in a separate document.

This document focuses on detailing out feature discovery procedure and pre-configuration procedure for ONVIF Device Test Tool (hereafter, it is referred to as ONVIF Client) to identify which functionality is supported / not supported by DUT and to prepare DUT for further testing.

Based on these results of the feature discovery, which test cases shall be executed and passed will be determined for the purpose of claiming conformance to Profile specification. Some feature detection will be based on capability query, and some other feature detection will be based on error code response retrieval toward a specific request.

Pre-configuration procedure provide possibility automatically pre-configure the DUT before test execution to prevent additional time expenses for Test Operator or conformance time.

1.1 Scope

The scope of this document is to define:

- Feature discovery procedure of the functionality listed in Section 1.1.1.
- Pre-configuration procedure for the items listed in Section 1.1.2.

1.1.1 Feature Discovery Procedure

Feature discovery procedure includes logic description for the following features:

- Security
  - WS-UsernameToken
  - HTTP digest authentication
- Discovery
  - Bye Message support
  - Types
    - dn:NetworkVideoTransmitter

- tds:Device
- Device Service
  - Capabilities
    - GetCapabilities
    - GetServices
  - Network
    - Zero Configuration
    - NTP support
    - IPv6
    - DHCPv6
    - Dynamic DNS
    - IP Filter
    - HTTPS
  - System
    - System logging
    - Http System Logging
    - Http Firmware Upgrade
    - Http Support Information
    - Http System Backup
  - Security
    - Maximum Users
    - Default Access Policy
    - Remote User Handling
    - Maximum Username Length

- Maximum Password Length
- TLS1.0
- TLS1.1
- TLS1.2

- I/O functionality
  - Relay Outputs
    - Bistable
      - Open
      - Closed
    - Monostable
      - Open
      - Closed

- Monitoring Events
  - Monitoring/ProcessorUsage
  - Monitoring/OperatingTime/LastReset
  - Monitoring/OperatingTime/LastReboot
  - Monitoring/OperatingTime/LastClockSynchronization
  - Monitoring/Backup/Last
  - Monitoring/Mechanical/FanFailed
  - Monitoring/Mechanical/PowerSupplyFailed
  - Monitoring/Mechanical/StorageFailed
  - Monitoring/EnvironmentalConditions/CriticalTemperature
- Device scopes retrieval via GetDeviceScopes
- Event Service

- Persistent notification storage support
- WS Basic Notification
- Message Content Filter
  - ONVIF Message Content Filter Dialect
- GetServiceCapabilities
  - MaxPullPoints capability
- Pull-Point Notification
- Event Broker
  - Protocols
    - mqtt
    - mqtts
    - ws
    - wss
- Media Service
  - Video
    - JPEG
    - H.264
    - MPEG4
  - Audio
    - G.711
    - G.726
    - AAC
  - Audio outputs
    - G.711

- G.726
- AAC

- Real-time streaming
  - RTP/UDP
  - RTP/RTSP/HTTP
  - RTP/RTSP/TCP
  - RTP-Multicast/UDP
- GetSnapshotUri support

- Media2 Service
  - Video
    - H.265
    - H.264
  - Audio
    - G.711
    - AAC
  - Audio outputs
    - G.711
    - AAC
  - Real-time streaming
    - RTP/UDP
    - RTP/RTSP/HTTP
    - RTP/RTSP/HTTPS
    - RTP/RTSP/TCP
    - RTP-Multicast/UDP

- RTSP WebSocket
- Snapshot Uri
- Video Source Mode
- OSD
  - Types
    - Text
    - Image
- Analytics
- Metadata
- Media2 Events
  - Media/ProfileChanged
  - Media/ConfigurationChanged

- PTZ Service
  - Get Compatible Configurations
  - Move Status
  - Status Position
  - For each PTZ node:
    - Absolut Move
      - Pan/Tilt Movement
      - Zoom Movement
    - Relative Move
      - Pan/Tilt Movement
      - Zoom Movement
    - Continuous Move

- Pan/Tilt Movement
- Zoom Movement
- Preset
- Home Position
  - Configurable
  - Fixed
- Auxiliary operations
- Speed
  - Speed for Pan/Tilt
  - Speed for Zoom

- Device IO Service
  - Relay Outputs
    - Relay Output Options
    - For each Relay Output node:
      - Bistable Mode
        - Open Idle Sate
        - Closed Idle Sate
      - Monostable Mode
        - Open Idle Sate
        - Closed Idle Sate
  - Digital Inputs
    - Digital Input Options
  - Serial Ports
- Imaging Service

- IrCutfilterConfiguration
- Tampering Events
  - Image Too Blurry
  - Image Too Dark
  - Image Too Bright
  - Global Scene Change
- Motion Alarm
- Focus Control

- Analytics Service
  - Rule Engine
    - Rule Options
    - Supported Rules
      - Motion Region Detector Rule
      - Face Recognition Rule
      - License Plate Recognition Rule
  - Analytics Modules
    - Analytics Module Options
    - Supported Metadata
      - Geo Location
      - Object Classification
        - Vehicle Info
        - Human Face
        - Human Body
        - License Plate Info

- Image Sending
  - Embedded Image Sending Type
  - Local Storage Image Sending Type
  - Remote Storage Image Sending Type
- Recording Control Service
  - Dynamic Recordings
  - Dynamic Tracks
  - Audio Recording
  - Recording Options
  - tns1:RecordingConfig/DeleteTrackData
  - Metadata Recording
  - Encoding
    - JPEG
    - H.264
    - MPEG4
- Recording Search Service
  - Metadata Search
  - PTZ Position Search
- Replay Service
  - Reverse Replay
  - RTP/RTSP/TCP
- Receiver Service
- Door Control Service
  - Door Entity
    - Access Door

    - Lock Door
    - Unlock Door
    - Double Lock Door
    - Block Door
    - Lock Down Door
    - Lock Open Door
    - Door Monitor
    - Lock Monitor
    - Double Lock Monitor
    - Alarm
    - Tamper
    - Fault
- Door Events support
    - Door/State/DoorMode
    - Door/State/DoorPhysicalState
    - Door/State/LockPhysicalState
    - Door/State/DoubleLockPhysicalState
    - Door/State/DoorAlarm
    - Door/State/DoorTamper
    - Door/State/DoorFault
    - Configuration/Door/Changed
    - Configuration/Door/Removed
- Door Management
- Client Supplied Token (Door)

- Access Control Service
  - Area Entity
  - Access Point Entity
    - Enable/Disable Access Point
    - Duress
    - Access Taken
    - External Authorization
    - Anonymous Access
    - Identifier Access
  - Access Point Events
    - AccessControl/AccessGranted/Anonymous
    - AccessControl/AccessGranted/Credential
    - AccessControl/AccessTaken/Anonymous
    - AccessControl/AccessTaken/Credential
    - AccessControl/AccessNotTaken/Anonymous
    - AccessControl/AccessNotTaken/Credential
    - AccessControl/Denied/Anonymous
    - AccessControl/Denied/Credential
    - AccessControl/Denied/CredentialNotFound/Card
    - AccessControl/Denied/CredentialNotFound
    - AccessControl/Duress
    - AccessControl/Request/Anonymous
    - AccessControl/Request/Credential
    - AccessControl/Request/Timeout
    - AccessPoint/State/Enabled

- Configuration/AccessPoint/Changed
- Configuration/AccessPoint/Removed
- Configuration/Area/Changed
- Configuration/Area/Removed
- AccessControl/AccessGranted/Identifier
- AccessControl/Denied/Identifier

- Access Point Management
- Area Management
- Client Supplied Token (Area, Access Point)

- Security Configuration Service
  - Keystore features support
    - RSA Key Pair Generation
    - PKCS10 External Certification with RSA
    - Self-Signed Certificate Creation with RSA
    - Passphrase Management
    - PKCS8 Container Upload
    - PKCS12 Container Upload
    - CRLs
    - Certification path validation policies
    - TLS WWW client auth extended key usage extension
    - No Private Key Sharing
  - TLS features support
    - TLS Server
    - TLS client authentication

- Enabled TLS Versions

- Credential Service
  - Supported Identifier Types
    - pt:Card
    - pt:PIN
    - pt:Fingerprint
    - pt:Face
    - pt:Iris
    - pt:Vein
  - Credential Validity
  - Credential Access Profile Validity
  - Validity Supports Time Value
  - Reset Antipassback Violation
  - Client Supplied Token
  - Whitelist
  - Blacklist
  - SupportedExemptionType
    - pt:ExemptFromAuthentication
- Access Rules Service
  - Multiple Schedules per Access Point
  - Client Supplied Token
- Schedule Service
  - Extended Recurrence
  - Special Days

- State Reporting
- Client Supplied Token

- Provisioning Service
- Thermal Service
- Auxiliary Commands
  - tt:Wiper|On
  - tt:Wiper|Off
  - tt:Washer|On
  - tt:Washer|Off
  - tt:WashingProcedure|On
  - tt:WashingProcedure|Off
  - tt:IRLamp|On
  - tt:IRLamp|Off
  - tt:IRLamp|Auto

The coverage of the discovery procedure will be evolving in accordance with [ONVIF Test] version evolution and addition of Profile Specification.

1.1.2 Pre-Configuration Procedure

Pre-configuration procedure includes logic description for the following items:

- Enabling of IPv6 network configuration

Pre-configuration procedure shall have no impact on conformance results even in the case of failures.

2 Normative References

- [ONVIF Network Interface Specs] ONVIF Network Interface Specification documents:
  https://www.onvif.org/profiles/specifications/
- [ONVIF Conformance] ONVIF Conformance Process Specification:
  https://www.onvif.org/profiles/conformance/
- [ONVIF Test] ONVIF Device Test Specification:
  https://www.onvif.org/profiles/conformance/device-test/
- [ONVIF Base Test] ONVIF Base Device Test Specification:
  https://www.onvif.org/profiles/conformance/device-test/
- [ONVIF Media Test] ONVIF Test Specification:
  https://www.onvif.org/profiles/conformance/device-test/
- [ONVIF PTZ Test] ONVIF PTZ Test Specification:
  https://www.onvif.org/profiles/conformance/device-test/
- [ONVIF Imaging Test] ONVIF Imaging Test Specification:
  https://www.onvif.org/profiles/conformance/device-test/
- [ONVIF Recording Control Test] ONVIF Recording Control Test Specification:
  https://www.onvif.org/profiles/conformance/device-test/
- [ONVIF Receiver Test] ONVIF Receiver Test Specification:
  https://www.onvif.org/profiles/conformance/device-test/
- [ONVIF Replay Control Test] ONVIF Replay Control Test Specification:
  https://www.onvif.org/profiles/conformance/device-test/
- [ONVIF Recording Search Test] ONVIF Recording Search Test Specification:
  https://www.onvif.org/profiles/conformance/device-test/
- [ONVIF Access Test] ONVIF Access Control Test Specification:
  https://www.onvif.org/profiles/conformance/device-test/

- [ONVIF Door Test] ONVIF Door Control Test Specification:

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

- [ONVIF Security Configuration Test] ONVIF Security Configuration Device Test Specification:

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

- [ONVIF Credential Test] ONVIF Credential Test Specification:

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

- [ONVIF Access Rules Test] ONVIF Access Rules Test Specification:

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

- [ONVIF Schedule Test] ONVIF Schedule Test Specification:

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

3 Informative References

- [ONVIF Test Case Summary] ONVIF Test Case Summary for Profile Conformance

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

- [ONVIF Profile S] ONVIF Profile S Specification

  https://www.onvif.org/profiles/profile-s/

- [ONVIF Profile G] ONVIF Profile G Specification

  https://www.onvif.org/profiles/profile-g/

- [ONVIF Profile C] ONVIF Profile C Specification

  https://www.onvif.org/profiles/profile-c/

- [ONVIF Profile Q] ONVIF Profile Q Specification

  https://www.onvif.org/profiles/profile-q/

- [ONVIF Profile A] ONVIF Profile A Specification

  https://www.onvif.org/profiles/profile-a/

- [ONVIF Profile T] ONVIF Profile T Specification

  https://www.onvif.org/profiles/profile-t/

4 Terms and Definitions

4.1 Definitions

This section defines terms that are specific to the ONVIF Feature Discovery.

Capability The capability commands allow a client to ask for the services provided by an ONVIF device.

Network A network is an interconnected group of devices communicating using the Internet protocol.

ONVIF Client ONVIF Device Test Tool in the context of this document

Key A key is an input to a cryptographic algorithm. Sufficient randomness of the key is usually a necessary condition for the security of the algorithm. This specification supports RSA key pairs as keys.

Key Pair A key that consists of a public key and (optionally) a private key.

RSA key pair A key pair that is accepted as input by the RSA algorithm.

Certificate A certificate as used in this specification binds a public key to a subject entity. The certificate is digitally signed by the certificate issuer (the certification authority) to allow for verifying its authenticity

4.2 Abbreviations

This section describes abbreviations used in this document.

DUT Device Under Test

DNS Domain Name System

DHCP Dynamic Host Configuration Protocol

IP Internet Protocol

IPv4 Internet Protocol version 4

IPv6 Internet Protocol version 6

NTP Network Time Protocol

RTCP RTP Control Protocol

RTSP Real Time Streaming Protocol

RTP Real-time Transport Protocol

URI Uniform Resource Identifier

TLS Transport Layer Security

5 Discovery Procedure

This section describes policy on how ONVIF Client assumes that a certain feature is supported or not, followed by respective feature discovery procedures.

5.1 General Policy

ONVIF Client will issue capability query command (GetCapabilities or GetServices/ GetServiceCapabilities depending on DUT possibilities) to get to know whether a certain feature is supported by DUT. If the DUT returns correct response, ONVIF Client determines whether a feature in question is supported or not, based on the content of the response. If the DUT returns unexpected response or it does not return any response, ONVIF Client assumes that the capability query command which is being used is not supported by DUT.

There are a number of functions which are defined as conditionally required. And there are some of functions which are not present in any capability response fields. As for these commands, ONVIF Client will issue the very function command to determine whether the function is supported or not. In the case that the DUT returns a correct response to indicate no function support such as SOAP fault env:Receiver/ter:ActionNotSupported/ter:NoSuchService, ONVIF Client assumes that the function is not supported by DUT. In case the DUT returns an unexpected response or it does not return any response, ONVIF Client will mark the function support in question as undefined.

After going through all the feature discovery steps based on the above general policy, what are marked as supported and undefined will be processed as supported features during conformance testing.

5.2 Feature Support Criteria

Feature support criteria in the specification are defined using the following table format outlined in Table 5.1

Table 5.1. Feature Support Criteria Description Outline Used in This Specification

Criterion Item	Criteria_item_description	
Feature <(for each Entity_type)>	**Supported**	**Not Supported**
Feature_name1	**Supported_criteria_feature1**	**NotSupported_criteria_feature1**
Feature_name2	**Supported_criteria_feature2**	**NotSupported_criteria_feature2**
...	**...**	**...**
Feature_nameN	**Supported_criteria_featureN**	**NotSupported_criteria_featureN**

The **Feature** column includes a list of features that is defined in current table.

The **Feature** cell could have optional brackets with definition of entity type for which the features will be defined (for example, for each PTZ Node). If there are no brackets the features are general for the DUT.

The **Criterion item** field contains item description which value will be used as criteria to check feature support.

The **Supported** column includes conditions when feature in the same row will be assumed as supported depending on criteria item value.

The **Not Supported** column includes conditions when feature in the same row will be assumed as not supported depending on criteria item value.

5.3 Discovery Types Support

From the first version of ONVIF Core Specification document, Device Type that is required for Discovery functionality was "dn:NetworkVideoTransmitter". Device Type was modified in the later version of [ONVIF Network Interface Specs] to "tds:Device". Which Device Type shall be used by DUT is defined in the scope of [ONVIF Profile S], [ONVIF Profile G] and [ONVIF Profile C].The following procedure discovers which Types are supported by DUT.

Discovery Procedure:

1. ONVIF Client invokes Unicast Probe request (empty Types, empty Scopes) to get ProbeMatches response.
2. ONVIF Client receives ProbeMatch and checks features support as defined in Table 5.2.

Note: If DUT does not return ProbeMatch or ProbeMatch <d:Types> does not contain neither "dn:NetworkVideoTransmitter" nor "tds:Device", the following features will be marked as undefined:

- WS-Discovery\Types\tds:Device
- WS-Discovery\Types\dn:NetworkVideoTransmitter

Table 5.2. Discovery Types

Criterion Item	<d:Types> in ProbeMatch response	
Feature	**Supported**	**Not Supported**
tds:Device	**Contains "tds:Device"**	**Does not contain "tds:Device"**

Criterion Item	<d:Types> in ProbeMatch response	
Feature	**Supported**	**Not Supported**
dn:NetworkVideoTransmitter	**Contains "dn:NetworkVideoTransmitter**	**Does not contain "dn:NetworkVideoTransmitter"**

5.4 Capabilities

From the first version of ONVIF Core Specification document, GetCapabilities command defined in Device Management Service was the only command to get to know the various feature capabilities by DUT. This capability query scheme was modified in the later version of [ONVIF Network Interface Specs] in order to enhance its scalability. In the case when only GetCapabilities command is supported by DUT, the following procedure focuses on GetCapabilities commands as supported capability query method by DUT. In the case when GetServices/GetServiceCapabilities commands are supported by DUT the following procedure focuses on GetServices commands as supported capability query method, which provides more possibilities for feature discovery.

Discovery Procedure:

1. ONVIF Client invokes GetCapabilitiesRequest message without any authentication to retrieve the capabilities and check GetCapabilities command support by DUT.
   a. If DUT returns correct GetCapabilitiesResponse message. Go to step 4
   b. If DUT returns fault message (any SOAP fault except Sender/NotAuthorized) or it does not return any response. Go to step 4.
   c. If DUT returns fault message (SOAP fault Sender/NotAuthorized), go to step 2.
   d. If DUT returns HTTP 401 Unauthorized error, go to step 3.
2. ONVIF Client invokes GetCapabilitiesRequest message with WS-UsernameToken authentication to retrieve the capabilities and check GetCapabilities command support of DUT.
   a. If DUT returns correct GetCapabilitiesResponse message. Go to step 4.
   b. If DUT does not return correct GetCapabilitiesResponse message. Go to step 4.
3. ONVIF Client invokes GetCapabilitiesRequest message with HTTP Digest authentication to retrieve the capabilities and check GetCapabilities command support of DUT.
   a. If DUT returns correct GetCapabilitiesResponse message. Go to step 4.
   b. If DUT does not return correct GetCapabilitiesResponse message. Go to step 4.

4. ONVIF Client invokes GetServicesRequest message without any authentication to retrieve the capabilities and check GetServices command support by DUT.
   a. If DUT returns GetServicesResponse message. Go to step 7.
   b. If DUT returns fault message (any SOAP fault except Sender/NotAuthorized) or it does not return any response. Go to step 7.
   c. If DUT returns fault message (Sender/NotAuthorized), go to step 5.
   d. If DUT returns HTTP 401 Unauthorized error, go to step 6.
5. ONVIF Client invokes GetServicesRequest message with WS-UsernameToken authentication to retrieve the capabilities and check GetServices command support of DUT.
   a. If DUT returns GetServicesResponse message. Go to step 7.
   b. If DUT does not return GetServicesResponse message. Go to step 7.
6. ONVIF Client invokes GetServicesRequest message with HTTP Digest authentication to retrieve the capabilities and check GetServices command support by DUT.
   a. If DUT returns GetServicesResponse message.
   b. If DUT does not return GetServicesResponse message.
7. ONVIF Client checks features support as defined in Table 5.3.

Note: If both GetCapabilities and GetServices functions are defined as unsupported, other features will be marked as undefined.

Note: Next steps will depend on GetServices support. If GetServices is supported by DUT, then 5.5 Discovery Procedure (GetServices and GetServiceCapabilities) will be used. If only GetCapabilities is supported by DUT then 5.6 Discovery Procedure (GetCapabilities) will be used.

Table 5.3. Capabilities

Criterion Item	GetServicesResponse message and GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
GetCapabilities	**GetCapabilitiesResponse was received**	**No GetCapabilitiesResponse was received (fault was received or DUT does not return any response)**
GetServices	**GetServicesResponse was received**	**No GetServicesResponse was received (fault was**

Criterion Item	GetServicesResponse message and GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
		received or DUT does not return any response)

5.5 Discovery Procedure (GetServices and GetServiceCapabilities)

If GetServices is supported by the DUT, then GetServices and GetServiceCapabilities commands will be used for feature discovery procedure. The following provides with the functionality discovery procedure for this case.

5.5.1 Device Management Service Capabilities

There are various device management functions defined in [ONVIF Core] as a part of ONVIF Device Management Service.

In the first version of [ONVIF Core], WS-UsernameToken support was the only method defined as a mandatory feature for user authentication. This has been changed in the later version of [ONVIF Core] where it also defines the HTTP digest authentication support as a mandatory feature.

The following discovery procedure will be performed for ONVIF Client to determine which user authentication function will be used in conformance testing. Also network configuration, security support, WS-Discovery features, and system logging will be done during this discovery procedure.

Discovery Procedure:

1. ONVIF Client invokes GetServiceCapabilitiesRequest message for Device Management without any authentication to retrieve the Device Management Service capabilities of the DUT.

   a. If the DUT returns correct GetServiceCapabilitiesResponse message, go to step 4.

   b. If the DUT returns fault message (SOAP fault Sender/NotAuthorized), go to step 2.

   c. If the DUT returns HTTP 401 Unauthorized error, go to step 3.

2. ONVIF Client invokes GetServiceCapabilitiesRequest message with WS-UsernameToken authentication to retrieve the Device Management Service capabilities of the DUT. Go to the step 4.

3. ONVIF Client invokes GetServiceCapabilitiesRequest message with HTTP Digest authentication to retrieve the Device Management Service capabilities of the DUT. Go to the step 4.

4. ONVIF Client checks features support as defined in Table 5.4.

Note: If the DUT returns no response for step 1 or response differs from the provided in a, b, and c items at step 1, then all Device Management features will be marked as undefined.

Note: If the DUT does not return GetServiceCapabilitiesResponse message for steps 2 or 3, then all Device Management features will be marked as undefined.

Note: If both of WS-UsernameToken and Digest are defined as unsupported, WS-UsernameToken will be used for test performance and discovering of following features.

Note: If HTTP digest authentication is assumed as supported, the HTTP digest authentication scheme will be used in the following feature discovery procedure whenever necessary as well as in conformance testing.

Table 5.4. Device Capabilities Configuration Functionality in Device Management Service (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
NTP	**Capabilities.Network.NTP > 0**	**Skipped Capabilities.Network.NTP or Capabilities.Network.NTP = 0**
IPv6	**Capabilities.Network. IPVersion6 = true**	**Skipped Capabilities.Network. IPVersion6 or Capabilities.Network. IPVersion6 = false**
Zero Configuration	**Capabilities.Network. ZeroConfiguration = true**	**Skipped Capabilities.Network. ZeroConfiguration or Capabilities.Network. ZeroConfiguration = false**
Dynamic DNS	**Capabilities.Network.DynDNS = true**	**Skipped Capabilities.Network. DynDNS or Capabilities.Network. DynDNS = false**
IP Filter	**Capabilities.Network.IPFilter = true**	**Skipped Capabilities.Network. IPFilter or Capabilities.Network. IPFilter = false**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Stateful IPv6 DHCP	**Capabilities.Network.DHCPv6 = true**	**Skipped Capabilities.Network. DHCPv6 or Capabilities.Network. DHCPv6 = false**
WS-UsernameToken	**Capabilities.Security. UsernameToken = true**	**Skipped Capabilities.Security. UsernameToken or Capabilities. Security.UsernameToken = false**
HTTP Digest	**Capabilities.Security. HttpDigest = true**	**Skipped Capabilities.Security. HttpDigest or Capabilities. Security.HttpDigest = false**
Maximum Users	**Capabilities.Security. MaxUsers element is present**	**Capabilities.Security. MaxUsers element is not present**
Default Access Policy	**Capabilities.Security. DefaultAccessPolicy = true**	**Skipped Capabilities.Security. HttpDigest or Capabilities. Security. DefaultAccessPolicy = false**
Remote User Handling	**Capabilities.Security. RemoteUserHandling = true**	**Skipped Capabilities. Security. RemoteUserHandling or Capabilities.Security. RemoteUserHandling = false**
Maximum Username Length	**Capabilities.Security. MaxUsernameLength element is present**	**Capabilities.Security. MaxUsernameLength element is not present**
Maximum Password Length	**Capabilities.Security. MaxPasswordLength element is present**	**Capabilities.Security. MaxPasswordLength element is not present**
TLS1.0	**Capabilities.Security. TLS1.0 = true**	**Skipped Capabilities.Security. TLS1.0 or Capabilities.Security. TLS1.0 = false**
TLS1.1	**Capabilities.Security. TLS1.1 = true**	**Skipped Capabilities.Security. TLS1.1 or**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
		Capabilities.Security. TLS1.1 = false
TLS1.2	**Capabilities.Security. TLS1.2 = true**	**Skipped Capabilities.Security. TLS1.2 or Capabilities.Security. TLS1.2 = false**
Bye Message	**Capabilities.System. DiscoveryBye = true**	**Skipped Capabilities.System. DiscoveryBye or Capabilities. System.DiscoveryBye = false**
System logging	**Capabilities.System. SystemLogging = true**	**Skipped Capabilities.System. SystemLogging or Capabilities. System.SystemLogging = false**
Http Firmware Upgrade	**Capabilities.System. HttpFirmwareUpgrade = true**	**Skipped Capabilities.System. HttpFirmwareUpgrade or Capabilities. System. HttpFirmwareUpgrade = false**
Http System Backup	**Capabilities.System. HttpSystemBackup = true**	**Skipped Capabilities.System. HttpSystemBackup or Capabilities. System. HttpSystemBackup = false**
Http System Logging	**Capabilities.System. HttpSystemLogging = true**	**Skipped Capabilities.System. HttpSystemLogging or Capabilities. System. HttpSystemLogging = false**
Http Support Information	**Capabilities.System. HttpSupportInformation = true**	**Skipped Capabilities.System. HttpSupportInformation or Capabilities. System. HttpSupportInformation = false**
Auxiliary Commands	**Capabilities.Misc. AuxiliaryCommands is not empty**	**Capabilities.Misc. AuxiliaryCommands is empty or Capabilities.Misc. AuxiliaryCommands is not present**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
tt:Wiper|On	**Capabilities.Misc. AuxiliaryCommands contains tt:Wiper|On**	**Capabilities.Misc. AuxiliaryCommands does not contain tt:Wiper|On**
tt:Wiper|Off	**Capabilities.Misc. AuxiliaryCommands contains tt:Wiper|Off**	**Capabilities.Misc. AuxiliaryCommands does not contain tt:Wiper|Off**
tt:Washer|On	**Capabilities.Misc. AuxiliaryCommands contains tt:Washer|On**	**Capabilities.Misc. AuxiliaryCommands does not contain tt:Washer|On**
tt:Washer|Off	**Capabilities.Misc. AuxiliaryCommands contains tt:Washer|Off**	**Capabilities.Misc. AuxiliaryCommands does not contain tt:Washer|Off**
tt:WashingProcedure|On	**Capabilities.Misc. AuxiliaryCommands contains tt:WashingProcedure|On**	**Capabilities.Misc. AuxiliaryCommands does not contain tt:WashingProcedure|On**
tt:WashingProcedure|Off	**Capabilities.Misc. AuxiliaryCommands contains tt:WashingProcedure|Off**	**Capabilities.Misc. AuxiliaryCommands does not contain tt:WashingProcedure|Off**
tt:IRLamp|On	**Capabilities.Misc. AuxiliaryCommands contains tt:IRLamp|On**	**Capabilities.Misc. AuxiliaryCommands does not contain tt:IRLamp|On**
tt:IRLamp|Off	**Capabilities.Misc. AuxiliaryCommands contains tt:IRLamp|Off**	**Capabilities.Misc. AuxiliaryCommands does not contain tt:IRLamp|Off**
tt:IRLamp|Auto	**Capabilities.Misc. AuxiliaryCommands contains tt:IRLamp|Auto**	**Capabilities.Misc. AuxiliaryCommands does not contain tt:IRLamp|Auto**

5.5.2 HTTPS Support

The following is the procedure to determine the function support.

Pre-requisite: ONVIF Client and DUT

- If DUT supports HTTPS, then HTTPS is configured on the DUT in case TLS Server is not supported by DUT. Security Configuration Service is received from the DUT, if TLS Server is supported by DUT.

Discovery Procedure:

- If DUT supports **Security Configuration Service\TLS features support\TLS Server** feature as defined in Table 5.65, then HTTPS feature is assumed as supported.
- Otherwise, ONVIF Client invokes GetNetworkProtocols request message to retrieve network protocols supported by DUT.
- The DUT returns GetNetworkProtocolsResponse. ONVIF Client checks features support as defined in Table 5.5.

Note: If the DUT does not return GetNetworkProtocolsResponse, ONVIF Client assumes that RTP/RTSP/HTTPS function support is marked as undefined.

Table 5.5. HTTPS Support (GetServices)

Criterion Item	GetNetworkProtocolsResponse	
Feature	**Supported**	**Not Supported**
HTTPS	**Includes NetworkProtocols element with Name = HTTPS and with Enabled=true or DUT supports TLS Server feature**	**Does not include NetworkProtocols element with Name = HTTPS or includes NetworkProtocols element with Name = HTTPS and with Enabled=false when DUT does not support TLS Server feature**

5.5.3 I/O Functionality in Device Management Service

I/O related functionality support can be retrieved by checking correspondent element of GetCapabilitiesResponse. The following is the procedure to determine the function support.

Discovery Procedure:

1. ONVIF Client invokes GetCapabilitiesRequest to check I/O functionality support.
2. ONVIF Client receives GetCapabilitiesResponse and checks features support as defined in Table 5.6.
3. ONVIF Client invokes GetRelayOutputsRequest message to retrieve a relay output list.

4. The DUT returns GetRelayOutputsResponse with a list of relay outputs.
5. ONVIF Client invokes SetRelayOutputSettingsRequest message (RelayOutputToken = "[first token from GetRelayOutputsResponse]", Properties.Mode = "Bistable", Properties.DelayTime = "PT30S", Properties.IdleState = "open").
6. ONVIF Client receives SetRelayOutputSettingsResponse.
7. ONVIF Client invokes SetRelayOutputSettingsRequest message (RelayOutputToken = "[first token from GetRelayOutputsResponse]", Properties.Mode = "Bistable", Properties.DelayTime = "PT30S", Properties.IdleState = "closed").
8. ONVIF Client receives SetRelayOutputSettingsResponse.
9. ONVIF Client invokes SetRelayOutputSettingsRequest message (RelayOutputToken = "[first token from GetRelayOutputsResponse]", Properties.Mode = "Monostable", Properties.DelayTime = "PT30S", Properties.IdleState = "open").
10. ONVIF Client receives SetRelayOutputSettingsResponse.
11. ONVIF Client invokes SetRelayOutputSettingsRequest message (RelayOutputToken = "[first token from GetRelayOutputsResponse]", Properties.Mode = "Monostable", Properties.DelayTime = "PT30S", Properties.IdleState = "closed").
12. ONVIF Client receives SetRelayOutputSettingsResponse.
13. ONVIF Client checks features support as defined in Table 5.7.

Note: Absence of Capabilities.Device.IO element in the GetCapabilitiesResponse will be defined as absence of Capabilities.Device.IO.RelayOutputs.

Note: If Capabilities.Device element is not included in the GetCapabilitiesResponse, Relay Outputs feature will be marked as unsupported.

Note: If the DUT does not return GetRelayOutputsResponse or list of relay outputs in the GetRelayOutputsResponse is empty, Relay Outputs features will be marked as undefined.

Note: If GetCapabilities command is not supported by the DUT I/O feature for Device Management Service will be defined as unsupported.

Table 5.6. Relay Outputs in Device Management Service (GetServices)

Criterion Item	GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
RelayOutputs	Capabilities.Device.IO. RelayOutputs > 0	Skipped Capabilities.Device.IO. RelayOutputs or Capabilities. Device.IO.RelayOutputs = 0

Table 5.7. Relay Outputs Mode and Idle State in Device Management Service (GetServices)

Criterion Item	SetRelayOutputSettingsResponse	
Feature	**Supported**	**Not Supported**
Bistable Mode/Open Idle State	**DUT returns SetRelayOutputSettings Response for step 6.**	**DUT returns any SOAP fault for step 6.**
Bistable Mode/Closed Idle State	**DUT returns SetRelayOutputSettings Response for step 8**	**DUT returns any SOAP fault for step 8.**
Bistable Mode	**DUT returns SetRelayOutputSettings Response for step 6 or 8.**	**DUT returns any SOAP fault for step 6 and 8.**
Monostable Mode/ Open Idle State	**DUT returns SetRelayOutputSettings Response for step 9.**	**DUT returns any SOAP fault for step 9.**
Monostable Mode/ Closed Idle State	**DUT returns SetRelayOutputSettings Response for step 11.**	**DUT returns any SOAP fault for step 11.**
Monostable Mode	**DUT returns SetRelayOutputSettings Response for step 9 or 11.**	**DUT returns any SOAP fault for step 9 and 11.**

5.5.4 Monitoring Events Support

Monitoring Events support under Device Control Service is determined according to the following procedure.

Pre-requisite: ONVIF Client and DUT

- This procedure assumes that GetEventPropertiesResponse has already been retrieved via preceding procedure described in Section 5.5.7.1.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.8.

Note: If the DUT does not return Event Service or GetEventPropertiesResponse message, then the following features will be marked as undefined:

- Monitoring/ProcessorUsage

- Monitoring/OperatingTime/LastReset
- Monitoring/OperatingTime/LastReboot
- Monitoring/OperatingTime/LastClockSynchronization
- Monitoring/Backup/Last
- Device/HardwareFailure/TemperatureCritical
- Device/HardwareFailure/FanFailure
- Device/HardwareFailure/PowerSupplyFailure
- Device/HardwareFailure/StorageFailure

Table 5.8. Monitoring Events Support (GetServices)

Criterion Item	GetEventPropertiesResponse	
Feature	**Supported**	**Not Supported**
Monitoring/ProcessorUsage	**Contains tns1:Monitoring/ ProcessorUsage Event topic**	**Does not contain tns1:Monitoring/ ProcessorUsage Event topic**
Monitoring/ OperatingTime/LastReset	**Contains tns1:Monitoring/ OperatingTime/ LastReset Event topic**	**Does not contain tns1:Monitoring/ OperatingTime/ LastReset Event topic**
Monitoring/ OperatingTime/LastReboot	**Contains tns1:Monitoring/ OperatingTime/ LastReboot Event topic**	**Does not contain tns1:Monitoring/ OperatingTime/ LastReboot Event topic**
Monitoring/OperatingTime/ LastClockSynchronization	**Contains tns1:Monitoring/ OperatingTime/ LastClockSynchronization Event topic**	**Does not contain tns1:Monitoring/ OperatingTime/ LastClockSynchronization Event topic**
Monitoring/Backup/Last	**Contains tns1:Monitoring/ Backup/Last Event topic**	**Does not contain tns1:Monitoring/ Backup/Last Event topic**
Device/HardwareFailure/ TemperatureCritical	**Contains tns1:Device/ HardwareFailure/ TemperatureCritical Event topic**	**Does not contain tns1:Device/ HardwareFailure/ TemperatureCritical Event topic**

Criterion Item	GetEventPropertiesResponse	
Feature	**Supported**	**Not Supported**
Device/HardwareFailure/ FanFailure	**Contains tns1:Device/ HardwareFailure/ FanFailure Event topic**	**Does not contain tns1:Device/ HardwareFailure/ FanFailure Event topic**
Device/HardwareFailure/ PowerSupplyFailure	**Contains tns1:Device/ HardwareFailure/ PowerSupplyFailure Event topic**	**Does not contain tns1:Device/ HardwareFailure/ PowerSupplyFailure Event topic**
Device/HardwareFailure/ StorageFailure	**Contains tns1:Device/ HardwareFailure/ StorageFailure Event topic**	**Does not contain tns1:Device/ HardwareFailure/ StorageFailure Event topic**

5.5.5 Media Service – General

Media Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.9.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/media/wsdl" namespace ONVIF Client will use service with the latest version.

Note: If Media service is not supported, the following feature discovery (Media Service features support) will be skipped.

Table 5.9. Media Service – General (GetServices)

Criterion Item	GetServicesResponse	
Feature	**Supported**	**Not Supported**
Media Service	**Includes service with "http://www.onvif.org/ver10/media/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/media/wsdl" namespace**

5.5.5.1 Media Service – Video Encoding Support

Video encoding function support in Media Service is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetVideoEncoderConfigurationOptionsRequest (no ConfigurationToken, no ProfileToken) message to retrieve all supported codecs.
2. The DUT returns GetVideoEncoderConfigurationOptionsResponse with a list of supported codecs. ONVIF Client checks features support as defined in Table 5.10.

Note: If the DUT does not return GetVideoEncoderConfigurationOptionsResponse, MPEG4 and H.264 feature will be marked as undefined.

Table 5.10. Media Service – Video Encoding Support (GetServices)

Criterion Item	GetVideoEncoderConfigurationOptionsResponse	
Feature	**Supported**	**Not Supported**
JPEG	**Mandatory functionality**	**-**
MPEG-4	**Includes Options.MPEG4**	**Does not include Options.MPEG4**
H.264	**Includes Options.H264**	**Does not include Options.H264**

5.5.5.2 Media Service – Audio Encoding Support

Audio encoding function support in Media Service is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetAudioEncoderConfigurationOptionsRequest (no ConfigurationToken, no ProfileToken) message to retrieve all supported audio codecs.
2. The DUT returns GetAudioEncoderConfigurationOptionsResponse with a list of supported codecs or SOAP fault. ONVIF Client checks features support as defined in Table 5.11.

Note: If the DUT returns no response for GetAudioEncoderConfigurationOptionsRequest, Audio encoding feature will be marked as undefined.

Table 5.11. Media Service – Audio Encoding Support (GetServices)

Criterion Item	GetAudioEncoderConfigurationOptionsResponse	
Feature	**Supported**	**Not Supported**
Audio encoding	**DUT returns GetAudioEncoderConfigurationOptionsResponse**	**DUT returns any SOAP fault**
G.711	**DUT returns GetAudioEncoderConfigurationOptionsResponse**	**DUT returns any SOAP fault**
G.726	**Includes Options.Options.Encoding = "G726"**	**Does not include Options.Options.Encoding = "G726"**
AAC	**Includes Options.Options.Encoding = "AAC"**	**Does not include Options.Options.Encoding = "AAC"**

5.5.5.3 Media Service – Real-Time Streaming

Real-time streaming support in Media Service is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetServiceCapabilitiesRequest message to check Multicast streaming capability support by the DUT.
2. The DUT returns GetServiceCapabilitiesResponse. ONVIF Client checks features support as defined in Table 5.12.

Note: If the DUT does not return GetServiceCapabilitiesResponse, then Real-time streaming feature and features from Section 5.5.5.4 will be marked as undefined. Procedure described in Section 5.5.5.4 will be skipped.

Note: If the DUT does not support Real-time streaming feature, all features from Section 5.5.5.4 will be marked as unsupported. Procedure described in Section 5.5.5.4 will be skipped.

Table 5.12. Media Service – Supported Real-Time Streaming Setup (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Real-time streaming	**Skipped Capabilities. StreamingCapabilities. NoRTSPStreaming or Capabilities. StreamingCapabilities. NoRTSPStreaming = false**	**Capabilities. StreamingCapabilities. NoRTSPStreaming = true**

5.5.5.4 Media Service – Supported Real-Time Streaming Setup

Which Real-time streaming Setup features is supported under Real-time Streaming is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServiceCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.5.5.3.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.13.

Table 5.13. Media Service – Supported Real-Time Streaming Setup (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
RTP/UDP	**Mandatory functionality, if Real-time Streaming supported**	**-**
RTP/RTSP/HTTP	**Mandatory functionality, if Real-time Streaming supported**	**-**
RTP/RTSP/TCP	**Capabilities. StreamingCapabilities. RTP_RTSP_TCP = true**	**Skipped Capabilities. StreamingCapabilities. RTP_RTSP_TCP or Capabilities.**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
		StreamingCapabilities. RTP_RTSP_TCP = false
RTP-Multicast/UDP	**Capabilities. StreamingCapabilities. RTPMulticast = true**	**Skipped Capabilities. StreamingCapabilities. RTPMulticast or Capabilities. StreamingCapabilities. RTPMulticast = false**

5.5.5.5 Media Service - GetSnapshotUri

GetSnapshotUri function support is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetProfilesRequest message to retrieve existing Media Profiles list.
2. The DUT returns GetProfilesResponse with the list of existing Media Profiles.
3. ONVIF Client looks for ready-to-use profile (a profile with VideoSourceConfiguration and VideoEncoderConfiguration in the GetProfilesResponse. If there are no ready-to-use profiles found in the GetProfilesResponse, ONVIF Client marks GetSnapshotUri support by DUT as undefined.
4. ONVIF Client invokes GetSnapshotUriRequest (ProfileToken = found ready-to-use profile token) message to get Snapshot URI.
5. The DUT returns GetSnapshotUriResponse or SOAP fault. ONVIF Client checks features support as defined in Table 5.14.

Note: If no GetProfilesResonse is returned by the DUT, GetSnapshotUri function support by the DUT is marked as undefined.

Note: If no GetSnapshotUriResponse is returned by the DUT, GetSnapshotUri function support by the DUT is marked as undefined.

Table 5.14. Media Service – GetSnapshotUri (GetServices)

Criterion Item	GetSnapshotUriResponse	
Feature	**Supported**	**Not Supported**
GetSnapshotUri	**DUT returns GetSnapshotUriResponse**	**DUT returns any SOAP fault**

5.5.5.6 Media Service – Audio Outputs Support

Audio outputs support in conjunction with its Audio decoding function is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetAudioOutputsRequest message to retrieve Audio outputs list.
2. The DUT returns GetAudioOutputsResponse or SOAP fault. ONVIF Client checks features support as defined in Table 5.15. Go to the next feature definition.
3. ONVIF Client invokes GetAudioDecoderConfigurationOptionsRequest (no ConfigurationToken, no ProfileToken) message to retrieve all supported Audio codec's for decoding by DUT.
4. The DUT returns GetAudioDecoderConfigurationOptionsResponse. ONVIF Client checks features support as defined in Table 5.16.

Note: If the DUT does not return GetAudioDecoderConfigurationOptionsResponse, ONVIF Client assumes that G.711, G.726 and AAC Audio decoding function support is marked as undefined.

Table 5.15. Media Service – Audio Outputs Support (GetServices)

Criterion Item	GetAudioOutputsResponse	
Feature	**Supported**	**Not Supported**
Audio output	**DUT returns GetAudioOutputsResponse and there are at least one AudioOutput on the list**	**DUT returns any SOAP fault or GetAudioOutputsResponse and there are no AudioOutput on the list**

Table 5.16. Media Service – Audio Outputs Decoding Support (GetServices)

Criterion Item	GetAudioDecoderConfigurationOptionsResponse	
Feature	**Supported**	**Not Supported**
G.711	**Includes Options.G711DecOptions**	**Does not include Options.G711DecOptions**
G.726	**Includes Options.G726DecOptions**	**Does not include Options.G726DecOptions**
AAC	**Includes Options.AACDecOptions**	**Does not include Options.AACDecOptions**

5.5.6 Media2 Service – General

Media2 Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.17.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver20/media/wsdl" namespace ONVIF Client will use service with the latest version.

Note: If Media2 service is not supported, the following feature discovery (Media2 Service features support) will be skipped.

Table 5.17. Media2 Service – General (GetServices)

Criterion Item	GetServicesResponse	
Feature	**Supported**	**Not Supported**
Media2 Service	**Includes service with "http://www.onvif.org/ver20/media/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver20/media/wsdl" namespace**

Media2 features support

Media2 capabilities support under Media2 Service is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetServiceCapabilitiesRequest message to retrieve Media2 Service capabilities.
2. The DUT returns GetServiceCapabilitiesResponse with Media2 Service capabilities. ONVIF Client checks features support as defined in Table 5.18.

Note: If the DUT returns no response for GetServiceCapabilitiesRequest, then all features defined in Table 5.18 will be marked as undefined.

Table 5.18. Media2 Service Features Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Video	**Capabilities. ProfileCapabilities. ConfigurationsSupported list contains "VideoEncoder"**	**Capabilities. ProfileCapabilities. ConfigurationsSupported list does not contain "VideoEncoder"**
Audio	**Capabilities. ProfileCapabilities. ConfigurationsSupported list contains "AudioEncoder"**	**Capabilities. ProfileCapabilities. ConfigurationsSupported list does not contain "AudioEncoder"**
Audio Output	**Capabilities. ProfileCapabilities. ConfigurationsSupported list contains "AudioOutput"**	**Capabilities. ProfileCapabilities. ConfigurationsSupported list does not contain "AudioOutput"**
Real-time Streaming	**Capabilities. StreamingCapabilities. RTSPStreaming = true**	**Skipped Capabilities. StreamingCapabilities. RTSPStreaming or Capabilities. StreamingCapabilities. RTSPStreaming = false**
RTP/UDP	**Mandatory functionality, if Real-time Streaming supported**	**-**
RTP/RTSP/HTTP	**Mandatory functionality, if Real-time Streaming supported**	**-**
RTP/RTSP/TCP	**Capabilities. StreamingCapabilities. RTP_RTSP_TCP = true**	**Skipped Capabilities. StreamingCapabilities. RTP_RTSP_TCP or Capabilities. StreamingCapabilities. RTP_RTSP_TCP = false**
RTP-Multicast/UDP	**Capabilities. StreamingCapabilities. RTPMulticast = true**	**Skipped Capabilities. StreamingCapabilities. RTPMulticast or Capabilities.**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
		StreamingCapabilities. RTPMulticast = false
RTSP WebSocket	**Capabilities. StreamingCapabilities contains RTSPWebSocketUri**	**Skipped Capabilities. StreamingCapabilities. RTSPWebSocketUri**
Snapshot Uri	**Capabilities.SnapshotUri = true**	**Skipped Capabilities.SnapshotUri or Capabilities.SnapshotUri = false**
OSD	**Capabilities.OSD = true**	**Skipped Capabilities.OSD or Capabilities.OSD = false**
Video Source Mode	**Capabilities. VideoSourceMode = true**	**Skipped Capabilities. VideoSourceMode or Capabilities. VideoSourceMode = false**
Analytics	**Capabilities. ProfileCapabilities. ConfigurationsSupported list contains "Analytics"**	**Capabilities. ProfileCapabilities. ConfigurationsSupported list does not contain "Analytics"**
Metadata	**Capabilities. ProfileCapabilities. ConfigurationsSupported list contains "Metadata"**	**Capabilities. ProfileCapabilities. ConfigurationsSupported list does not contain "Metadata"**

5.5.6.1 Media2 Service – Video Encoding Support

Pre-requisite:

- DUT supports **Video** feature according to Section 5.5.6.

Video encoding function support in Media2 Service is determined according to the following discovery procedure.

Discovery Procedure:

1. ONVIF Client invokes GetVideoEncoderConfigurationOptionsRequest (no ConfigurationToken, no ProfileToken) message to retrieve all supported codecs.
2. The DUT returns GetVideoEncoderConfigurationOptionsResponse with a list of supported codecs. ONVIF Client checks features support as defined in Table 5.19.

Note: If the DUT does not return GetVideoEncoderConfigurationOptionsResponse, H.264 and H.265 feature will be marked as undefined.

Table 5.19. Media2 Service – Video Encoding Support (GetServices)

Criterion Item	GetVideoEncoderConfigurationOptionsResponse	
Feature	**Supported**	**Not Supported**
H.265	**Includes Options.Encoding= "H265"**	**Does not include Options.Encoding="H265"**
H.264	**Options.Encoding= "H264"**	**Does not include Options.Encoding= "H264"**

5.5.6.2 Media2 Service – Audio Encoding Support

Pre-requisite:

- DUT supports **Audio** feature according to Section 5.5.6.

Audio encoding function support in Media2 Service is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetAudioEncoderConfigurationOptionsRequest (no ConfigurationToken, no ProfileToken) message to retrieve all supported audio codecs.
2. The DUT returns GetAudioEncoderConfigurationOptionsResponse with a list of supported codecs or SOAP fault. ONVIF Client checks features support as defined in Table 5.20.

Note: If the DUT does not return response for GetAudioEncoderConfigurationOptionsRequest, Media2 Audio encoding features (G.711 and AAC) will be marked as undefined.

Table 5.20. Media2 Service – Audio Encoding Support (GetServices)

Criterion Item	GetAudioEncoderConfigurationOptionsResponse	
Feature	**Supported**	**Not Supported**
G.711	**Includes Options.Encoding= "PCMU"**	**Does not include Options.Encoding= "PCMU"**
AAC	**Includes Options.Encoding= "MP4A-LATM" or "MPEG4-GENERIC"**	**Does not include Options.Encoding= "MP4A-LATM" and does not include Options.Encoding = "MPEG4-GENERIC"**

5.5.6.3 Media2 Service – Audio Decoding Support

Pre-requisite:

- DUT supports **Audio Output** feature according to Section 5.5.6.

Audio decoding support is determined according to the following procedure.

Discovery Procedure:

- ONVIF Client invokes GetAudioDecoderConfigurationOptionsRequest (no ConfigurationToken, no ProfileToken) message to retrieve all supported Audio codec's for decoding by DUT.
- The DUT returns GetAudioDecoderConfigurationOptionsResponse. ONVIF Client checks features support as defined in Table 5.21.

Note: If the DUT does not return GetAudioDecoderConfigurationOptionsResponse, ONVIF Client assumes that G.711, and AAC Audio decoding function support is marked as undefined.

Table 5.21. Media2 Service – Audio Outputs Decoder Support (GetServices)

Criterion Item	GetAudioDecoderConfigurationOptionsResponse	
Feature	**Supported**	**Not Supported**
G.711	**Includes Options.Encoding= "PCMU"**	**Does not include Options.Encoding= "PCMU"**
AAC	**Includes Options.Encoding= "MP4A-LATM" or "MPEG4-GENERIC"**	**Does not include Options.Encoding= "MP4A-LATM" and does not include Options.Encoding= "MPEG4-GENERIC"**

5.5.6.4 Media2 Service – RTP/RTSP/HTTPS support

Pre-requisite:

- DUT supports **Real-time Streaming** feature as defined in Media2 Service Features Support Table, Real-time Streaming.

RTP/RTSP/HTTPS support is determined according to the following procedure.

Discovery Procedure:

- If DUT supports **HTTPS** feature as defined in Section 5.5.2, then RTP/RTSP/HTTPS feature is assumed as supported.

5.5.6.5 Media2 Service – OSD Types support

Pre-requisite:

- DUT supports **OSD** according to Table 5.18

OSD Types features supporting is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client retrieves a list of Video Source Configuration by following the procedure mentioned in Annex A.7 with the following input and output parameters
   - out *videoSourceConfList* - Video Source Configurations list
2. For each Video Source Configuration *videoSorceConfig* in *videoSourceConfList*
   2.1. ONVIF Client invokes **GetOSDOptions** with parameters
      - ConfigurationToken =: *videoSorceConfig*.token
   2.2. The DUT responds with **GetOSDOptionsResponse** with parameters
      - OSDOptions =: *osdOptions*
3. ONVIF Client checks OSD Types features support as defined in Table 5.22.

Note: If the DUT does not return GetOSDOptionsResponse, ONVIF Client assumes the following features as undefined:

- Media2\OSD\Types\Text
- Media2\OSD\Types\Image

Table 5.22. Media2 Service – OSD Types Support (GetServices)

Criterion Item	GetOSDOptionsResponse	
Feature	**Supported**	**Not Supported**
Text	**Includes OSDOptions \Type = "Text" and MaximumNumberOfOSDs. @Total > 0**	**Does not include OSDOptions\Type = "Text" or MaximumNumberOfOSDs. @Total = 0**

Criterion Item	GetOSDOptionsResponse	
Feature	**Supported**	**Not Supported**
Image	**Includes OSDOptions \Type = "Image" and MaximumNumberOfOSDs. @Total > 0**	**Does not include OSDOptions\Type = "Image" or MaximumNumberOfOSDs. @Total = 0**

5.5.6.6 Media2 Events Support

Media2 Events support under Media2 Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that GetEventPropertiesResponse has already been retrieved via preceding procedure described in Section 5.5.7.1, otherwise all features defined in Table 5.23 will be marked as UNDEFINED.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.23.

Table 5.23. Media2 Events Support (GetServices)

Criterion Item	GetEventPropertiesResponse message	
Feature	**Supported**	**Not Supported**
Media/ProfileChanged	**Contains tns1:Media/ ProfileChanged Event topic**	**Does not contain tns1:Media/ ProfileChanged Event topic**
Media/ConfigurationChanged	**Contains tns1:Media/ ConfigurationChanged Event topic**	**Does not contain tns1:Media/ ConfigurationChanged Event topic**

5.5.7 Event Service

Event Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.24.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/events/wsdl" namespace ONVIF Client will use service with the latest version.

Note: If Event service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.7.1.
- Section 5.5.7.2.
- Section 5.5.7.3.
- Section 5.5.7.4.

Table 5.24. Event Service (GetServices)

Criterion Item	GetServicesResponse message	
Feature	**Supported**	**Not Supported**
Event Service	**Includes service with "http://www.onvif.org/ver10/events/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/events/wsdl" namespace**

5.5.7.1 Get Event Properties

The following discovery procedure will be performed for ONVIF Client to receive topic set and message content filter dialect list from the DUT. Received topic set will be used in event topics discovery procedures for different services. Received message content filter dialect list will be used in Section 5.5.7.3 and in Section 5.5.7.4.

Pre-requisite:

- This procedure assumes that Event Service is supported by the DUT as defined in Section 5.5.7, otherwise this procedure will be skipped.

Discovery Procedure:

1. ONVIF Client invokes **GetEventProperties** message.
2. The DUT responds with a **GetEventPropertiesResponse** message with parameters
   - TopicNamespaceLocation list
   - FixedTopicSet

- TopicSet =: *topicSet*
- TopicExpressionDialect list
- MessageContentFilterDialect list := *msgContentFilterDialectList*
- MessageContentSchemaLocation list

5.5.7.2 Event service features

Event capabilities support under Event Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Event Service is supported by the DUT as defined in Section 5.5.7, otherwise all features defined in Table 5.25 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client invokes GetServiceCapabilitiesRequest message for Event Service.
2. The DUT returns GetServiceCapabilitiesResponse
3. ONVIF Client checks features support as defined in Table 5.25.

Note: If the DUT does not return GetServiceCapabilitiesResponse then all features defined in Table 5.25 will be marked as UNDEFINED.

Note: Value of Capabilities.MaxPullPoints shall be saved to be used during Profiles support check.

Table 5.25. Event Service Features (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Persistent Notification Storage	**Capabilities. PersistentNotificationStorage = true**	**Skipped Capabilities. PersistentNotificationStorage or Capabilities. PersistentNotificationStorage = false**
WS Basic Notification	**Capabilities. MaxNotificationProducers > 0 or skipped Capabilities. MaxNotificationProducers**	**Capabilities. MaxNotificationProducers = 0**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
GetServiceCapabilities \MaxPullPoints capability	**Includes Capabilities.MaxPullPoints**	**Does not include Capabilities.MaxPullPoints**
Event Broker	**Includes Capabilities.MaxEventBrokers with value > 0**	**Does not include Capabilities.MaxEventBrokers OR MaxEventBrokers < = 0**
Protocols / mqtt	**Capabilities. EventBrokerProtocols list contains "mqtt" value**	**Capabilities. EventBrokerProtocols list does not contain "mqtt" value**
Protocols / mqtts	**Capabilities. EventBrokerProtocols list contains "mqtts" value**	**Capabilities. EventBrokerProtocols list does not contain "mqtts" value**
Protocols / ws	**Capabilities. EventBrokerProtocols list contains "ws" value**	**Capabilities. EventBrokerProtocols list does not contain "ws" value**
Protocols / wss	**Capabilities. EventBrokerProtocols list contains "wss" value**	**Capabilities. EventBrokerProtocols list does not contain "wss" value**

5.5.7.3 Event Service - Message Content Filter Support

Message Content Filter function support in Event Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that Event Service is supported by the DUT as defined in Section 5.5.7, otherwise all features defined in Table 5.26 will be marked as NOT SUPPORTED.
- This procedure assumes that GetEventPropertiesResponse has already been retrieved via preceding procedure described in Section 5.5.7.1, otherwise all features defined in Table 5.26 will be marked as UNDEFINED.

Discovery Procedure:

1. ONVIF Client checks Message Content Filter feature support as defined in Table 5.26.

Note: If Message Content Filter feature is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.7.4.

Table 5.26. Message Content Filter Feature (GetServices)

Criterion Item	GetEventPropertiesResponse message	
Feature	**Supported**	**Not Supported**
Message Content Filter	***msgContentFilterDialectList*** **contains at least one item with non empty value**	***msgContentFilterDialectList*** **does not contain at least one item with non empty value**

5.5.7.4 Event Service - ONVIF Message Content Filter Dialect Support

ONVIF Message Content Filter Dialect function support in Event Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that Event Service is supported by the DUT as defined in Section 5.5.7, otherwise all features defined in Table 5.26 will be marked as NOT SUPPORTED.
- This procedure assumes that GetEventPropertiesResponse has already been retrieved via preceding procedure described in Section 5.5.7.1, otherwise all features defined in Table 5.26 will be marked as UNDEFINED.
- This procedure assumes that DUT supports Message Content Filter feature according to Section 5.5.7.3, otherwise all features defined in Table 5.27 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client checks ONVIF Message Content Filter Dialect feature support as defined in Table 5.27.

Table 5.27. ONVIF Message Content Filter Dialect (GetServices)

Criterion Item	GetEventPropertiesResponse message	
Feature	**Supported**	**Not Supported**
ONVIF Message Content Filter Dialect	***msgContentFilterDialectList*** **contains item with value is equal to "http://www.onvif.org/ver10/tev/messageContentFilter/ItemFilter"**	***msgContentFilterDialectList*** **does not contain item with value is equal to "http://www.onvif.org/ver10/tev/messageContentFilter/ItemFilter"**

5.5.7.5 Event Service - Pull-Point Notification Support

Pull-Point Notification function support in Event Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that Event Service is supported by the DUT as defined in Section 5.5.7, otherwise all features defined in Table 5.26 will be marked as NOT SUPPORTED.
- This procedure assumes that GetServiceCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.5.7.2, otherwise all features defined in Table 5.28 will be marked as UNDEFINED.
- This procedure assumes that GetScopesResponse has already been retrieved via preceding procedure described in Section 5.7, otherwise all features defined in Table 5.28 will be marked as UNDEFINED.

Discovery Procedure:

1. ONVIF Client checks Pull-Point Notification feature support as defined in Table 5.28.

Table 5.28. Pull-Point Notification (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message and GetScopesResponse message	
Feature	**Supported**	**Not Supported**
Pull-Point Notification	**(Capabilities.MaxPullPoints skipped AND *scopesList* contains item with value is equal to "onvif://www.onvif.org/Profile/Streaming") OR Capabilities.MaxPullPoints > 0**	**Capabilities.MaxPullPoints = 0 OR (*scopesList* does not contain item with value is equal to "onvif://www.onvif.org/Profile/Streaming" AND Capabilities.MaxPullPoints skipped)**

5.5.8 Device IO Service

Device IO Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.29.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/deviceIO/wsdl" namespace ONVIF Client will use service with the latest version.

Note: If Device IO service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.8.1.
- Section 5.5.8.2.
- Section 5.5.8.3.

Table 5.29. Device IO Service (GetServices)

Criterion Item	GetServicesResponse message	
Feature	**Supported**	**Not Supported**
Device IO Service	**Includes service with "http://www.onvif.org/ver10/deviceIO/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/deviceIO/wsdl" namespace**

5.5.8.1 Device IO Features Support

Device IO capabilities support under Device IO Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Device IO Service is supported by the DUT as defined in Section 5.5.8, otherwise all features defined in Table 5.30 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client invokes **GetServiceCapabilities** request for Device IO Service.
2. The DUT responds with **GetServiceCapabilitiesResponse** message with parameters
   - Capabilities =: *cap*
3. ONVIF Client checks features support as defined in Table 5.30.

Note: If the DUT does not return GetServiceCapabilitiesResponse then all features defined in Table 5.30 will be marked as UNDEFINED, the following feature discovery procedures will be skipped and related features will be marked as UNDEFINED:

- Section 5.5.8.2.
- Section 5.5.8.3.

.

Note: If Relay Outputs feature is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.8.2.
- Section 5.5.8.3.

Table 5.30. Device IO Features Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Relay Outputs	***cap*.RelayOutputs > 0**	**Skipped *cap*.RelayOutputs or *cap*.RelayOutputs = 0**
Digital Inputs	***cap*.DigitalInputs > 0**	**Skipped *cap*.DigitalInputs or *cap*.DigitalInputs = 0**
Serial Ports	***cap*.SerialPorts > 0**	**Skipped *cap*.SerialPorts or *cap*.SerialPorts = 0**
Digital Input Options	***cap*.DigitalInputOptions = true**	**Skipped *cap*.DigitalInputOptions or *cap*.DigitalInputOptions = false**

5.5.8.2 Device IO Relay Output Options Support

Device IO Relay Output Options support in Device IO Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that Device IO Service is supported by the DUT as defined in Section 5.5.8, otherwise all features defined in Table 5.31 will be marked as NOT SUPPORTED.
- This procedure assumes that Relay Outputs is supported by the DUT as defined in Section 5.5.8.1, otherwise all features defined in Table 5.31 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client invokes **GetRelayOutputOptions** request with parameters
   - RelayOutputToken skipped
2. The DUT responds with **GetRelayOutputOptionsResponse** message or SOAP 1.2 fault.
3. ONVIF Client checks features support as defined in Table 5.31.

Note: If the DUT returns no response for GetRelayOutputOptions request, then all features defined in Table 5.31 will be marked as UNDEFUNED, the following feature discovery procedures will be skipped and related features will be marked as UNDEFUNED:

- Section 5.5.8.3

Note: If Relay Outputs Options feature is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.8.3

Table 5.31. Device IO Service – Relay Output Options Support (GetServices)

Criterion Item	GetRelayOutputOptionsResponse message	
Feature	**Supported**	**Not Supported**
Relay Output Options	**DUT returns GetRelayOutputOptionsResponse Response for step 2.**	**DUT returns any SOAP fault for step 2.**

5.5.8.3 Device IO Relay Outputs Features Support

Device IO Relay Output features support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that Device IO Service is supported by the DUT as defined in Section 5.5.8, otherwise all features defined in Table 5.32, Table 5.33, and Table 5.34 will be marked as NOT SUPPORTED.
- This procedure assumes that Relay Outputs is supported by the DUT as defined in Section 5.5.8.1, otherwise all features defined in Table 5.32, Table 5.33, and Table 5.34 will be marked as NOT SUPPORTED.
- This procedure assumes that Relay Output Options is supported by the DUT as defined in Section 5.5.8.2, otherwise all features defined in Table 5.32, Table 5.33, and Table 5.34 will be marked as NOT SUPPORTED.

- This procedure assumes that GetRelayOutputOptionsResponse has already been retrieved via preceding procedure described in Section 5.5.8.2, otherwise all features defined in Table 5.32, Table 5.33, and Table 5.34 will be marked as UNDEFINED.

Discovery Procedure:

1. ONVIF Client invokes **GetRelayOutputs** request to retrieve a list of all available relay outputs and their settings.
2. The DUT sends the **GetRelayOutputsResponse** message with parameters
   - RelayOutputs list =: *relayOutputsList*
3. For each Relay Output (*relayOutput*) from *relayOutputsList* do the following:
   3.1. ONVIF Client checks features support as defined in Table 5.32.
   3.2. If Relay Output supports Monostable Mode:
      3.2.1. ONVIF Client invokes **SetRelayOutputSettings** request with parameters
         - RelayOutput.@token := *relayOutput*.@token
         - RelayOutput.Properties.Mode := Monostable
         - RelayOutput.Properties.DelayTime := RelayOutputOptions[0].DelayTimes[0] from GetRelayOutputOptionsResponse, where RelayOutputOptions[0] is RelayOutputOptions with token = *relayOutput*.@token
         - RelayOutput.Properties.IdleState := closed
      3.2.2. The DUT responds with **SetRelayOutputSettingsResponse** message or SOAP 1.2 fault.
      3.2.3. ONVIF Client invokes **SetRelayOutputSettings** request with parameters
         - RelayOutput.@token := *relayOutput*.@token
         - RelayOutput.Properties.Mode := Monostable
         - RelayOutput.Properties.DelayTime := RelayOutputOptions[0].DelayTimes[0] from GetRelayOutputOptionsResponse, where RelayOutputOptions[0] is RelayOutputOptions with token = *relayOutput*.@token
         - RelayOutput.Properties.IdleState := open

3.2.4. The DUT responds with **SetRelayOutputSettingsResponse** message or SOAP 1.2 fault.

3.2.5. If for both steps and SOAP 1.2 fault was returned features listed in Table 5.34 will be marked as UNDEFINED for this Relay Output, otherwise ONVIF Client checks features support as defined in Table 5.33.

3.3. If Relay Output supports Bistable Mode:

3.3.1. ONVIF Client invokes **SetRelayOutputSettings** request with parameters

- RelayOutput.@token := *relayOutput*.@token
- RelayOutput.Properties.Mode := Bistable
- RelayOutput.Properties.DelayTime := *relayOutput*.Properties.DelayTime
- RelayOutput.Properties.IdleState := closed

3.3.2. The DUT responds with **SetRelayOutputSettingsResponse** message or SOAP 1.2 fault.

3.3.3. ONVIF Client invokes **SetRelayOutputSettings** request with parameters

- RelayOutput.@token := *relayOutput*.@token
- RelayOutput.Properties.Mode := Bistable
- RelayOutput.Properties.DelayTime := *relayOutput*.Properties.DelayTime
- RelayOutput.Properties.IdleState := open

3.3.4. The DUT responds with **SetRelayOutputSettingsResponse** message or SOAP 1.2 fault.

3.3.5. If for both steps and SOAP 1.2 fault was returned features listed in Table 5.34 will be marked as UNDEFINED for this Relay Output, otherwise ONVIF Client checks features support as defined in Table 5.34.

Note: If the DUT returns no RelayOutputOptions in GetRelayOutputOptionsResponse message, then all features defined in Table 5.32, Table 5.33, and Table 5.34 will be marked as UNDEFUNED.

Note: If the DUT returns no DelayTimes element in RelayOutputOptions with supporting of Monostable mode in GetRelayOutputOptionsResponse message, then all features defined in Table 5.33 will be marked as UNDEFUNED.

Note: If the DUT returns no response for SetRelayOutputSettings request, then all features defined in Table 5.33 and Table 5.34 will be marked as UNDEFUNED.

Table 5.32. Relay Outputs Features - Modes (GetServices)

Criterion Item	Current Relay Output from GetRelayOutputOptionsResponse message	
Feature (for each Relay Output)	**Supported**	**Not Supported**
Monostable Mode	**RelayOutputOptions.Mode contains Monostable**	**RelayOutputOptions.Mode does not contain Monostable**
Bistable Mode	**RelayOutputOptions.Mode contains Bistable**	**RelayOutputOptions.Mode does not contain Bistable**

Table 5.33. Relay Outputs Features - Idle States - Monostable (GetServices)

Criterion Item	Current Relay Output from GetRelayOutputOptionsResponse message	
Feature (for each Relay Output)	**Supported**	**Not Supported**
Monostable Mode \Closed Idle State	**DUT returns SetRelayOutputSettings Response at step 3.2.2**	**DUT returns SOAP fault at step 3.2.2**
Monostable Mode \Open Idle State	**DUT returns SetRelayOutputSettings Response at step 3.2.4**	**DUT returns SOAP fault at step 3.2.4**

Table 5.34. Relay Outputs Features - Idle States - Bistable (GetServices)

Criterion Item	Current Relay Output from GetRelayOutputOptionsResponse message	
Feature (for each Relay Output)	**Supported**	**Not Supported**
Bistable Mode\Closed Idle State	**DUT returns SetRelayOutputSettings Response at step 3.3.2**	**DUT returns SOAP fault at step 3.3.2**
Bistable Mode\Open Idle State	**DUT returns SetRelayOutputSettings Response at step 3.3.4**	**DUT returns SOAP fault at step 3.3.4**

5.5.9 PTZ Service Support

PTZ Service support is defined according to the following procedure.

Pre-requisite: This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure: ONVIF Client checks features support as defined in Table 5.35.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver20/ptz/wsdl" namespace, ONVIF Client will use service with the latest version.

Note: If the DUT does not support PTZ Service, all features from Section 5.5.8.1, Section 5.5.8.2, and Section 5.5.8.3 will be marked as unsupported. Procedure described in Section 5.5.8.1, Section 5.5.8.2, and Section 5.5.8.3 will be skipped.

Table 5.35. PTZ Service (GetServices)

Criterion Item	GetServicesResponse message	
Feature	**Supported**	**Not Supported**
PTZ Service	**Includes service with "http://www.onvif.org/ver20/ptz/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver20/ptz/wsdl" namespace**

5.5.9.1 PTZ Service Features Support

PTZ capabilities support under PTZ Service is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetServiceCapabilities request to retrieve PTZ Service capabilities.
2. The DUT returns GetServiceCapabilitiesResponse message with PTZ Service capabilities.
3. ONVIF Client checks features support as defined in Table 5.36.

Note: If the DUT returns no response for GetServiceCapabilities request, then all features defined in Table 5.36 will be marked as undefined.

Table 5.36. PTZ Service Features Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Get Compatible Configurations	**Capabilities. GetCompatibleConfigurations = true**	**Skipped Capabilities. GetCompatibleConfigurations or Capabilities.**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
		GetCompatibleConfigurations = false
Move Status	**Capabilities.MoveStatus = true**	**Skipped Capabilities.MoveStatus or Capabilities.MoveStatus = false**
Status Position	**Capabilities.StatusPosition = true**	**Skipped Capabilities.StatusPosition or Capabilities.StatusPosition = false**

5.5.9.2 PTZ Nodes Features Support

PTZ Nodes features support is defined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetNodes request to retrieve PTZ Nodes capabilities.
2. The DUT returns GetNodesResponse message with PTZ Nodes capabilities.
3. For each PTZ Node ONVIF Client checks features support as defined in Table 5.37.
4. For each PTZ Node with Home Position support ONVIF Client checks features support as defined in Section 5.5.8.3. For all others PTZ Nodes procedure described in Section 5.5.8.3 will be skipped and all features defined in Section 5.5.8.3 will be marked as unsupported.

Note: If the DUT returns no response for GetNodes request or the DUT returns GetNodesResponse message with empty PTZNode list, then all features defined in Table 5.37 and Section 5.5.8.3 will be marked as undefined.

Table 5.37. PTZ Nodes Features (GetServices)

Criterion Item	Current PTZNode from GetNodesResponse message	
Feature (for each PTZ Node)	**Supported**	**Not Supported**
Continuous Pan/Tilt movement	**SupportedPTZSpaces. ContinuousPanTiltVelocity Space element is present**	**SupportedPTZSpaces. ContinuousPanTiltVelocity Space element is not present**
Continuous Zoom movement	**SupportedPTZSpaces. ContinuousZoomVelocity Space element is present**	**SupportedPTZSpaces. ContinuousZoomVelocity Space element is not present**

Criterion Item	Current PTZNode from GetNodesResponse message	
Feature (for each PTZ Node)	**Supported**	**Not Supported**
Continuous movement	**Mandatory**	**-**
Absolute Pan/Tilt movement	**SupportedPTZSpaces. AbsolutePanTiltPositionSpace element is present**	**SupportedPTZSpaces. AbsolutePanTiltPositionSpace element is not present**
Absolute Zoom movement	**SupportedPTZSpaces. AbsoluteZoomPositionSpace element is present**	**SupportedPTZSpaces. AbsoluteZoomPositionSpace element is not present**
Absolute movement	**Absolute Pan/Tilt movement or Absolute Zoom movement is supported**	**Absolute Pan/Tilt movement and Absolute Zoom movement is not supported**
Relative Pan/Tilt movement	**SupportedPTZSpaces. RelativePanTiltTranlation Space element is present**	**SupportedPTZSpaces. RelativePanTiltTranlationSpace element is not present**
Relative Zoom movement	**SupportedPTZSpaces. RelativeZoomTranlationSpace element is present**	**SupportedPTZSpaces. RelativeZoomTranlationSpace element is not present**
Relative movement	**Relative Pan/Tilt movement or Relative Zoom movement is supported**	**Relative Pan/Tilt movement and Relative Zoom movement is not supported**
Speed configuration function	**Speed configuration function for Pan/Tilt movement or Speed configuration function for Zoom movement is supported**	**Speed configuration function for Pan/Tilt movement and Speed configuration function for Zoom movement is not supported**
Speed configuration function for Pan/Tilt movement	**SupportedPTZSpaces. PanTiltSpeedSpace element is present**	**SupportedPTZSpaces. PanTiltSpeedSpace element is not present**
Speed configuration function for Zoom movement	**SupportedPTZSpaces. ZoomSpeedSpace element is present**	**SupportedPTZSpaces. ZoomSpeedSpace element is not present**
Preset position	**MaximumNumberOfPresets > 0**	**MaximumNumberOfPresets = 0**
Auxiliary operation	**AuxiliaryCommands element is present**	**AuxiliaryCommands element is not present**
Home Position	**HomeSupported = true**	**HomeSupported = false**

5.5.9.3 Fixed/Configurable Home Position Support for PTZ Node

In case the PTZ Node supports Home Position function, the PTZ Node shall support either Fixed or Configurable Home Position. The following defines the discovery procedure to determine which Home Position function is supported by the PTZ Node.

Pre-requisite:

- This procedure assumes that PTZ Node was recieved in GetNodesResponse message via preceding procedure described in Section 5.5.8.2.

Discovery Procedure:

1. If FixedHomePosition attribute is defined in GetNodesResponse message for this PTZ Node, ONVIF Client checks features support as defined in Table 5.38 and skips other steps of this procedure.
2. ONVIF Client invokes GetConfigurations request to retrieve a PTZ Configurations list.
3. The DUT returns GetConfigurationsResponse message with the list of PTZConfiguration that contains PTZNode. ONVIF Client identifies first PTZConfiguration which has the corresponding current PTZ Node.
4. If DUT supports Media2 Service according to Section 5.5.6, ONVIF Client either selects or creates Media Profile anew along with the identified PTZConfiguration (refer to Annex A.8 for the details) and goes to the step 7.
5. If DUT supports Media Service according to Section 5.5.5, ONVIF Client either selects or creates Media Profile anew along with the identified PTZConfiguration (refer to Annex A.1 for the details) and goes to the step 7.
6. Otherwise Configurable Home Position and Fixed Home Position features will be marked as undefined.
7. ONVIF Client invokes SetHomePosition request (ProfileToken = selected or newly created profile token) message to check Configurable Home Position is supported by DUT.
8. ONVIF Client checks features support as defined in Table 5.39.
9. ONVIF Client restores Media Profiles setting in case it changes some of the Media Profiles configuration.

Table 5.38. Fixed/Configurable Home Position Support with FixedHomePosition Attribute (GetServices)

Criterion Item	Current PTZNode from GetNodesResponse message	
Feature (for each PTZ Node with Home Position support)	**Supported**	**Not Supported**
Configurable Home Position	**FixedHomePosition = false**	**FixedHomePosition = true**
Fixed Home Position	**FixedHomePosition = true**	**FixedHomePosition = false**

Table 5.39. Fixed/Configurable Home Position Support without FixedHomePosition Attribute (GetServices)

Criterion Item	SetHomePositionResponse	
Feature (for each PTZ Node with Home Position support)	**Supported**	**Not Supported**
Configurable Home Position	**DUT returns SetHomePositionResponse**	**DUT returns SOAP fault**
Fixed Home Position	**DUT returns SOAP fault**	**DUT returns SetHomePositionResponse**

5.5.10 Imaging Service Support

Imaging Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.40.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver20/imaging/wsdl" namespace ONVIF Client will use service with the latest version.

Note: If Imaging service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.10.1.
- Section 5.5.10.3.

Table 5.40. Imaging Service (GetServices)

Criterion Item	GetServicesResponse message	
Feature	**Supported**	**Not Supported**
Imaging Service	**Includes service with "http://www.onvif.org/ver20/imaging/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver20/imaging/wsdl" namespace**

5.5.10.1 IrCutfilterConfiguration Feature Support

IrCutfilterConfiguration function support in Imaging Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that Imaging Service is supported by the DUT as defined in Section 5.5.10, otherwise all features defined in Table 5.41 will be marked as NOT SUPPORTED.
- This procedure assumes that Device IO Service is supported by the DUT as defined in Section 5.5.8 or Media Service is supported by the DUT as defined in Section 5.5.5, otherwise all features defined in Table 5.41 will be marked as UNDEFUNED.

Discovery Procedure:

1. ONVIF Client retrieves a list of Video Sources by following the procedure mentioned in Annex A.6 with the following input and output parameters
   - out *videoSorceTokenList* - a list of Video Source tokens
2. For each Video Source token *videoSorceToken* in *videoSorceTokenList*
   2.1. ONVIF Client invokes **GetOptions** with parameters
      - VideoSourceToken =: *videoSorceToken*

   2.2. The DUT responds with **env:Receiver/ter:ActionNotSupported/ter:NoImagingForSource** SOAP 1.2 fault or with **GetOptionsResponse** with parameters
      - ImagingOptions =: *imagingOptions*

   2.3. If *imagingOptions*.IrCutFilterModes list contains at least two items and one of them is equal to OFF, skip other steps.
3. ONVIF Client checks features support as defined in Table 5.41.

Note: If the DUT does not return **env:Receiver/ter:ActionNotSupported/ ter:NoImagingForSource** SOAP 1.2 fault or **GetMoveOptionsResponse**, then all features defined in Table 5.41 will be marked as UNDEFUNED.

Note: If the DUT does not return *videoSorceTokenList* list or *videoSorceTokenList* list is empty, then all features defined in Table 5.41 will be marked as UNDEFUNED.

Table 5.41. IrCutfilter Configuration Function Support in Imaging Service (GetServices)

Criterion Item	GetOptionsResponse message	
Feature	**Supported**	**Not Supported**
IrCutfilter Configuration	**For at least one GetOptionsResponse message at least two IrCutFilterModes elements are present in GetOptionsResponse. ImagingOptions and one of them equal to OFF**	**There are no GetOptionsResponse messages with at least two IrCutFilterModes elements in GetOptionsResponse. ImagingOptions with one of them equal to OFF**

5.5.10.2 Imaging Events Support

Imaging Events support under Imaging Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetEventPropertiesResponse has already been retrieved via preceding procedure described in Section 5.5.7.1, otherwise all features defined in Table 5.42 will be marked as UNDEFINED.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.42.

Table 5.42. Imaging Events Support (GetServices)

Criterion Item	GetEventPropertiesResponse	
Feature	**Supported**	**Not Supported**
Image Too Blurry	**Contains tns1:VideoSource/ ImageTooBlurry/ ImagingService or tns1:VideoSource/ ImageTooBlurry/**	**Does not contain tns1:VideoSource/ ImageTooBlurry/ ImagingService and tns1:VideoSource/**

Criterion Item	GetEventPropertiesResponse	
Feature	**Supported**	**Not Supported**
	AnalyticsService or tns1:VideoSource/ ImageTooBlurry/ RecordingService Event topic	**ImageTooBlurry/ AnalyticsService and tns1:VideoSource/ ImageTooBlurry/ RecordingService Event topic**
Image Too Dark	**Contains tns1:VideoSource/ ImageTooDark/ ImagingService or tns1:VideoSource/ ImageTooDark/ AnalyticsService or tns1:VideoSource/ ImageTooDark/ RecordingService Event topic**	**Does not contain tns1:VideoSource/ ImageTooDark/ ImagingService and tns1:VideoSource/ ImageTooDark/ AnalyticsService and tns1:VideoSource/ ImageTooDark/ RecordingService Event topic**
Image Too Bright	**Contains tns1:VideoSource/ ImageTooBright/ ImagingService or tns1:VideoSource/ ImageTooBright/ AnalyticsService or tns1:VideoSource/ ImageTooBright/ RecordingService Event topic**	**Does not contain tns1:VideoSource/ ImageTooBright/ ImagingService and tns1:VideoSource/ ImageTooBright/ AnalyticsService and tns1:VideoSource/ ImageTooBright/ RecordingService Event topic**
Global Scene Change	**Contains tns1:VideoSource/ GlobalSceneChange/ ImagingService or tns1:VideoSource/ GlobalSceneChange/ AnalyticsService or tns1:VideoSource/ GlobalSceneChange/ RecordingService Event topic**	**Does not contain tns1:VideoSource/ GlobalSceneChange/ ImagingService and tns1:VideoSource/ GlobalSceneChange/ AnalyticsService and tns1:VideoSource/ GlobalSceneChange/ RecordingService Event topic**
Motion Alarm	**Contains tns1:VideoSource/ MotionAlarm event topic**	**Does not contain tns1:VideoSource/ MotionAlarm event topic**

5.5.10.3 Focus Control Function Support

Focus Control function support in Imaging Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that Imaging Service is supported by the DUT as defined in Section 5.5.10, otherwise all features defined in Table 5.43 will be marked as NOT SUPPORTED.
- This procedure assumes that Device IO Service is supported by the DUT as defined in Section 5.5.8 or Media Service is supported by the DUT as defined in Section 5.5.5, otherwise all features defined in Table 5.43 will be marked as UNDEFUNED.

Discovery Procedure:

1. ONVIF Client retrieves a list of Video Sources by following the procedure mentioned in Annex A.6 with the following input and output parameters
   - out *videoSorceTokenList* - a list of Video Source tokens
2. For each Video Source token *videoSorceToken* in *videoSorceTokenList*
   2.1. ONVIF Client invokes **GetMoveOptions** with parameters
      - VideoSourceToken =: *videoSorceToken*

   2.2. The DUT responds with **env:Receiver/ter:ActionNotSupported/ter:NoImagingForSource** SOAP 1.2 fault or with **GetMoveOptionsResponse** with parameters
      - MoveOptions =: *moveOptions*

   2.3. If *moveOptions* contains **MoveOptions/Absolute** or **MoveOptions/Relative** or **MoveOptions/Continuous**, skip other steps.
3. ONVIF Client checks features support as defined in Table 5.43.

Note: If the DUT does not return **env:Receiver/ter:ActionNotSupported/ter:NoImagingForSource** SOAP 1.2 fault or **GetMoveOptionsResponse**, then all features defined in Table 5.43 will be marked as UNDEFUNED.

Note: If the DUT does not return *videoSorceTokenList* list or *videoSorceTokenList* list is empty, then all features defined in Table 5.43 will be marked as UNDEFUNED.

Table 5.43. Focus Control (GetServices)

Criterion Item	GetMoveOptionsResponse	
Feature	**Supported**	**Not Supported**
Focus Control	**Contains MoveOptions \Absolute or MoveOptions\Relative or MoveOptions\Continuous**	**Does not contain MoveOptions\Absolute and MoveOptions\Relative and MoveOptions\Continuous**

5.5.11 Analytics Service Support

Analytics Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.44.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver20/analytics/wsdl" namespace ONVIF Client will use service with the latest version.

Note: If Analytics Service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.11.1.
- Section 5.5.11.2.
- Section 5.5.11.3.
- Section 5.5.11.3.

Table 5.44. Analytics Service (GetServices)

Criterion Item	GetServicesResponse	
Feature	**Supported**	**Not Supported**
Analytics Service	**Includes service with "http://www.onvif.org/ver20/analytics/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver20/analytics/wsdl" namespace**

5.5.11.1 Analytics Features Support

Rule Engine support under Analytics Service is determined according to the following procedure in conjunction with the above procedure.

Discovery Procedure:

1. ONVIF Client invokes GetServiceCapabilitiesRequest message for Analytics Service capabilities support by the DUT.
2. The DUT returns GetServiceCapabilitiesResponse. ONVIF Client checks features support as defined in Table 5.45.

Note: If the DUT does not return GetServiceCapabilitiesResponse then Rule Engine feature will be marked as undefined.

Table 5.45. Analytics Features Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Rule Engine	**Capabilities.RuleSupport = true**	**Skipped Capabilities.RuleSupport or Capabilities.RuleSupport = false**
Rule Options	**RuleOptionsSupported = true**	**Skipped Capabilities.RuleOptionsSupported or Capabilities.RuleOptionsSupported = false**
Analytics Modules	**Capabilities.AnalyticsModuleSupport = true**	**Skipped Capabilities.AnalyticsModuleSupport or Capabilities.AnalyticsModuleSupport = false**
Analytics Module Options	**Capabilities.AnalyticsModuleOptionsSupported = true**	**Skipped Capabilities.AnalyticsModuleOptionsSupported or Capabilities.AnalyticsModuleOptionsSupported = false**
Supported Metadata	**Capabilities.SupportedMetadata = true**	**Skipped Capabilities.SupportedMetadata or**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
		Capabilities.SupportedMetadata = false
Image Sending	**Capabilities contains non-empty @ImageSendingType**	**Skipped Capabilities.@ImageSendingType or @ImageSendingType has empty list value**
Embedded Image Sending Type	**Capabilities contains @ImageSendingType with "Embedded" value in the list**	**Skipped Capabilities.@ImageSendingType or @ImageSendingType does not contain "Embedded" value**
Local Storage Image Sending Type	**Capabilities contains @ImageSendingType with "LocalStorage" value in the list**	**Skipped Capabilities.ImageSendingType or @ImageSendingType does not contain "LocalStorage" value**
Remote Storage Image Sending Type	**Capabilities contains @ImageSendingType with "RemoteStorage" value in the list**	**Skipped Capabilities.ImageSendingType or @ImageSendingType does not contain "RemoteStorage" value**

5.5.11.2 Supported Rules

Supported Rules support under Analytics Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- DUT supports **Rule Engine** feature according to Table 5.45, otherwise all features defined in Table 5.46 will be marked as NOT SUPPORTED.
- DUT supports **Media2 Service** feature according to Section 5.5.6, otherwise all features defined in Table 5.46 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client retrieves a list of Analytics Configurations by following the procedure mentioned in Annex A.5 with the following input and output parameters
   - out *analyticsConfList* - a list of Analytics configurations

2. For each Analytics Configuration *analyticsConf* in *analyticsConfList* repeat the following steps:

   2.1. ONVIF Client invokes **GetSupportedRules** request with parameters

   - ConfigurationToken := *analyticsConf*.token

   2.2. DUT responds with **GetSupportedRulesResponse** message with parameters

   - SupportedRules =: *supportedRules1*

   2.3. *supportedRules* := *supportedRules* + *supportedRules1*

3. ONVIF Client checks features support as defined in Table 5.46.

Note: If the DUT does not return GetSupportedRulesResponse then all features defined in Table 5.46 will be marked as UNDEFINED.

Table 5.46. Supported Rules (GetServices)

Criterion Item	GetSupportedRulesResponse	
Feature	**Supported**	**Not Supported**
Motion Region Detector Rule	**Contains RuleDescription element with Name value is equal to tt:MotionRegionDetector**	**Does not contain RuleDescription element with Name value is equal to tt:MotionRegionDetector**
Face Recognition Rule	**Contains RuleDescription element with Name value is equal to tt:FaceRecognition**	**Does not contain RuleDescription element with Name value is equal to tt:FaceRecognition**
License Plate Recognition Rule	**Contains RuleDescription element with Name value is equal to tt:LicensePlateRecognition**	**Does not contain RuleDescription element with Name value is equal to tt:LicensePlateRecognition**

5.5.11.3 Analytics Service - Supported Metadata

Supported Metadata supporting under Analytics Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Analytics Service is supported by the DUT as defined in Section 5.5.11, otherwise all features defined in Table 5.47 will be marked as NOT SUPPORTED.

- This procedure assumes that Metadata Supported is supported by the DUT as defined in Table 5.45, otherwise all features defined in Table 5.47 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client invokes **GetSupportedMetadata** request with parameters
   - Type skipped
2. DUT responds with **GetSupportedMetadataResponse** message with parameters
   - AnalyticsModule list =: *analyticsModuleList*

Note: If the DUT does not return GetSupportedMetadataResponse then the following features will be marked as undefined:

- Object Classification
- Vehicle Info
- Geo Location
- Human Face
- Human Body
- License Plate Info

Table 5.47. Supported Metadata (GetServices)

Criterion Item	GetSupportedMetadataResponse	
Feature	**Supported**	**Not Supported**
Object Classification	**Contains at least one AnalyticsModule element with SampleFrame.Object. Appearance.Class.Type**	**Does not contain AnalyticsModule element with SampleFrame.Object. Appearance.Class.Type**
Vehicle Info	**Contains at least one AnalyticsModule.SampleFrame.Object.Appearance with VehicleInfo element**	**Does not contain AnalyticsModule.SampleFrame.Object.Appearance with VehicleInfo element**
Human Face	**Contains at least one AnalyticsModule.SampleFrame.Object.Appearance with HumanFace element**	**Does not contain AnalyticsModule.SampleFrame.Object.Appearance with HumanFace element**
Human Body	**Contains at least one AnalyticsModule.SampleFrame.Object.Appearance with HumanBody element**	**Does not contain AnalyticsModule.SampleFrame.Object.Appearance with HumanBody element**

Criterion Item	GetSupportedMetadataResponse	
Feature	Supported	Not Supported
License Plate Info	**Contains at least one AnalyticsModule.SampleFrame.Object.Appearance with LicensePlateInfo element**	**Does not contain AnalyticsModule.SampleFrame.Object.Appearance with LicensePlateInfo element**
Geo Location	**Contains at least one AnalyticsModule element with SampleFrame.Object. Appearance.GeoLocation element**	**Does not contain AnalyticsModule element with SampleFrame.Object. Appearance with GeoLocation element**

5.5.12 Recording Control Service Support

Recording Control Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.48.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/recording/wsdl" namespace ONVIF Client will use service with the latest version.

Note: If Recording Control service is not supported, the following feature discovery (Dynamic Recordings, Dynamic Tracks and Audio Recording features support) will be skipped.

Table 5.48. Recording Control Service (GetServices)

Criterion Item	GetServicesResponse	
Feature	Supported	Not Supported
Recording Control Service	**Includes service with "http://www.onvif.org/ver10/recording/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/recording/wsdl" namespace**

Recording Control features support

Dynamic Recordings, Dynamic Tracks, Audio Recording, and Recording Options support under Recording Control Service is determined according to the following procedure in conjunction with the above procedure.

Discovery Procedure:

1. ONVIF Client invokes GetServiceCapabilitiesRequest message to check Dynamic Recording capability support by the DUT.
2. The DUT returns GetServiceCapabilitiesResponse. ONVIF Client checks features support as defined in Table 5.49.

Note: If the DUT does not return GetServiceCapabilitiesResponse then Dynamic Recordings feature, Dynamic Tracks feature, Audio Recording feature, Recording Options, Metadata Recording feature, JPEG, H.264, and MPEG4 will be marked as undefined.

Table 5.49. Recording Control Features Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Dynamic Recordings	**Capabilities. DynamicRecordings = true**	**Skipped Capabilities. DynamicRecordings or Capabilities. DynamicRecordings = false**
Dynamic Tracks	**Capabilities.DynamicTracks = true**	**Skipped Capabilities. DynamicTracks or Capabilities. DynamicTracks = false**
Audio Recording	**Includes Capabilities.Encoding with at least one audio codec (AAC, G711, or G726)**	**Does not include Capabilities.Encoding with at least one audio codec (AAC, G711, or G726)**
Recording Options	**Capabilities.Options = true**	**Skipped Capabilities.Options or Capabilities.Options = false**
tns1:RecordingConfig/ DeleteTrackData	**GetEventProperties contains tns1:RecordingConfig/ DeleteTrackData topic**	**GetEventProperties doesn't contain tns1:RecordingConfig/ DeleteTrackData topic**
Metadata Recording	**Capabilities. MetadataRecording = true**	**Skipped Capabilities. MetadataRecording or Capabilities. MetadataRecording = false**
JPEG	**Includes Capabilities.Encoding with JPEG**	**Does not include Capabilities.Encoding with JPEG**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
H.264	**Includes Capabilities.Encoding with H264**	**Does not include Capabilities.Encoding with H264**
MPEG4	**Includes Capabilities.Encoding with MPEG4**	**Does not include Capabilities.Encoding with MPEG4**

5.5.13 Recording Search Service Support

Recording Search Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.50.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/search/wsdl" namespace ONVIF Client will use service with the latest version.

Note: If Recording Search service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.13.1.
- Section 5.5.13.2.

Table 5.50. Recording Search Service (GetServices)

Criterion Item	GetServicesResponse message	
Feature	**Supported**	**Not Supported**
Recording Search Service	**Includes service with "http://www.onvif.org/ver10/search/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/search/wsdl" namespace**

5.5.13.1 Metadata Search Support

Metadata Search support under Recording Search Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Recording Search Service Service is supported by the DUT as defined in Section 5.5.13, otherwise all features defined in Table 5.51 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client invokes **GetServiceCapabilities** request for Recording Search Service.
2. The DUT responds with **GetServiceCapabilitiesResponse** message with parameters
   - Capabilities =: *cap*
3. ONVIF Client checks features support as defined in Table 5.51.

Note: If the DUT does not return GetServiceCapabilitiesResponse then all features defined in Table 5.51 will be marked as UNDEFINED.

Table 5.51. Metadata Search Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Metadata Search	***cap*.MetadataSearch = true**	**Skipped *cap*.MetadataSearch or *cap*.MetadataSearch = false**

5.5.13.2 PTZ Position Search Support

PTZ Position Search support under Recording Search Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Recording Search Service Service is supported by the DUT as defined in Section 5.5.13, otherwise all features defined in Table 5.52 will be marked as NOT SUPPORTED.
- This procedure assumes that all pre-requisite defined in Annex A.10 are fulfilled, otherwise states of the features listed in Table 5.52 could not be defined correctly.

Discovery Procedure:

1. ONVIF Client invokes **GetRecordingInformation** requiest with parameters

- RecordingToken := *recordingToken*

2. The DUT responds with **GetRecordingInformationResponse** message with parameters
   - RecordingInformation =: *recordingInformation*
3. ONVIF Client invokes **FindPTZPosition** requiest with parameters
   - If *recordingInformation*.EarliestRecording is specified:
     - StartPoint := *recordingInformation*.EarliestRecording
   - otherwise:
     - StartPoint := minimal value of DataFrom element in *recordingInformation*.Track list
   - If *recordingInformation*.EarliestRecording is specified:
     - EndPoint := *recordingInformation*.LatestRecording
   - otherwise:
     - EndPoint is skipped
   - Scope is empty element
   - SearchFilter.MinPosition.PanTilt.x := -1
   - SearchFilter.MinPosition.PanTilt.y := -1
   - SearchFilter.MinPosition.Zoom is skipped
   - SearchFilter.MaxPosition.PanTilt.x := 1
   - SearchFilter.MaxPosition.PanTilt.y := 1
   - SearchFilter.MaxPosition.Zoom is skipped
   - SearchFilter.EnterOrExit := false
   - MaxMatches is skipped
   - KeepAliveTime := "PT3S"
4. The DUT responds with SOAP 1.2 fault or **FindPTZPositionResponse** message with parameters
   - SearchToken =: *searchToken*

5. ONVIF Client checks features support as defined in Table 5.52.
6. If the DUT returns **FindPTZPositionResponse** message:
   - ONVIF Client invokes **EndSearch** requiest with parameters
     - SearchToken := *searchToken*
   - The DUT responds with **EndSearchResponse** message with parameters
     - Endpoint

Note: *recordingToken* will be taken from 'Recording from tests' field of ONVIF Device Test Tool.

Note: If the DUT does not return GetRecordingInformationResponse then all features defined in Table 5.52 will be marked as UNDEFINED.

Note: If the DUT does not return *recordingInformation*.EarliestRecording and there are no *recordingInformation*.Track items then all features defined in Table 5.52 will be marked as UNDEFINED.

Table 5.52. PTZ Position Search Support (GetServices)

Criterion Item	FindPTZPositionResponse message	
Feature	**Supported**	**Not Supported**
PTZ Position Search	**DUT returns FindPTZPositionResponse**	**DUT returns SOAP 1.2 fault**

5.5.14 Replay Service Support

Replay Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.53.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/replay/wsdl" namespace, ONVIF Client will use service with the latest version.

Note: If Replay service is not supported, the following feature discovery (Reverse Replay features support) will be skipped.

Table 5.53. Replay Service (GetServices)

Criterion Item	GetServicesResponse	
Feature	**Supported**	**Not Supported**
Replay Service	**Includes service with "http://www.onvif.org/ver10/replay/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/replay/wsdl" namespace**

Replay features support

Reverse Replay and RTP/RTSP/TCP support under Replay Service is determined according to the following procedure in conjunction with the above procedure.

Discovery Procedure:

1. ONVIF Client invokes GetServiceCapabilitiesRequest message to check Reverse Replay capability support by DUT.
2. The DUT returns GetServiceCapabilitiesResponse. ONVIF Client checks features support as defined in Table 5.54.

Note: If the DUT does not return GetServiceCapabilitiesResponse then Reverse Replay feature and RTP/RTSP/TCP feature will be marked as undefined.

Table 5.54. Replay Control Features Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Reverse Replay	**Capabilities.ReversePlayback = true**	**Skipped Capabilities.ReversePlayback or Capabilities.ReversePlayback = false**
RTP/RTSP/TCP	**Capabilities.RTP_RTSP_TCP = true**	**Skipped Capabilities.RTP_RTSP_TCP or Capabilities.RTP_RTSP_TCP = false**

5.5.15 Receiver Service Support

Receiver Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.55.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/receiver/wsdl" namespace, ONVIF Client will use service with the latest version.

Table 5.55. Receiver Service (GetServices)

Criterion Item	GetServicesResponse	
Feature	**Supported**	**Not Supported**
Receiver Service	**Includes service with "http://www.onvif.org/ver10/receiver/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/receiver/wsdl" namespace**

5.5.16 Door Control Service Support

Door Control Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.56.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/doorcontrol/wsdl" namespace, ONVIF Client will use service with the latest version.

Note: If Door Control service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.16.1
- Section 5.5.16.2
- Section 5.5.16.3

Table 5.56. Door Control Service (GetServices)

Criterion Item	GetServicesResponse	
Feature	**Supported**	**Not Supported**
Door Control Service	**Includes service with "http://www.onvif.org/ver10/doorcontrol/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/doorcontrol/wsdl" namespace**

5.5.16.1 Door Control Features Support

Door Control capabilities support under Door Control Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Door Control Service is supported by the DUT as defined in Section 5.5.16, otherwise all features defined in Table 5.57 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client invokes **GetServiceCapabilities** request for Door Control Service.
2. The DUT responds with **GetServiceCapabilitiesResponse** message with parameters
   - Capabilities =: *cap*
3. ONVIF Client checks features support as defined in Table 5.57.

Note: If the DUT does not return GetServiceCapabilitiesResponse, then all features defined in Table 5.57 will be marked as UNDEFINED.

Table 5.57. Door Control Features Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Door Entity	***cap*. MaxDoors > 0 or *cap*. MaxDoors is skipped**	***cap*. MaxDoors = 0**
Door Management	***cap*. DoorManagementSupported = true**	**Skipped *cap*. DoorManagementSupported or *cap*.**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
		DoorManagementSupported = false
Client Supplied Token	***cap.* ClientSuppliedTokenSupported = true**	**Skipped *cap.* ClientSuppliedTokenSupported or *cap.* ClientSuppliedTokenSupported = false**

5.5.16.2 Door Entity Features Support

Door Entity Features support under Door Control Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Door Control Service is supported by the DUT as defined in Section 5.5.16, otherwise all features defined in Table 5.58 will be marked as NOT SUPPORTED.
- This procedure assumes that Door Entity is supported by the DUT as defined in Section 5.5.16.1, otherwise all features defined in Table 5.58 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client retrieves a list of Doors with Door Info by following the procedure mentioned in Annex A.2 with the following input and output parameters
   - out *doorInfoList* - Door Info list
2. ONVIF Client checks features support as defined in Table 5.58.

Note: If the DUT does not return GetDoorInfoListResponse message, then all features defined in Table 5.58 will be marked as UNDEFINED.

Note: Lock Down Door feature support means that both LockDownDoor and LockDownReleaseDoor command are expected to be supported by the DUT.

Note: Lock Open Door feature support means that both LockOpenDoor and LockOpenReleaseDoor command are expected to be supported by the DUT.

Table 5.58. Door Entity Support (GetServices)

Criterion Item	All DoorInfos	
Feature	**Supported**	**Not Supported**
Access Door	**Contains at least one Door with Capabilities.Access = true**	**Contains no Doors with Capabilities.Access = true**
Lock Door	**Contains at least one Door with Capabilities.Lock = true**	**Contains no Doors with Capabilities.Lock = true**
Unlock Door	**Contains at least one Door with Capabilities.Unlock = true**	**Contains no Doors with Capabilities.Unlock = true**
Double Lock Door	**Contains at least one Door with Capabilities.DoubleLock = true**	**Contains no Doors with Capabilities.DoubleLock = true**
Block Door	**Contains at least one Door with Capabilities.Block = true**	**Contains no Doors with Capabilities.Block = true**
Lock Down Door	**Contains at least one Door with Capabilities.LockDown = true**	**Contains no Doors with Capabilities.LockDown = true**
Lock Open Door	**Contains at least one Door with Capabilities.LockOpen = true**	**Contains no Doors with Capabilities.LockOpen = true**
Door Monitor	**Contains at least one Door with Capabilities.DoorMonitor = true**	**Contains no Doors with Capabilities.DoorMonitor = true**
Lock Monitor	**Contains at least one Door with Capabilities.LockMonitor = true**	**Contains no Doors with Capabilities.LockMonitor = true**
Double Lock Monitor	**Contains at least one Door with Capabilities. DoubleLockMonitor = true**	**Contains no Doors with Capabilities. DoubleLockMonitor = true**
Alarm	**Contains at least one Door with Capabilities.Alarm = true**	**Contains no Doors with Capabilities. Alarm = true**
Tamper	**Contains at least one Door with Capabilities.Tamper = true**	**Contains no Doors with Capabilities. Tamper = true**

Criterion Item	All DoorInfos	
Feature	**Supported**	**Not Supported**
Fault	**Contains at least one Door with Capabilities.Fault = true**	**Contains no Doors with Capabilities. Fault = true**

5.5.16.3 Door Control Events Support

Door Control Events support under Door Control Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Door Control Service is supported by the DUT as defined in Section 5.5.16, otherwise all features defined in Table 5.59 will be marked as NOT SUPPORTED.
- This procedure assumes that Door Entity is supported by the DUT as defined in Section 5.5.16.1, otherwise all features defined in Table 5.59 will be marked as NOT SUPPORTED.
- This procedure assumes that GetEventPropertiesResponse has already been retrieved via preceding procedure described in Section 5.5.7.1, otherwise all features defined in Table 5.59 will be marked as UNDEFINED.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.59.

Table 5.59. Door Control Events Support (GetServices)

Criterion Item	GetEventPropertiesResponse	
Feature	**Supported**	**Not Supported**
Door/State/DoorMode	**Contains tns1:Door/State/ DoorMode Event topic**	**Does not contain tns1:Door/ State/DoorMode Event topic**
Door/State/DoorPhysicalState	**Contains tns1:Door/ State/ DoorPhysicalState Event topic**	**Does not contain tns1:Door/ State/ DoorPhysicalState Event topic**
Door/State/LockPhysicalState	**Contains tns1:Door/ State/ LockPhysicalState Event topic**	**Does not contain tns1:Door/ State/ LockPhysicalState Event topic**
Door/State/ DoubleLockPhysicalState	**Contains tns1:Door/State/ DoubleLockPhysicalState Event topic**	**Does not contain tns1:Door/State/**

Criterion Item	GetEventPropertiesResponse	
Feature	**Supported**	**Not Supported**
		DoubleLockPhysicalState Event topic
Door/State/DoorAlarm	**Contains tns1:Door/State/DoorAlarm Event topic**	**Does not contain tns1:Door/State/DoorAlarm Event topic**
Door/State/DoorTamper	**Contains tns1:Door/State/DoorTamper Event topic**	**Does not contain tns1:Door/State/DoorTamper Event topic**
Door/State/DoorFault	**Contains tns1:Door/State/DoorFault Event topic**	**Does not contain tns1:Door/State/DoorFault Event topic**
Configuration/Door/Changed	**Contains tns1:Configuration/Door/Changed Event topic**	**Does not contain tns1:Configuration/Door/Changed Event topic**
Configuration/Door/Removed	**Contains tns1:Configuration/Door/Removed Event topic**	**Does not contain tns1:Configuration/Door/Removed Event topic**

5.5.17 Access Control Service Support

Access Control Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.60.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/accesscontrol/wsdl" namespace, ONVIF Client will use service with the latest version.

Note: If Access Control service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.17.1.
- Section 5.5.17.2.

Table 5.60. Access Control Service Support (GetServices)

Criterion Item	GetServicesResponse message	
Feature	**Supported**	**Not Supported**
Access Control Service	**Includes service with "http://www.onvif.org/ver10/accesscontrol/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/accesscontrol/wsdl" namespace**

5.5.17.1 Area Entity Support

Area Entity support under Access Control Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Access Control Service is supported by the DUT as defined in Section 5.5.17, otherwise all features defined in Table 5.61 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client retrieves a complete list of area info items by following the procedure mentioned in Annex A.4 with the following input and output parameters
   - out *areaInfoCompleteList* - complete areas info list
   - out *areasNumber* - areas number
2. ONVIF Client checks features support as defined in Table 5.61.

Note: If DUT fails procedure described in Annex A.4, all features from Table 5.61 will be marked as UNDEFINED.

Table 5.61. Area Entity Support (GetServices)

Criterion Item	GetAreaInfoListResponse messages	
Feature	**Supported**	**Not Supported**
Area Entity	***areasNumber* > 0**	***areasNumber* = 0**

5.5.17.2 Access Point Entity Support and Access Point Features Support

Access Point Entity support and sub-features support under Access Control Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Access Control Service is supported by the DUT as defined in Section 5.5.17, otherwise all features defined in Table 5.62 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. Access Point Entity shall be defined as supported as it is a mandatory feature to be supported by DUT.
2. ONVIF Client retrieves a complete list of access point info items by following the procedure mentioned in Annex A.3 with the following input and output parameters
   - out *accessPointInfoCompleteList* - complete access points info list
   - out *accessPointsNumber* - access points number
3. ONVIF Client checks features support as defined in Table 5.62.

Note: If DUT fails procedure described in Annex A.3, all features from Table 5.62 except Access Point Entity will be marked as UNDEFINED.

Table 5.62. Access Control Entity Support (GetServices)

Criterion Item	All AccessPointInfos from *accessPointInfoCompleteList*	
Feature	**Supported**	**Not Supported**
Access Point Entity	**Mandatory**	**-**
Enable/Disable Access Point	**Contains at least one AccessPointInfo with Capabilities. DisableAccessPoint = true**	**Contains no AccessPointInfo with Capabilities. DisableAccessPoint = true**
Duress	**Contains at least one AccessPointInfo with Capabilities.Duress = true**	**Contains no AccessPointInfo with Capabilities.Duress = true**
Access Taken	**Contains at least one AccessPointInfo with**	**Contains no AccessPointInfo with**

Criterion Item	All AccessPointInfos from *accessPointInfoCompleteList*	
Feature	**Supported**	**Not Supported**
	Capabilities.AccessTaken = true	**Capabilities.AccessTaken = true**
External Authorization	**Contains at least one AccessPointInfo with Capabilities. ExternalAuthorization = true**	**Contains no AccessPointInfo with Capabilities. ExternalAuthorization = true**
Anonymous Access	**Contains at least one AccessPointInfo with Capabilities. AnonymousAccess = true**	**Contains no AccessPointInfo with Capabilities. AnonymousAccess = true**
Identifier Access	**Contains at least one AccessPointInfo with Capabilities. IdentifierAccess = true**	**Contains no AccessPointInfo with Capabilities. IdentifierAccess = true**

5.5.17.3 Access Control Events Support

Access Control Events support under Access Control Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that GetEventPropertiesResponse has already been retrieved via preceding procedure described in Section 5.5.7.1, otherwise all features defined in Table 5.63 will be marked as UNDEFINED.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.63.

Table 5.63. Access Control Events Support (GetServices)

Criterion Item	GetEventPropertiesResponse message	
Feature	**Supported**	**Not Supported**
AccessControl/ AccessGranted/Anonymous	**Contains tns1:AccessControl/ AccessGranted/ Anonymous Event topic**	**Does not contain tns1:AccessControl/ AccessGranted/ Anonymous Event topic**
AccessControl/ AccessGranted/Credential	**Contains tns1:AccessControl/**	**Does not contain tns1:AccessControl/**

Criterion Item	GetEventPropertiesResponse message	
Feature	**Supported**	**Not Supported**
	AccessGranted/ Credential Event topic	**AccessGranted/ Credential Event topic**
AccessControl/ AccessTaken/Anonymous	**Contains tns1:AccessControl/ AccessTaken/ Anonymous Event topic**	**Does not contain tns1:AccessControl/ AccessTaken/ Anonymous Event topic**
AccessControl/ AccessTaken/Credential	**Contains tns1:AccessControl/ AccessTaken/ Credential Event topic**	**Does not contain tns1:AccessControl/ AccessTaken/ Credential Event topic**
AccessControl/ AccessNotTaken/Anonymous	**Contains tns1:AccessControl/ AccessNotTaken/ Anonymous Event topic**	**Does not contain tns1:AccessControl/ AccessNotTaken/ Anonymous Event topic**
AccessControl/ AccessNotTaken/Credential	**Contains tns1:AccessControl/ AccessNotTaken/ Credential Event topic**	**Does not contain tns1:AccessControl/ AccessNotTaken/ Credential Event topic**
AccessControl/ Denied/Anonymous	**Contains tns1:AccessControl/Denied/ Anonymous Event topic**	**Does not contain tns1:AccessControl/Denied/ Anonymous Event topic**
AccessControl/ Denied/Credential	**Contains tns1:AccessControl/Denied/ Credential Event topic**	**Does not contain tns1:AccessControl/Denied/ Credential Event topic**
AccessControl/Denied/ CredentialNotFound/Card	**Contains tns1:AccessControl/Denied/ CredentialNotFound/ Card Event topic**	**Does not contain tns1:AccessControl/Denied/ CredentialNotFound/ Card Event topic**
AccessControl/Duress	**Contains tns1:AccessControl/ Duress Event topic**	**Does not contain tns1:AccessControl/ Duress Event topic**
AccessControl/ Request/Anonymous	**Contains tns1:AccessControl/ Request/Anonymous Event topic**	**Does not contain tns1:AccessControl/Request/ Anonymous Event topic**

Criterion Item	GetEventPropertiesResponse message	
Feature	**Supported**	**Not Supported**
AccessControl/ Request/Credential	**Contains tns1:AccessControl/ Request/Credential Event topic**	**Does not contain tns1:AccessControl/Request/ Credential Event topic**
AccessControl/ Request/Timeout	**Contains tns1:AccessControl/ Request/ Timeout Event topic**	**Does not contain tns1:AccessControl/ Request/ Timeout Event topic**
AccessPoint/State/Enabled	**Contains tns1:AccessPoint/ State/ Enabled Event topic**	**Does not contain tns1:AccessPoint/State/ Enabled Event topic**
Configuration/ AccessPoint/Changed	**Contains tns1:Configuration/ AccessPoint/ Changed Event topic**	**Does not contain tns1:Configuration/ AccessPoint/ Changed Event topic**
Configuration/ AccessPoint/Removed	**Contains tns1:Configuration/ AccessPoint/ Removed Event topic**	**Does not contain tns1:Configuration/ AccessPoint/ Removed Event topic**
Configuration/Area/Changed	**Contains tns1:Configuration/ Area/ Changed Event topic**	**Does not contain tns1:Configuration/Area/ Changed Event topic**
Configuration/Area/Removed	**Contains tns1:Configuration/ Area/ Removed Event topic**	**Does not contain tns1:Configuration/Area/ Removed Event topic**
AccessControl/Denied/ CredentialNotFound	**Contains tns1:AccessControl/ Denied/ CredentialNotFound Event topic**	**Does not contain tns1:AccessControl/ Denied/ CredentialNotFound Event topic**
AccessControl/ AccessGranted/Identifier	**Contains tns1:AccessControl/ AccessGranted/ Identifier Event topic**	**Does not contain tns1:AccessControl/ AccessGranted/ Identifier Event topic**
AccessControl/Denied/Identifier	**Contains tns1:AccessControl/ Denied/ Identifier Event topic**	**Does not contain tns1:AccessControl/ Denied/ Identifier Event topic**

5.5.17.4 Access Control Service Features Support

Access Control capabilities support under Access Control Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Access Control Service is supported by the DUT as defined in Section 5.5.16, otherwise all features defined in Table 5.64 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client invokes **GetServiceCapabilities** request for Access Control Service.
2. The DUT responds with **GetServiceCapabilitiesResponse** message with parameters
   - Capabilities =: *cap*
3. ONVIF Client checks features support as defined in Table 5.64.

Note: If the DUT does not return GetServiceCapabilitiesResponse, then all features defined in Table 5.64 will be marked as UNDEFINED.

Table 5.64. Access Control Features Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Access Point Management	***cap*.AccessPointManagementSupported = true**	**Skipped *cap*.AccessPointManagementSupported or *cap*.AccessPointManagementSupported = false**
Area Management	***cap*.AreaManagementSupported = true**	**Skipped *cap*.AreaManagementSupported or *cap*.AreaManagementSupported = false**
Client Supplied Token (Area, Access Point)	***cap*.ClientSuppliedTokenSupported = true**	**Skipped *cap*.ClientSuppliedTokenSupported or *cap*.ClientSuppliedTokenSupported = false**

5.5.18 Security Configuration Service Support

Security Configuration Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.65.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/advancedsecurity/wsdl" namespace ONVIF Client will use service with the latest version.

Note: If Security Configuration Service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.18.1.

Table 5.65. Security Configuration Service (GetServices)

Criterion Item	GetServicesResponse message	
Feature	**Supported**	**Not Supported**
Security Configuration Service	**Includes service with "http://www.onvif.org/ver10/advancedsecurity/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/advancedsecurity/wsdl" namespace**

5.5.18.1 Security Configuration Features Support

Keystore, TLS Server, and 802.1X configuration capabilities support under Security Configuration Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Security Configuration Service is supported by the DUT as defined in Section 5.5.18, otherwise all features defined in Table 5.66 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client invokes **GetServiceCapabilities** request for Security Configuration Service.
2. The DUT responds with **GetServiceCapabilitiesResponse** message with parameters

- Capabilities =: *cap*

3. ONVIF Client checks features support as defined in Table 5.66.

Note: If the DUT does not return GetServiceCapabilitiesResponse then all features defined in Table 5.66 will be marked as UNDEFINED.

Table 5.66. Security Configuration Features Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
RSA Key Pair Generation	**Capabilities. KeystoreCapabilities. RSAKeyPairGeneration = true**	**Skipped Capabilities. KeystoreCapabilities. RSAKeyPairGeneration or Capabilities. KeystoreCapabilities. RSAKeyPairGeneration = false**
PKCS10 External Certification with RSA	**Capabilities. KeystoreCapabilities. PKCS10ExternalCertification WithRSA = true**	**Skipped Capabilities. KeystoreCapabilities. PKCS10ExternalCertification WithRSA or Capabilities. KeystoreCapabilities. PKCS10ExternalCertification WithRSA = false**
Self-Signed Certificate Creation with RSA	**Capabilities. KeystoreCapabilities. SelfSignedCertificate CreationWithRSA = true**	**Skipped Capabilities. KeystoreCapabilities. SelfSignedCertificateCreation WithRSA or Capabilities. KeystoreCapabilities. SelfSignedCertificateCreation WithRSA = false**
Passphrase Management	**Capabilities. KeystoreCapabilities. MaximumNumberOf Passphrases > 0**	**Skipped Capabilities. KeystoreCapabilities. MaximumNumberOf Passphrases or Capabilities. KeystoreCapabilities. MaximumNumberOf Passphrases = 0**
PKCS8 Container Upload	**Capabilities. KeystoreCapabilities.**	**Skipped Capabilities. KeystoreCapabilities.**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
	PKCS8RSAKeyPairUpload = true	**PKCS8RSAKeyPairUpload or Capabilities. KeystoreCapabilities. PKCS8RSAKeyPairUpload = false**
PKCS12 Container Upload	**Capabilities. KeystoreCapabilities. PKCS12CertificateWithRSA PrivateKeyUpload = true**	**Skipped Capabilities. KeystoreCapabilities. PKCS12CertificateWithRSA PrivateKeyUpload or Capabilities. KeystoreCapabilities. PKCS12CertificateWithRSA PrivateKeyUpload = false**
CRLs	**Capabilities. KeystoreCapabilities. MaximumNumberOfCRLs > 0**	**Skipped Capabilities. KeystoreCapabilities. MaximumNumberOfCRLs or Capabilities. KeystoreCapabilities. MaximumNumberOfCRLs = 0**
Certification path validation policies	**Capabilities. KeystoreCapabilities. MaximumNumber OfCertificationPath ValidationPolicies > 0**	**Skipped Capabilities. KeystoreCapabilities. MaximumNumber OfCertificationPath ValidationPolicies or Capabilities. KeystoreCapabilities. MaximumNumber OfCertificationPath ValidationPolicies = 0**
TLS WWW client auth extended key usage extension	**Capabilities. KeystoreCapabilities. EnforceTLSWebClientAuth ExtKeyUsage = true**	**Skipped Capabilities. KeystoreCapabilities. EnforceTLSWebClientAuth ExtKeyUsage or Capabilities. KeystoreCapabilities. EnforceTLSWebClientAuth ExtKeyUsage = false**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
No Private Key Sharing	**Capabilities. KeystoreCapabilities. NoPrivateKeySharing = true**	**Skipped Capabilities. KeystoreCapabilities. NoPrivateKeySharing or Capabilities. KeystoreCapabilities. NoPrivateKeySharing = false**
TLS Server Support	**Capabilities. TLSServerCapabilities. TLSServerSupported list is not empty**	**Skipped Capabilities. TLSServerCapabilities. TLSServerSupported or empty Capabilities. TLSServerCapabilities. TLSServerSupported list**
TLS client authentication	**Capabilities. TLSServerCapabilities. TLSClientAuthSupported = true**	**Skipped Capabilities. TLSServerCapabilities. TLSClientAuthSupported or Capabilities. TLSServerCapabilities. TLSClientAuthSupported = false**
Enabled TLS Versions	**Capabilities. TLSServerCapabilities. EnabledVersionsSupported = true**	**Skipped Capabilities. TLSServerCapabilities. EnabledVersionsSupported or Capabilities. TLSServerCapabilities. EnabledVersionsSupported = false**

5.5.19 Credential Service Support

Credential Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.67.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/credential/wsdl" namespace, ONVIF Client will use service with the latest version.

Note: If Credential service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.19.1.

Table 5.67. Credential Service (GetServices)

Criterion Item	GetServicesResponse message	
Feature	**Supported**	**Not Supported**
Credential Service	**Includes service with "http://www.onvif.org/ver10/credential/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/credential/wsdl" namespace**

5.5.19.1 Credential Features Support

Credential capabilities support under Credential Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Credential Service is supported by the DUT as defined in Section 5.5.19, otherwise all features defined in Table 5.68 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client invokes **GetServiceCapabilities** request for Credential Service.
2. The DUT responds with **GetServiceCapabilitiesResponse** message with parameters
   - Capabilities =: *cap*
3. ONVIF Client checks features support as defined in Table 5.68.

Note: If the DUT does not return GetServiceCapabilitiesResponse, then all features defined in Table 5.68 will be marked as UNDEFINED.

Table 5.68. Credential Features Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
pt:Card	***cap*.SupportedIdentifierType contains pt:Card**	***cap*.SupportedIdentifierType does not contain pt:Card**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
pt:PIN	***cap*.SupportedIdentifierType contains pt:PIN**	***cap*.SupportedIdentifierType does not contain pt:PIN**
pt:Fingerprint	***cap*.SupportedIdentifierType contains pt:Fingerprint**	***cap*.SupportedIdentifierType does not contain pt:Fingerprint**
pt:Face	***cap*.SupportedIdentifierType contains pt:Face**	***cap*.SupportedIdentifierType does not contain pt:Face**
pt:Iris	***cap*.SupportedIdentifierType contains pt:Iris**	***cap*.SupportedIdentifierType does not contain pt:Iris**
pt:Vein	***cap*.SupportedIdentifierType contains pt:Vein**	***cap*.SupportedIdentifierType does not contain pt:Vein**
Credential Validity	***cap*. CredentialValiditySupported = true**	***cap*. CredentialValiditySupported = false**
Credential Access Profile Validity	***cap*.CredentialAccessProfile ValiditySupported = true**	***cap*.CredentialAccessProfile ValiditySupported = false**
Validity Supports Time Value	***cap*.ValiditySupportsTimeValue = true**	***cap*.ValiditySupportsTimeValue = false**
Reset Antipassback Violation	***cap*. ResetAntipassbackSupported = true**	***cap*. ResetAntipassbackSupported = false**
pt:ExemptFromAuthentication	***cap*.Extension. SupportedExemptionType = pt:ExemptFromAuthentication**	***cap*.Extension does not contain SupportedExemptionType with value = pt:ExemptFromAuthentication**
Credential Service \Client Supplied Token	***cap*. ClientSuppliedTokenSupported = true**	**Skipped *cap*. ClientSuppliedTokenSupported or *cap*. ClientSuppliedTokenSupported = false**
Whitelist	***cap*.MaxWhitelistedItems > 0**	**Skipped *cap*.MaxWhitelistedItems or *cap*.MaxWhitelistedItems = 0**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Blacklist	***cap*.MaxBlacklistedItems > 0**	**Skipped *cap*.MaxBlacklistedItems or *cap*.MaxBlacklistedItems = 0**

5.5.20 Access Rules Service Support

Access Rules Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.69.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/accessrules/wsdl" namespace, ONVIF Client will use service with the latest version.

Note: If Credential service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.20.1.

Table 5.69. Access Rules Service (GetServices)

Criterion Item	GetServicesResponse message	
Feature	**Supported**	**Not Supported**
Access Rules Service	**Includes service with "http://www.onvif.org/ver10/accessrules/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/accessrules/wsdl" namespace**

5.5.20.1 Access Rules Features Support

Access Rules capabilities support under Access Rules Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Access Rules Service is supported by the DUT as defined in Section 5.5.20, otherwise all features defined in Table 5.70 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client invokes **GetServiceCapabilities** request for Access Rules Service.
2. The DUT responds with **GetServiceCapabilitiesResponse** message with parameters
   - Capabilities =: *cap*
3. ONVIF Client checks features support as defined in Table 5.70.

Note: If the DUT does not return GetServiceCapabilitiesResponse, then all features defined in Table 5.70 will be marked as UNDEFINED.

Table 5.70. Access Rules Features Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Multiple Schedules per Access Point	***cap*. MultipleSchedulesPerAccessPointSupported = true**	***cap*. MultipleSchedulesPerAccessPointSupported = false**
Access Rules Service \Client Supplied Token	***cap*. ClientSuppliedTokenSupported = true**	**Skipped *cap*. ClientSuppliedTokenSupported or *cap*. ClientSuppliedTokenSupported = false**

5.5.21 Schedule Service Support

Schedule Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.71.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/schedule/wsdl" namespace, ONVIF Client will use service with the latest version.

Note: If Schedule service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.5.21.1.

Table 5.71. Schedule Service (GetServices)

Criterion Item	GetServicesResponse message	
Feature	**Supported**	**Not Supported**
Schedule Service	**Includes service with "http://www.onvif.org/ver10/schedule/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/schedule/wsdl" namespace**

5.5.21.1 Schedule Features Support

Schedule capabilities support under Schedule Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Schedule Service is supported by the DUT as defined in Section 5.5.21, otherwise all features defined in Table 5.72 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client invokes **GetServiceCapabilities** request for Schedule Service.
2. The DUT responds with **GetServiceCapabilitiesResponse** message with parameters
   - Capabilities =: *cap*
3. ONVIF Client checks features support as defined in Table 5.72.

Note: If the DUT does not return GetServiceCapabilitiesResponse, then all features defined in Table 5.72 will be marked as UNDEFINED.

Table 5.72. Schedule Features Support (GetServices)

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Extended Recurrence	***cap*.ExtendedRecurrenceSupported = true**	***cap*.ExtendedRecurrenceSupported = false**
Special Days	***cap*.SpecialDaysSupported = true**	***cap*.SpecialDaysSupported = false**

Criterion Item	GetServiceCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
State Reporting	***cap*.StateReportingSupported = true**	***cap*.StateReportingSupported = false**
Schedule Service \Client Supplied Token	***cap*. ClientSuppliedTokenSupported = true**	**Skipped *cap*. ClientSuppliedTokenSupported or *cap*. ClientSuppliedTokenSupported = false**

5.5.22 Provisioning Service Support

Provisioning Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.73.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/provisioning/wsdl" namespace ONVIF Client will use service with the latest version.

Table 5.73. Provisioning Service (GetServices)

Criterion Item	GetServicesResponse	
Feature	**Supported**	**Not Supported**
Provisioning Service	**Includes service with "http://www.onvif.org/ver10/provisioning/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/provisioning/wsdl" namespace**

5.5.23 Thermal Service Support

Thermal Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetServicesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.74.

Note: If GetServicesResponse contains several services with "http://www.onvif.org/ver10/thermal/wsdl" namespace ONVIF Client will use service with the latest version.

Table 5.74. Thermal Service (GetServices)

Criterion Item	GetServicesResponse	
Feature	**Supported**	**Not Supported**
Thermal Service	**Includes service with "http://www.onvif.org/ver10/thermal/wsdl" namespace**	**Does not include service with "http://www.onvif.org/ver10/thermal/wsdl" namespace**

5.6 Discovery Procedure (GetCapabilities)

If only GetCapabilities is supported by the DUT, then GetCapabilities command will be used for feature discovery procedure. The following provides with the functionality discovery procedure for this case.

5.6.1 Device Service Capabilities Configuration Functionality in Device Management Service

There are various network configuration functions defined in [ONVIF Core] as a part of ONVIF Device Management Service. The following provides with the functionality discovery procedure which is related to network configuration.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.75.

Note: Absence of Capabilities.Device or Capabilities.Device.Network element in GetCapabilitiesResponse will be regarded as no support for the following functionalities:

- ZeroConfiguration
- IPv6
- IP Filter
- Dynamic DNS
- DHCPv6

Note: Absence of Capabilities.Device or Capabilities.Device.System or Capabilities.Device.Security element in GetCapabilitiesResponse will be regarded as no support for the following functionalities:

- BYE Message
- System logging
- HttpFirmwareUpgrade
- TLS1.0
- TLS1.1
- TLS1.2

Note: Since the DUT does not support GetServices feature, the following functionalities are defined as not supported:

- Maximum Users
- Maximum Username Length
- Maximum Password Length
- DefaultAccessPolicy
- Auxiliary Commands support

Table 5.75. Device Service Capabilities Configuration Functionality in Device Management Service (GetCapabilities)

Criterion Item	GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
IPv6	**Capabilities.Device.Network.IPVersion6 = true**	**Skipped Capabilities.Device.Network.IPVersion6 or Capabilities.Device.Network.IPVersion6 = false**
Zero Configuration	**Capabilities.Device.Network.ZeroConfiguration = true**	**Skipped Capabilities.Device.Network.ZeroConfiguration or Capabilities.Device.Network.ZeroConfiguration = false**
Dynamic DNS	**Capabilities.Device.Network.DynDNS = true**	**Skipped Capabilities.Device.Network.**

Criterion Item	GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
		DynDNS or Capabilities.Device.Network. DynDNS = false
IP Filter	**Capabilities.Device.Network. IPFilter = true**	**Skipped Capabilities.Device. Network.IPFilter or Capabilities.Device.Network. IPFilter = false**
Stateful IPv6 DHCP	**Capabilities.Device.Network. DHCPv6 = true**	**Skipped Capabilities.Device. Network.DHCPv6 or Capabilities.Device.Network. DHCPv6 = false**
Remote User Handling	**Capabilities.Device.Security. Extension.Extension. RemoteUserHandling = true**	**Skipped Capabilities.Device. Security. Extension.Extension. RemoteUserHandling or Capabilities.Security. Extension.Extension. RemoteUserHandling = false**
TLS1.0	**Capabilities.Device.Security. Extension.TLS1.0 = true**	**Skipped Capabilities.Device.Security. Extension or Capabilities.Device.Security. Extension.TLS1.0 = false**
TLS1.1	**Capabilities.Device. Security.TLS1.1 = true**	**Skipped Capabilities.Device.Security or Capabilities.Device.Security. TLS1.1 = false**
TLS1.2	**Capabilities.Device. Security.TLS1.2 = true**	**Skipped Capabilities.Device.Security or Capabilities.Device.Security. TLS1.2 = false**
Bye Message	**Capabilities.Device.System. DiscoveryBye = true**	**Skipped Capabilities.Device. System.DiscoveryBye or Capabilities.Device.System. DiscoveryBye = false**

Criterion Item	GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
System logging	**Capabilities.Device.System. SystemLogging = true**	**Skipped Capabilities.Device. System.SystemLogging or Capabilities.Device.System. SystemLogging = false**
Http Firmware Upgrade	**Capabilities.System. Extension. HttpFirmwareUpgrade = true**	**Skipped Capabilities.System. Extension. HttpFirmwareUpgrade or Capabilities. System. Extension. HttpFirmwareUpgrade = false**
Http System Backup	**Capabilities.System. Extension.HttpSystemBackup = true**	**Skipped Capabilities.System. Extension.HttpSystemBackup or Capabilities.System.Extension. HttpSystemBackup = false**
Http System Logging	**Capabilities.System. Extension.HttpSystemLogging = true**	**Skipped Capabilities.System. Extension.HttpSystemLogging or Capabilities.System.Extension. HttpSystemLogging = false**
Http Support Information	**Capabilities.System. Extension. HttpSupportInformation = true**	**Skipped Capabilities.System. Extension. HttpSupportInformation or Capabilities.System.Extension. HttpSupportInformation = false**

5.6.2 HTTPS Support

The following is the procedure to determine the function support.

Pre-requisite: ONVIF Client and DUT

- If DUT supports HTTPS, then HTTPS is configured on the DUT.

Discovery Procedure:

- ONVIF Client invokes GetNetworkProtocols request message to retrieve network protocols supported by DUT.

- The DUT returns GetNetworkProtocolsResponse. ONVIF Client checks features support as defined in Table 5.76.

Note: If the DUT does not return GetNetworkProtocolsResponse, ONVIF Client assumes that RTP/RTSP/HTTPS function support is marked as undefined.

Table 5.76. HTTPS Support (GetCapabilities)

Criterion Item	GetNetworkProtocolsResponse	
Feature	**Supported**	**Not Supported**
HTTPS	**Includes NetworkProtocols element with Name = HTTPS and with Enabled=true or DUT supports TLS Server feature**	**Does not include NetworkProtocols element with Name = HTTPS or includes NetworkProtocols element with Name = HTTPS and with Enabled=false**

5.6.3 Security (HTTP Digest Authentication) Support

In the first version of [ONVIF Core], WS-UsernameToken support was the only method defined as a mandatory feature for user authentication. This has been changed in the later version of [ONVIF Core] where it also defines the HTTP digest authentication support as a mandatory feature.

The following discovery procedure will be performed for ONVIF Client to determine which user authentication function will be used in conformance testing.

Discovery Procedure:

1. ONVIF Client invokes request message for command with security support without any user authentication (no WS-Security and no HTTP digest authentication) to check HTTP digest authentication support.

2. ONVIF Client checks features support as defined in Table 5.77.

Note: The command which is being used for this discovery procedure should be provided before performing the steps.

Note: If HTTP digest authentication is assumed as supported, the HTTP digest authentication scheme will be used in the following feature discovery procedure whenever necessary as well as in conformance testing.

Table 5.77. HTTP Digest Authentication (GetCapabilities)

Criterion Item	Command with security support response message	
Feature	**Supported**	**Not Supported**
HTTP digest	**HTTP 401 Unauthorized error**	**Not HTTP 401 Unauthorized error**

5.6.4 NTP Support

Whether the DUT supports NTP functionality is determined by the following discovery procedure.

Discovery Procedure:

1. ONVIF Client invokes GetNTPRequest.
2. ONVIF Client checks features support as defined in Table 5.78.

Note: In any other case than the above, NTP function support will be marked as undefined.

Table 5.78. NTP Functionality (GetCapabilities)

Criterion Item	GetNTPResponse	
Feature	**Supported**	**Not Supported**
NTP	**DUT returns GetNTP Response**	**DUT returns any SOAP fault**

5.6.5 I/O Functionality in Device Management Service

I/O related functionality support can be retrieved by checking correspondent element of GetCapabilitiesResponse. The following is the procedure to determine the function support.

Pre-requisite:

- This procedure assumes that GetCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client invokes GetCapabilitiesRequest to check I/O functionality support.
2. ONVIF Client receives GetCapabilitiesResponse and checks features support as defined in Table 5.79.
3. ONVIF Client invokes GetRelayOutputsRequest message to retrieve a relay output list.
4. The DUT returns GetRelayOutputsResponse with a list of relay outputs.

5. ONVIF Client invokes SetRelayOutputSettingsRequest message (RelayOutputToken = "[first token from GetRelayOutputsResponse]", Properties.Mode = "Bistable", Properties.DelayTime = "PT30S", Properties.IdleState = "open").
6. ONVIF Client receives SetRelayOutputSettingsResponse.
7. ONVIF Client invokes SetRelayOutputSettingsRequest message (RelayOutputToken = "[first token from GetRelayOutputsResponse]", Properties.Mode = "Bistable", Properties.DelayTime = "PT30S", Properties.IdleState = "closed").
8. ONVIF Client receives SetRelayOutputSettingsResponse.
9. ONVIF Client invokes SetRelayOutputSettingsRequest message (RelayOutputToken = "[first token from GetRelayOutputsResponse]", Properties.Mode = "Monostable", Properties.DelayTime = "PT30S", Properties.IdleState = "open").
10. ONVIF Client receives SetRelayOutputSettingsResponse.
11. ONVIF Client invokes SetRelayOutputSettingsRequest message (RelayOutputToken = "[first token from GetRelayOutputsResponse]", Properties.Mode = "Monostable", Properties.DelayTime = "PT30S", Properties.IdleState = "closed").
12. ONVIF Client receives SetRelayOutputSettingsResponse.
13. ONVIF Client checks features support as defined in Table 5.80.

Note: Absence of Capabilities.Device.IO element in the GetCapabilitiesResponse will be defined as absence of Capabilities.Device.IO.RelayOutputs.

Note: If Capabilities.Device element is not included in the GetCapabilitiesResponse, Relay Outputs feature will be marked as unsupported.

Note: If DUT does not return GetRelayOutputsResponse or a list of relay outputs in the GetRelayOutputsResponse is empty, Relay Outputs features will be marked as undefined.

Note: If GetCapabilities command is not supported by the DUT, I/O feature for Device Management Service will be defined as unsupported.

Table 5.79. Relay Outputs in Device Management Service (GetCapabilities)

Criterion Item	GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
RelayOutputs	**Capabilities.Device.IO. RelayOutputs > 0**	**Skipped Capabilities.Device.IO. RelayOutputs or Capabilities. Device.IO.RelayOutputs = 0**

Table 5.80. Relay Outputs Mode and Idle State in Device Management Service (GetCapabilities)

Criterion Item	SetRelayOutputSettingsResponse	
Feature	**Supported**	**Not Supported**
Bistable Mode/Open Idle State	**DUT returns SetRelayOutputSettings Response for step 6.**	**DUT returns any SOAP fault for step 6.**
Bistable Mode/Closed Idle State	**DUT returns SetRelayOutputSettings Response for step 8.**	**DUT returns any SOAP fault for step 8.**
Bistable Mode	**DUT returns SetRelayOutputSettings Response for step 6 or 8.**	**DUT returns any SOAP fault for step 6 and 8.**
Monostable Mode/ Open Idle State	**DUT returns SetRelayOutputSettings Response for step 9.**	**DUT returns any SOAP fault for step 9.**
Monostable Mode/ Closed Idle State	**DUT returns SetRelayOutputSettings Response for step 11.**	**DUT returns any SOAP fault for step 11.**
Monostable Mode	**DUT returns SetRelayOutputSettings Response for step 9 or 11.**	**DUT returns any SOAP fault for step 9 and 11.**

5.6.6 Monitoring Events Support

Monitoring Events support under Device Control Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetEventPropertiesResponse has already been retrieved via preceding procedure described in Section 5.6.9.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.81.

Note: If the DUT does not return Event Service or GetEventPropertiesResponse message, then the following features will be marked as undefined:

- Monitoring/ProcessorUsage

- Monitoring/OperatingTime/LastReset
- Monitoring/OperatingTime/LastReboot
- Monitoring/OperatingTime/LastClockSynchronization
- Monitoring/Backup/Last
- Device/HardwareFailure/TemperatureCritical
- Device/HardwareFailure/FanFailure
- Device/HardwareFailure/PowerSupplyFailure
- Device/HardwareFailure/StorageFailure

Table 5.81. Monitoring Events Support (GetCapabilities)

Criterion Item	GetEventPropertiesResponse	
Feature	**Supported**	**Not Supported**
Monitoring/ProcessorUsage	**Contains tns1:Monitoring/ ProcessorUsage Event topic**	**Does not contain tns1:Monitoring/ ProcessorUsage Event topic**
Monitoring/ OperatingTime/LastReset	**Contains tns1:Monitoring/ OperatingTime/ LastReset Event topic**	**Does not contain tns1:Monitoring/ OperatingTime/ LastReset Event topic**
Monitoring/ OperatingTime/LastReboot	**Contains tns1:Monitoring/ OperatingTime/ LastReboot Event topic**	**Does not contain tns1:Monitoring/ OperatingTime/ LastReboot Event topic**
Monitoring/OperatingTime/ LastClockSynchronization	**Contains tns1:Monitoring/ OperatingTime/ LastClockSynchronization Event topic**	**Does not contain tns1:Monitoring/ OperatingTime/ LastClockSynchronization Event topic**
Monitoring/Backup/Last	**Contains tns:Monitoring/ Backup/Last Event topic**	**Does not contain tns:Monitoring/Backup/ Last Event topic**
Device/HardwareFailure/ TemperatureCritical	**Contains tns1:Device/ HardwareFailure/ TemperatureCritical Event topic**	**Does not contain tns1:Device/ HardwareFailure/ TemperatureCritical Event topic**

Criterion Item	GetEventPropertiesResponse	
Feature	**Supported**	**Not Supported**
Device/HardwareFailure/ FanFailure	**Contains tns1:Device/ HardwareFailure/ FanFailure Event topic**	**Does not contain tns1:Device/ HardwareFailure/ FanFailure Event topic**
Device/HardwareFailure/ PowerSupplyFailure	**Contains tns1:Device/ HardwareFailure/ PowerSupplyFailure Event topic**	**Does not contain tns1:Device/ HardwareFailure/ PowerSupplyFailure Event topic**
Device/HardwareFailure/ StorageFailure	**Contains tns1:Device/ HardwareFailure/ StorageFailure Event topic**	**Does not contain tns1:Device/ HardwareFailure/ StorageFailure Event topic**

5.6.7 Media Service – General

Media Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.82.

Note: If Media service is not supported, the following feature discovery (Media Service features support) will be skipped.

Table 5.82. Media Service – General (GetCapabilities)

Criterion Item	GetCapabilitiesResponse	
Feature	**Supported**	**Not Supported**
Media Service	**Includes Capabilities.Media element**	**Does not include Capabilities.Media element**

5.6.7.1 Media Service – Video Encoding Support

Video encoding function support in Media Service is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetVideoEncoderConfigurationOptionsRequest (no ConfigurationToken, no ProfileToken) message to retrieve all supported codecs.
2. The DUT returns GetVideoEncoderConfigurationOptionsResponse with a list of supported codecs. ONVIF Client checks features support as defined in Table 5.83.

Note: If the DUT does not return GetVideoEncoderConfigurationOptionsResponse, MPEG4 and H.264 feature will be marked as undefined.

Table 5.83. Media Service – Video Encoding Support (GetCapabilities)

Criterion Item	GetVideoEncoderConfigurationOptionsResponse	
Feature	**Supported**	**Not Supported**
JPEG	**Mandatory functionality**	**-**
MPEG-4	**Includes Options.MPEG4**	**Does not include Options.MPEG4**
H.264	**Includes Options.H264**	**Does not include Options.H264**

5.6.7.2 Media Service – Audio Encoding Support

Audio encoding function support in Media Service is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetAudioEncoderConfigurationOptionsRequest (no ConfigurationToken, no ProfileToken) message to retrieve all supported audio codecs.
2. The DUT returns GetAudioEncoderConfigurationOptionsResponse with a list of supported codecs or SOAP fault. ONVIF Client checks features support as defined in Table 5.84.

Note: If the DUT returns no response for GetAudioEncoderConfigurationOptionsRequest, Audio encoding feature will be marked as undefined.

Table 5.84. Media Service – Audio Encoding Support (GetCapabilities)

Criterion Item	GetAudioEncoderConfigurationOptionsResponse	
Feature	**Supported**	**Not Supported**
Audio encoding	**DUT returns GetAudioEncoderConfigurationOptionsResponse**	**DUT returns any SOAP fault**

Criterion Item	GetAudioEncoderConfigurationOptionsResponse	
Feature	**Supported**	**Not Supported**
G.711	**DUT returns GetAudioEncoderConfigurationOptionsResponse**	**DUT returns any SOAP fault**
G.726	**Includes Options.Options.Encoding = "G726"**	**Does not include Options.Options.Encoding = "G726"**
AAC	**Includes Options.Options.Encoding = "AAC"**	**Does not include Options.Options.Encoding = "AAC"**

5.6.7.3 Media Service – Real-Time Streaming

Since the DUT does not support GetServices feature Real-time streaming feature will be defined as supported.

5.6.7.4 Media Service – Supported Real-Time Streaming Setup

Which Real-time streaming Setup features is supported under Real-time Streaming is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetCapabilitiesRequest message to check Multicast streaming capability support by the DUT.
2. The DUT returns GetCapabilitiesResponse. ONVIF Client checks features support as defined in Table 5.85.

Note: If DUT does not return GetCapabilitiesResponse then RTP Multicast streaming (UDP) and RTP/RTSP/TCP Setup features will be marked as undefined.

Table 5.85. Media Service – Supported Real-time Streaming Setup (GetCapabilities)

Criterion Item	GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
RTP/UDP	**Mandatory functionality**	-
RTP/RTSP/HTTP	**Mandatory functionality**	-

Criterion Item	GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
RTP/RTSP/TCP	**Capabilities. Media.StreamingCapabilities. RTP_RTSP_TCP = true**	**Skipped Capabilities. Media.StreamingCapabilities. RTP_RTSP_TCP or Capabilities. StreamingCapabilities. RTP_RTSP_TCP = false**
RTP-Multicast/UDP	**Capabilities. Media.StreamingCapabilities. RTPMulticast = true**	**Skipped Capabilities. Media.StreamingCapabilities. RTPMulticast or Capabilities. Media.StreamingCapabilities. RTPMulticast = false**

5.6.7.5 Media Service - GetSnapshotUri

GetSnapshotUri function support is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetProfilesRequest message to retrieve existing Media Profiles list.
2. The DUT returns GetProfilesResponse with the list of existing Media Profiles.
3. ONVIF Client looks for ready-to-use profile (a profile with VideoSourceConfiguration and VideoEncoderConfiguration in the GetProfilesResponse. If there are no ready-to-use profiles found in the GetProfilesResponse, ONVIF Client marks GetSnapshotUri support by DUT as undefined.
4. ONVIF Client invokes GetSnapshotUriRequest (ProfileToken = found ready-to-use profile token) message to get Snapshot URI.
5. The DUT returns GetSnapshotUriResponse or SOAP fault. ONVIF Client checks features support as defined in Table 5.86.

Note: If no GetProfilesResonse is returned by the DUT, GetSnapshotUri function support by the DUT is marked as undefined.

Note: If no GetSnapshotUriResponse is returned by the DUT, GetSnapshotUri function support by the DUT is marked as undefined.

Table 5.86. Media Service – GetSnapshotUri (GetCapabilities)

Criterion Item	GetSnapshotUriResponse	
Feature	**Supported**	**Not Supported**
GetSnapshotUri	**DUT returns GetSnapshotUriResponse**	**DUT returns any SOAP fault**

5.6.7.6 Media Service – Audio Outputs Support

Audio outputs support in conjunction with its Audio decoding function is determined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetAudioOutputsRequest message to retrieve Audio outputs list.
2. The DUT returns GetAudioOutputsResponse or SOAP fault. ONVIF Client checks features support as defined in Table 5.87. Go to the next feature definition.
3. ONVIF Client invokes GetAudioDecoderConfigurationOptionsRequest (no ConfigurationToken, no ProfileToken) message to retrieve all supported Audio codec's for decoding by DUT.
4. The DUT returns GetAudioDecoderConfigurationOptionsResponse. ONVIF Client checks features support as defined in Table 5.88.

Note: If the DUT does not return GetAudioDecoderConfigurationOptionsResponse, ONVIF Client assumes that G.711, G.726 and AAC Audio decoding function support is marked as undefined.

Table 5.87. Media Service – Audio Outputs Support (GetCapabilities)

Criterion Item	GetAudioOutputsResponse	
Feature	**Supported**	**Not Supported**
Audio output	**DUT returns GetAudioOutputsResponse and there are at least one AudioOutput on the list**	**DUT returns any SOAP fault or GetAudioOutputsResponse and there are no AudioOutput on the list**

Table 5.88. Media Service – Audio Outputs Decoding Support (GetCapabilities)

Criterion Item	GetAudioDecoderConfigurationOptionsResponse	
Feature	**Supported**	**Not Supported**
G.711	**Includes Options.G711DecOptions**	**Does not include Options.G711DecOptions**
G.726	**Includes Options.G726DecOptions**	**Does not include Options.G726DecOptions**
AAC	**Includes Options.AACDecOptions**	**Does not include Options.AACDecOptions**

5.6.8 Media2 Service Support

Since the DUT does not support GetServices feature Media2 Service feature will be defined as not supported.

5.6.9 Event Service

Event Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.89.

Note: If Event service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.6.9.1.
- Section 5.6.9.2.
- Section 5.6.9.3.
- Section 5.6.9.4.

Table 5.89. Event Service (GetCapabilities)

Criterion Item	GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Event Service	**Includes Capabilities.Events element**	**Does not include Capabilities.Events element**

5.6.9.1 Get Event Properties

The following discovery procedure will be performed for ONVIF Client to receive topic set and message content filter dialect list from the DUT. Received topic set will be used in event topics discovery procedures for different services. Received message content filter dialect list will be used in Section 5.6.9.3 and in Section 5.6.9.4.

Pre-requisite:

- This procedure assumes that Event Service is supported by the DUT as defined in Section 5.6.9, otherwise this procedure will be skipped.

Discovery Procedure:

1. ONVIF Client invokes **GetEventProperties** message.
2. The DUT responds with a **GetEventPropertiesResponse** message with parameters
   - TopicNamespaceLocation list
   - FixedTopicSet
   - TopicSet =: *topicSet*
   - TopicExpressionDialect list
   - MessageContentFilterDialect list := *msgContentFilterDialectList*
   - MessageContentSchemaLocation list

5.6.9.2 Event service features

Pre-requisite:

- This procedure assumes that Event Service is supported by the DUT as defined in Section 5.6.9, otherwise this procedure will be skipped.

Since the DUT does not support GetServices feature, Event Service sub-features will be defined as described below:

- Persistent Notification Storage under Event Service is not supported by the DUT.
- WS Basic Notification under Event Service is supported by the DUT.
- GetServiceCapabilities\MaxPullPoints capability is not supported by the DUT.
- Event Broker is not supported by the DUT.
- Protocols / mqtt is not supported by the DUT.
- Protocols / mqtts is not supported by the DUT.
- Protocols / ws is not supported by the DUT.
- Protocols / wss is not supported by the DUT.

5.6.9.3 Event Service - Message Content Filter Support

Message Content Filter function support in Event Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that Event Service is supported by the DUT as defined in Section 5.6.9, otherwise all features defined in Table 5.90 will be marked as NOT SUPPORTED.
- This procedure assumes that GetEventPropertiesResponse has already been retrieved via preceding procedure described in Section 5.6.9.1, otherwise all features defined in Table 5.90 will be marked as UNDEFINED.

Discovery Procedure:

1. ONVIF Client checks Message Content Filter feature support as defined in Table 5.90.

Note: If Message Content Filter feature is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.6.9.4.

Table 5.90. Message Content Filter Feature (GetCapabilities)

Criterion Item	GetEventPropertiesResponse message	
Feature	**Supported**	**Not Supported**
Message Content Filter	***msgContentFilterDialectList*** **contains at least one item with non empty value**	***msgContentFilterDialectList*** **does not contain at least one item with non empty value**

5.6.9.4 Event Service - ONVIF Message Content Filter Dialect Support

ONVIF Message Content Filter Dialect function support in Event Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that Event Service is supported by the DUT as defined in Section 5.6.9, otherwise all features defined in Table 5.90 will be marked as NOT SUPPORTED.
- This procedure assumes that GetEventPropertiesResponse has already been retrieved via preceding procedure described in Section 5.6.9.1, otherwise all features defined in Table 5.90 will be marked as UNDEFINED.
- This procedure assumes that DUT supports Message Content Filter feature according to Section 5.6.9.3, otherwise all features defined in Table 5.91 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client checks ONVIF Message Content Filter Dialect feature support as defined in Table 5.91.

Table 5.91. ONVIF Message Content Filter Dialect (GetCapabilities)

Criterion Item	GetEventPropertiesResponse message	
Feature	**Supported**	**Not Supported**
ONVIF Message Content Filter Dialect	***msgContentFilterDialectList* contains item with value is equal to "http://www.onvif.org/ver10/tev/messageContentFilter/ItemFilter"**	***msgContentFilterDialectList* does not contain item with value is equal to "http://www.onvif.org/ver10/tev/messageContentFilter/ItemFilter"**

5.6.9.5 Event Service - Pull-Point Notification Support

Pull-Point Notification function support in Event Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that Event Service is supported by the DUT as defined in Section 5.6.9, otherwise all features defined in Table 5.92 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client checks Pull-Point Notification feature support as defined in Table 5.92.

Table 5.92. Pull-Point Notification (GetCapabilities)

Criterion Item	-	
Feature	**Supported**	**Not Supported**
Pull-Point Notification	**Mandatory**	-

5.6.10 Device IO Service

Device IO Service feature support is determined according to the following procedure.

Since the DUT does not support GetServices feature, the following Device IO Service sub-features will be defined as described below:

- Digital Inputs under Device IO Service is NOT SUPPORTED by the DUT.

Pre-requisite:

- This procedure assumes that GetCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.93.

Note: If Device IO service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.6.10.1.
- Section 5.6.10.2.
- Section 5.6.10.3.

Table 5.93. Device IO Service (GetCapabilities)

Criterion Item	GetCapabilitiesResponse	
Feature	**Supported**	**Not Supported**
Device IO Service	**Includes Extension.DeviceIO element**	**Does not include Extension.DeviceIO element**

5.6.10.1 Relay Outputs Support

Relay outputs support under Device IO Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that GetCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.2.
- This procedure assumes that Device IO Service is supported by the DUT as defined in Section 5.6.10, otherwise all features defined in Table 5.94 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.94.

Note: If the DUT does not support Relay Outputs feature, all features from Section 5.6.10.2 will be marked as unsupported. Procedure described in Section 5.6.10.2 will be skipped.

Table 5.94. Relay Outputs Support (GetCapabilities)

Criterion Item	GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Relay Outputs	**Capabilities. Extension. DeviceIO. RelayOutputs > 0**	**Capabilities. Extension. DeviceIO. RelayOutputs = 0**

5.6.10.2 Device IO Relay Output Options Support

Device IO Relay Output Options support in Device IO Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that Device IO Service is supported by the DUT as defined in Section 5.6.10, otherwise all features defined in Table 5.95 will be marked as NOT SUPPORTED.
- This procedure assumes that Relay Outputs is supported by the DUT as defined in Section 5.6.10.1, otherwise all features defined in Table 5.95 will be marked as NOT SUPPORTED.

Discovery Procedure:

1. ONVIF Client invokes **GetRelayOutputOptions** request with parameters

- RelayOutputToken skipped

2. The DUT responds with **GetRelayOutputOptionsResponse** message or SOAP 1.2 fault.
3. ONVIF Client checks features support as defined in Table 5.95.

Note: If the DUT returns no response for GetRelayOutputOptions request, then all features defined in Table 5.95 will be marked as UNDEFUNED, the following feature discovery procedures will be skipped and related features will be marked as UNDEFUNED:

- Section 5.6.10.3

Note: If Relay Outputs Options feature is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.6.10.3

Table 5.95. Device IO Service – Relay Output Options Support (GetCapabilities)

Criterion Item	GetRelayOutputOptionsResponse message	
Feature	**Supported**	**Not Supported**
Relay Output Options	**DUT returns GetRelayOutputOptions Response for step 2.**	**DUT returns any SOAP fault for step 2.**

5.6.10.3 Device IO Relay Outputs Features Support

Device IO Relay Output features support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that Device IO Service is supported by the DUT as defined in Section 5.6.10, otherwise all features defined in Table 5.96, Table 5.97, and Table 5.98 will be marked as NOT SUPPORTED.
- This procedure assumes that Relay Outputs is supported by the DUT as defined in Section 5.6.10.1, otherwise all features defined in Table 5.96, Table 5.97, and Table 5.98 will be marked as NOT SUPPORTED.
- This procedure assumes that Relay Output Options is supported by the DUT as defined in Section 5.6.10.2, otherwise all features defined in Table 5.96, Table 5.97, and Table 5.98 will be marked as NOT SUPPORTED.

- This procedure assumes that GetRelayOutputOptionsResponse has already been retrieved via preceding procedure described in Section 5.6.10.2, otherwise all features defined in Table 5.96, Table 5.97, and Table 5.98 will be marked as UNDEFINED.

Discovery Procedure:

1. ONVIF Client invokes **GetRelayOutputs** request to retrieve a list of all available relay outputs and their settings.
2. The DUT sends the **GetRelayOutputsResponse** message with parameters
   - RelayOutputs list =: *relayOutputsList*
3. For each Relay Output (*relayOutput*) from *relayOutputsList* do the following:
   3.1. ONVIF Client checks features support as defined in Table 5.96.
   3.2. If Relay Output supports Monostable Mode:
      3.2.1. ONVIF Client invokes **SetRelayOutputSettings** request with parameters
         - RelayOutput.@token := *relayOutput*.@token
         - RelayOutput.Properties.Mode := Monostable
         - RelayOutput.Properties.DelayTime := RelayOutputOptions[0].DelayTimes[0] from GetRelayOutputOptionsResponse, where RelayOutputOptions[0] is RelayOutputOptions with token = *relayOutput*.@token
         - RelayOutput.Properties.IdleState := closed
      3.2.2. The DUT responds with **SetRelayOutputSettingsResponse** message or SOAP 1.2 fault.
      3.2.3. ONVIF Client invokes **SetRelayOutputSettings** request with parameters
         - RelayOutput.@token := *relayOutput*.@token
         - RelayOutput.Properties.Mode := Monostable
         - RelayOutput.Properties.DelayTime := RelayOutputOptions[0].DelayTimes[0] from GetRelayOutputOptionsResponse, where RelayOutputOptions[0] is RelayOutputOptions with token = *relayOutput*.@token
         - RelayOutput.Properties.IdleState := open

3.2.4. The DUT responds with **SetRelayOutputSettingsResponse** message or SOAP 1.2 fault.

3.2.5. If for both steps and SOAP 1.2 fault was returned features listed in Table 5.98 will be marked as UNDEFINED for this Relay Output, otherwise ONVIF Client checks features support as defined in Table 5.97.

3.3. If Relay Output supports Bistable Mode:

3.3.1. ONVIF Client invokes **SetRelayOutputSettings** request with parameters

- RelayOutput.@token := *relayOutput*.@token
- RelayOutput.Properties.Mode := Bistable
- RelayOutput.Properties.DelayTime := *relayOutput*.Properties.DelayTime
- RelayOutput.Properties.IdleState := closed

3.3.2. The DUT responds with **SetRelayOutputSettingsResponse** message or SOAP 1.2 fault.

3.3.3. ONVIF Client invokes **SetRelayOutputSettings** request with parameters

- RelayOutput.@token := *relayOutput*.@token
- RelayOutput.Properties.Mode := Bistable
- RelayOutput.Properties.DelayTime := *relayOutput*.Properties.DelayTime
- RelayOutput.Properties.IdleState := open

3.3.4. The DUT responds with **SetRelayOutputSettingsResponse** message or SOAP 1.2 fault.

3.3.5. If for both steps and SOAP 1.2 fault was returned features listed in Table 5.98 will be marked as UNDEFINED for this Relay Output, otherwise ONVIF Client checks features support as defined in Table 5.98.

Note: If the DUT returns no RelayOutputOptions in GetRelayOutputOptionsResponse message, then all features defined in Table 5.96, Table 5.97, and Table 5.98 will be marked as UNDEFUNED.

Note: If the DUT returns no DelayTimes element in RelayOutputOptions with supporting of Monostable mode in GetRelayOutputOptionsResponse message, then all features defined in Table 5.33 will be marked as UNDEFUNED.

Note: If the DUT returns no response for SetRelayOutputSettings request, then all features defined in Table 5.97 and Table 5.98 will be marked as UNDEFUNED.

Table 5.96. Relay Outputs Features - Modes (GetCapabilities)

Criterion Item	Current Relay Output from GetRelayOutputOptionsResponse message	
Feature (for each Relay Output)	**Supported**	**Not Supported**
Monostable Mode	**RelayOutputOptions.Mode contains Monostable**	**RelayOutputOptions.Mode does not contain Monostable**
Bistable Mode	**RelayOutputOptions.Mode contains Bistable**	**RelayOutputOptions.Mode does not contain Bistable**

Table 5.97. Relay Outputs Features - Idle States - Monostable (GetCapabilities)

Criterion Item	Current Relay Output from GetRelayOutputOptionsResponse message	
Feature (for each Relay Output)	**Supported**	**Not Supported**
Monostable Mode \Closed Idle Sate	**DUT returns SetRelayOutputSettings Response at step 3.2.2**	**DUT returns SOAP fault at step 3.2.2**
Monostable Mode \Open Idle Sate	**DUT returns SetRelayOutputSettings Response at step 3.2.4**	**DUT returns SOAP fault at step 3.2.4**

Table 5.98. Relay Outputs Features - Idle States - Bistable (GetCapabilities)

Criterion Item	Current Relay Output from GetRelayOutputOptionsResponse message	
Feature (for each Relay Output)	**Supported**	**Not Supported**
Bistable Mode\Closed Idle Sate	**DUT returns SetRelayOutputSettings Response at step 3.3.2**	**DUT returns SOAP fault at step 3.3.2**
Bistable Mode\Open Idle Sate	**DUT returns SetRelayOutputSettings Response at step 3.3.4**	**DUT returns SOAP fault at step 3.3.4**

5.6.11 PTZ Service Support

PTZ Service support is defined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.99.

Note: If the DUT does not support PTZ Service, all features from Section 5.6.10.1, Section 5.6.10.2, and Section 5.6.10.3 will be marked as unsupported. Procedure described in Section 5.6.10.1, Section 5.6.10.2, and Section 5.6.10.3 will be skipped.

Table 5.99. PTZ Service (GetCapabilities)

Criterion Item	GetCapabilitiesResponse	
Feature	**Supported**	**Not Supported**
PTZ Service	**Includes Capabilities.PTZ element**	**Does not include Capabilities.PTZ element**

5.6.11.1 PTZ Service Features Support

Since the DUT does not support GetServices feature the following PTZ features will be defined as not supported:

- Get Compatible Configurations
- Move Status
- Status Position

5.6.11.2 PTZ Nodes Features Support

PTZ Nodes features support is defined according to the following procedure.

Discovery Procedure:

1. ONVIF Client invokes GetNodes request to retrieve PTZ Nodes capabilities.
2. The DUT returns GetNodesResponse message with PTZ Nodes capabilities.

3. For each PTZ Node ONVIF Client checks features support as defined in Table 5.100.

4. For each PTZ Node with Home Position support ONVIF Client checks features support as defined in Section 5.6.10.3. For all others PTZ Nodes procedure described in Section 5.6.10.3 will be skipped and all features defined in Section 5.6.10.3 will be marked as unsupported.

Note: If the DUT returns no response for GetNodes request or the DUT returns GetNodesResponse message with empty PTZNode list, then all features defined in Table 5.100 and Section 5.6.10.3 will be marked as undefined.

Table 5.100. PTZ Nodes Features (GetCapabilities)

Criterion Item	Current PTZNode from GetNodesResponse message	
Feature (for each PTZ Node)	**Supported**	**Not Supported**
Continuous Pan/Tilt movement	**SupportedPTZSpaces. ContinuousPanTiltVelocity Space element is present**	**SupportedPTZSpaces. ContinuousPanTiltVelocity Space element is not present**
Continuous Zoom movement	**SupportedPTZSpaces. ContinuousZoomVelocity Space element is present**	**SupportedPTZSpaces. ContinuousZoomVelocity Space element is not present**
Continuous movement	**Mandatory**	**-**
Absolute Pan/Tilt movement	**SupportedPTZSpaces. AbsolutePanTiltPositionSpace element is present**	**SupportedPTZSpaces. AbsolutePanTiltPositionSpace element is not present**
Absolute Zoom movement	**SupportedPTZSpaces. AbsoluteZoomPositionSpace element is present**	**SupportedPTZSpaces. AbsoluteZoomPositionSpace element is not present**
Absolute movement	**Absolute Pan/Tilt movement or Absolute Zoom movement is supported**	**Absolute Pan/Tilt movement and Absolute Zoom movement is not supported**
Relative Pan/Tilt movement	**SupportedPTZSpaces. RelativePanTiltTranlation Space element is present**	**SupportedPTZSpaces. RelativePanTiltTranlationSpace element is not present**
Relative Zoom movement	**SupportedPTZSpaces. RelativeZoomTranlationSpace element is present**	**SupportedPTZSpaces. RelativeZoomTranlationSpace element is not present**
Relative movement	**Relative Pan/Tilt movement or Relative Zoom movement is supported**	**Relative Pan/Tilt movement and Relative Zoom movement is not supported**

Criterion Item	Current PTZNode from GetNodesResponse message	
Feature (for each PTZ Node)	**Supported**	**Not Supported**
Speed configuration	**Speed configuration function for Pan/Tilt movement or Speed configuration function for Zoom movement is supported**	**>Speed configuration function for Pan/Tilt movement and Speed configuration function for Zoom movement is not supported**
Speed configuration function for Pan/Tilt movement	**SupportedPTZSpaces. PanTiltSpeedSpace element is present**	**SupportedPTZSpaces. PanTiltSpeedSpace element is not present**
Speed configuration function for Zoom movement	**SupportedPTZSpaces. ZoomSpeedSpace element is present**	**SupportedPTZSpaces. ZoomSpeedSpace element is not present**
Preset position	**MaximumNumberOfPresets > 0**	**MaximumNumberOfPresets = 0**
Auxiliary operation	**AuxiliaryCommands element is present**	**AuxiliaryCommands element is not present**
Home Position	**HomeSupported = true**	**HomeSupported = false**

5.6.11.3 Fixed/Configurable Home Position Support for PTZ Node

In case the PTZ Node supports Home Position function, the PTZ Node shall support either Fixed or Configurable Home Position. The following defines the discovery procedure to determine which Home Position function is supported by the PTZ Node.

Pre-requisite:

- This procedure assumes that PTZ Node was recieved in GetNodesResponse message via preceding procedure described in Section 5.6.10.2.

Discovery Procedure:

1. If FixedHomePosition attribute is defined in GetNodesResponse message for this PTZ Node, ONVIF Client checks features support as defined in Table 5.101 and skips other steps of this procedure.
2. ONVIF Client invokes GetConfigurations request to retrieve a PTZ Configurations list.
3. The DUT returns GetConfigurationsResponse message with the list of PTZConfiguration that contains PTZNode. ONVIF Client identifies first PTZConfiguration which has the corresponding current PTZ Node.

4. If DUT supports Media Service according to Section 5.6.6, ONVIF Client either selects or creates Media Profile anew along with the identified PTZConfiguration (refer to Annex A.1 for the details) and goes to the step 6.

5. Otherwise Configurable Home Position and Fixed Home Position features will be marked as undefined.

6. ONVIF Client invokes SetHomePosition request (ProfileToken = selected or newly created profile token) message to check Configurable Home Position is supported by DUT.

7. ONVIF Client checks features support as defined in Table 5.102.

8. ONVIF Client restores Media Profiles setting in case it changes some of the Media Profiles configuration.

Table 5.101. Fixed/Configurable Home Position Support with FixedHomePosition Attribute (GetServices)

Criterion Item	Current PTZNode from GetNodesResponse message	
Feature (for each PTZ Node with Home Position support)	**Supported**	**Not Supported**
Configurable Home Position	**FixedHomePosition = false**	**FixedHomePosition = true**
Fixed Home Position	**FixedHomePosition = true**	**FixedHomePosition = false**

Table 5.102. Fixed/Configurable Home Position Support without FixedHomePosition Attribute (GetServices)

Criterion Item	SetHomePositionResponse	
Feature (for each PTZ Node with Home Position support)	**Supported**	**Not Supported**
Configurable Home Position	**DUT returns SetHomePositionResponse**	**DUT returns SOAP fault**
Fixed Home Position	**DUT returns SOAP fault**	**DUT returns SetHomePositionResponse**

5.6.12 Imaging Service Support

Imaging Service feature support is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.103.

Note: If Imaging service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.6.12.1.
- Section 5.6.12.3.

Table 5.103. Imaging Service (GetCapabilities)

Criterion Item	GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Imaging Service	**Includes Capabilities.Imaging element**	**Does not include Capabilities.Imaging element**

5.6.12.1 IrCutfilterConfiguration Feature Support

IrCutfilterConfiguration function support in Imaging Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that Imaging Service is supported by the DUT as defined in Seection 5.6.12, otherwise all features defined in Table 5.104 will be marked as NOT SUPPORTED.
- This procedure assumes that Device IO Service is supported by the DUT as defined in Section 5.6.10 or Media Service is supported by the DUT as defined in Section 5.6.6, otherwise all features defined in Table 5.104 will be marked as UNDEFUNED.

Discovery Procedure:

1. ONVIF Client retrieves a list of Video Sources by following the procedure mentioned in Annex A.6 with the following input and output parameters
   - out *videoSorceTokenList* - a list of Video Source tokens
2. For each Video Source token *videoSorceToken* in *videoSorceTokenList*

   2.1. ONVIF Client invokes **GetOptions** with parameters

- VideoSourceToken =: *videoSorceToken*

2.2. The DUT responds with **env:Receiver/ter:ActionNotSupported/ter:NoImagingForSource** SOAP 1.2 fault or with **GetOptionsResponse** with parameters

- ImagingOptions =: *imagingOptions*

2.3. If *imagingOptions*.IrCutFilterModes list contains at least two items and one of them is equal to OFF, skip other steps.

3. ONVIF Client checks features support as defined in Table 5.104.

Note: If the DUT does not return **env:Receiver/ter:ActionNotSupported/ter:NoImagingForSource** SOAP 1.2 fault or **GetMoveOptionsResponse**, then all features defined in Table 5.104 will be marked as UNDEFUNED.

Note: If the DUT does not return *videoSorceTokenList* list or *videoSorceTokenList* list is empty, then all features defined in Table 5.104 will be marked as UNDEFUNED.

Table 5.104. IrCutfilter Configuration Function Support in Imaging Service (GetCapabilities)

Criterion Item	GetOptionsResponse message	
Feature	**Supported**	**Not Supported**
IrCutfilter Configuration	**For at least one GetOptionsResponse message at least two IrCutFilterModes elements are present in GetOptionsResponse. ImagingOptions and one of them equal to OFF**	**There are no GetOptionsResponse messages with at least two IrCutFilterModes elements in GetOptionsResponse. ImagingOptions with one of them equal to OFF**

5.6.12.2 Imaging Events Support

Imaging Events support under Imaging Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetEventPropertiesResponse has already been retrieved via preceding procedure described in Section 5.6.9, otherwise all features defined in Table 5.105 will be marked as UNDEFINED.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.105.

Table 5.105. Imaging Events Support (GetCapabilities)

Criterion Item	GetEventPropertiesResponse	
Feature	**Supported**	**Not Supported**
Image Too Blurry	**Contains tns1:VideoSource/ ImageTooBlurry/ ImagingService or tns1:VideoSource/ ImageTooBlurry/ AnalyticsService or tns1:VideoSource/ ImageTooBlurry/ RecordingService Event topic**	**Does not contain tns1:VideoSource/ ImageTooBlurry/ ImagingService and tns1:VideoSource/ ImageTooBlurry/ AnalyticsService and tns1:VideoSource/ ImageTooBlurry/ RecordingService Event topic**
Image Too Dark	**Contains tns1:VideoSource/ ImageTooDark/ ImagingService or tns1:VideoSource/ ImageTooDark/ AnalyticsService or tns1:VideoSource/ ImageTooDark/ RecordingService Event topic**	**Does not contain tns1:VideoSource/ ImageTooDark/ ImagingService and tns1:VideoSource/ ImageTooDark/ AnalyticsService and tns1:VideoSource/ ImageTooDark/ RecordingService Event topic**
Image Too Bright	**Contains tns1:VideoSource/ ImageTooBright/ ImagingService or tns1:VideoSource/ ImageTooBright/ AnalyticsService or tns1:VideoSource/ ImageTooBright/ RecordingService Event topic**	**Does not contain tns1:VideoSource/ ImageTooBright/ ImagingService and tns1:VideoSource/ ImageTooBright/ AnalyticsService and tns1:VideoSource/ ImageTooBright/ RecordingService Event topic**
Global Scene Change	**Contains tns1:VideoSource/ GlobalSceneChange/ ImagingService or tns1:VideoSource/**	**Does not contain tns1:VideoSource/ GlobalSceneChange/ ImagingService and**

Criterion Item	GetEventPropertiesResponse	
Feature	**Supported**	**Not Supported**
	GlobalSceneChange/ AnalyticsService or tns1:VideoSource/ GlobalSceneChange/ RecordingService Event topic	**tns1:VideoSource/ GlobalSceneChange/ AnalyticsService and tns1:VideoSource/ GlobalSceneChange/ RecordingService Event topic**
Motion Alarm	**Contains tns1:VideoSource/ MotionAlarm event topic**	**Does not contain tns1:VideoSource/ MotionAlarm event topic**

5.6.12.3 Focus Control Function Support

Focus Control function support in Imaging Service is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that Imaging Service is supported by the DUT as defined in Seection 5.6.12, otherwise all features defined in Table 5.106 will be marked as NOT SUPPORTED.
- This procedure assumes that Device IO Service is supported by the DUT as defined in Section 5.6.10 or Media Service is supported by the DUT as defined in Section 5.6.6, otherwise all features defined in Table 5.106 will be marked as UNDEFUNED.

Discovery Procedure:

1. ONVIF Client retrieves a list of Video Sources by following the procedure mentioned in Annex A.6 with the following input and output parameters
   - out *videoSorceTokenList* - a list of Video Source tokens
2. For each Video Source token *videoSorceToken* in *videoSorceTokenList*
   2.1. ONVIF Client invokes **GetMoveOptions** with parameters
   - VideoSourceToken =: *videoSorceToken*

   2.2. The DUT responds with **env:Receiver/ter:ActionNotSupported/ ter:NoImagingForSource** SOAP 1.2 fault or with **GetMoveOptionsResponse** with parameters
   - MoveOptions =: *moveOptions*

2.3. If *moveOptions* contains **MoveOptions/Absolute** or **MoveOptions/Relative** or **MoveOptions/Continuous**, skip other steps.

3. ONVIF Client checks features support as defined in Table 5.106.

Note: If the DUT does not return **env:Receiver/ter:ActionNotSupported/ter:NoImagingForSource** SOAP 1.2 fault or **GetMoveOptionsResponse**, then all features defined in Table 5.106 will be marked as UNDEFUNED.

Note: If the DUT does not return *videoSorceTokenList* list or *videoSorceTokenList* list is empty, then all features defined in Table 5.106 will be marked as UNDEFUNED.

Table 5.106. Focus Control (GetCapabilities)

Criterion Item	GetMoveOptionsResponse	
Feature	**Supported**	**Not Supported**
Focus Control	**Contains MoveOptions \Absolute or MoveOptions\Relative or MoveOptions\Continuous**	**Does not contain MoveOptions\Absolute and MoveOptions\Relative and MoveOptions\Continuous**

5.6.13 Analytics Service Support

Analytics Service feature support is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.107.

Note: If Analytics service is not supported, the following feature discovery (Rule Engine features support) will be skipped.

Table 5.107. Analytics Service (GetCapabilities)

Criterion Item	GetCapabilitiesResponse	
Feature	**Supported**	**Not Supported**
Analytics Service	**Includes Capabilities.Analytics element**	**Does not include Capabilities.Analytics element**

Criterion Item	GetCapabilitiesResponse	
Feature	**Supported**	**Not Supported**
Rule Engine	**Analytics.RuleSupport = true**	**Analytics.RuleSupport = false**
Rule Options	-	**Not supported**
Analytics Modules	-	**Not supported**
Analytics Module Options	-	**Not supported**
Supported Metadata	-	**Not supported**
Image Sending	-	**Not supported**
Embedded Image Sending Type	-	**Not supported**
Local Storage Image Sending Type	-	**Not supported**
Remote Storage Image Sending Type	-	**Not supported**

5.6.13.1 Supported Rules

Since the DUT does not support Media2 Service feature the following rules will be defined as NOT SUPPORTED:

- Motion Region Detector Rule
- Face Recognition Rule
- License Plate Recognition Rule

5.6.13.2 Analytics Service - Supported Metadata

Since GetCapabilitiesResponse does not have SupportedMetadata element, the following Supported Metadata features will be defined as NOT SUPPORTED:

- Object Classification
- Vehicle Info
- Human Face
- License Plate Info
- Geo Location

5.6.14 Recording Control Service Support

Recording Control Service feature support is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.108.

Note: If Recording Control service is not supported, the following feature discovery (Dynamic Recordings, Dynamic Tracks, Recording Options, Audio Recording, JPEG, H.264, and MPEG4 features support) will be skipped.

Table 5.108. Recording Control Service (GetCapabilities)

Criterion Item	GetCapabilitiesResponse	
Feature	**Supported**	**Not Supported**
Recording Control Service	**Includes Capabilities.Extension. Recording**	**Does not include Capabilities.Extension. Recording**
Dynamic Recordings	**Extension.Recording. DynamicRecordings = true**	**Extension.Recording. DynamicRecordings = false**
Dynamic Tracks	**Extension.Recording. DynamicTracks = true**	**Extension.Recording. DynamicTracks = false**
Audio Recording	**-**	**Not supported**
Recording Options	**-**	**Not supported**
tns1:RecordingConfig/ DeleteTrackData	**GetEventProperties contains tns1:RecordingConfig/ DeleteTrackData topic**	**GetEventProperties doesn't contain tns1:RecordingConfig/ DeleteTrackData topic**
JPEG	**-**	**Not supported**
H.264	**-**	**Not supported**
MPEG4	**-**	**Not supported**

5.6.15 Recording Search Service Support

Recording Search Service feature support is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.109.
2. If Recording Search service is supported by the DUT:
   2.1. ONVIF Client invokes **GetRecordingInformation** requiest with parameters
      - RecordingToken := *recordingToken*
   2.2. The DUT responds with **GetRecordingInformationResponse** message with parameters
      - RecordingInformation =: *recordingInformation*

Note: If Recording Search service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as UNDEFINED (if Recording Control service is supported by the DUT) or NOT SUPPORTED (if Recording Control service is not supported by the DUT):

- Section 5.6.15.1.

Note: If Recording Search service is not supported by the DUT, the following feature discovery procedures will be skipped and related features will be marked as NOT SUPPORTED:

- Section 5.6.15.2.

Table 5.109. Recording Search Service (GetCapabilities)

Criterion Item	GetCapabilitiesResponse message	
Feature	**Supported**	**Not Supported**
Recording Search Service	**Includes Capabilities.Extension.Search**	**Does not include Capabilities.Extension.Search**
Metadata Search	**Capabilities.Extension. Search.MetadataSearch = true**	**Capabilities.Extension.Search. MetadataSearch = false**

5.6.15.1 Metadata Recording Support

Metadata Recording support is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Recording Control Service Service is supported by the DUT as defined in Section 5.6.15, otherwise all features defined in Table 5.110 will be marked as NOT SUPPORTED.
- This procedure assumes that Recording Search Service Service is supported by the DUT as defined in Section 5.6.15, otherwise all features defined in Table 5.110 will be marked as UNDEFINED.
- This procedure assumes that all pre-requisite defined in Annex A.10 are fulfilled, otherwise states of the features listed in Table 5.110 could not be defined correctly.
- This procedure assumes that GetRecordingInformationResponse has already been retrieved via preceding procedure described in Section 5.6.15, otherwise all features defined in Table 5.110 will be marked as UNDEFINED.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.110.

Table 5.110. Metadata Recording Support (GetCapabilities)

Criterion Item	GetRecordingInformationResponse message	
Feature	**Supported**	**Not Supported**
Metadata Recording	**Contains at least one Track with TrackType = "Metadata" and DataFrom is less than DataTo for this Track**	**Does not contain at any Track with TrackType = "Metadata" or for all Tracks with TrackType = "Metadata" DataFrom is not less than DataTo for this Track**

5.6.15.2 PTZ Position Search Support

PTZ Position Search support under Recording Search Service is determined according to the following procedure in conjunction with the above procedure.

Pre-requisite:

- This procedure assumes that Recording Search Service Service is supported by the DUT as defined in Section 5.6.15, otherwise all features defined in Table 5.111 will be marked as NOT SUPPORTED.
- This procedure assumes that all pre-requisite defined in Annex A.10 are fulfilled, otherwise states of the features listed in Table 5.111 could not be defined correctly.

- This procedure assumes that GetRecordingInformationResponse has already been retrieved via preceding procedure described in Section 5.6.15, otherwise all features defined in Table 5.111 will be marked as UNDEFINED.

Discovery Procedure:

1. ONVIF Client invokes **FindPTZPosition** requiest with parameters
   - If *recordingInformation*.EarliestRecording is specified:
     - StartPoint := *recordingInformation*.EarliestRecording
   - otherwise:
     - StartPoint := minimal value of DataFrom element in *recordingInformation*.Track list
   - If *recordingInformation*.EarliestRecording is specified:
     - EndPoint := *recordingInformation*.LatestRecording
   - otherwise:
     - EndPoint is skipped
   - Scope is empty element
   - SearchFilter.MinPosition.PanTilt.x := -1
   - SearchFilter.MinPosition.PanTilt.y := -1
   - SearchFilter.MinPosition.Zoom is skipped
   - SearchFilter.MaxPosition.PanTilt.x := 1
   - SearchFilter.MaxPosition.PanTilt.y := 1
   - SearchFilter.MaxPosition.Zoom is skipped
   - SearchFilter.EnterOrExit := false
   - MaxMatches is skipped
   - KeepAliveTime := "PT3S"
2. The DUT responds with SOAP 1.2 fault or **FindPTZPositionResponse** message with parameters
   - SearchToken =: *searchToken*

3. ONVIF Client checks features support as defined in Table 5.111.
4. If the DUT returns **FindPTZPositionResponse** message:
   - ONVIF Client invokes **EndSearch** requiest with parameters
     - SearchToken := *searchToken*
   - The DUT responds with **EndSearchResponse** message with parameters
     - Endpoint

Note: *recordingToken* will be taken from 'Recording from tests' field of ONVIF Device Test Tool.

Note: If the DUT does not return *recordingInformation*.EarliestRecording and there are no *recordingInformation*.Track items then all features defined in Table 5.111 will be marked as UNDEFINED.

Table 5.111. PTZ Position Search Support (GetCapabilities)

Criterion Item	FindPTZPositionResponse message	
Feature	**Supported**	**Not Supported**
PTZ Position Search	**DUT returns FindPTZPositionResponse**	**DUT returns SOAP 1.2 fault**

5.6.16 Replay Service Support

Replay Service feature support is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.112.

Note: If Replay service is not supported, the following feature discovery (Reverse Replay features support) will be skipped.

Table 5.112. Replay Service (GetCapabilities)

Criterion Item	GetCapabilitiesResponse	
Feature	**Supported**	**Not Supported**
Replay Service	**Includes Capabilities.Extension.Replay element**	**Does not include Capabilities.Extension.Replay element**
Reverse Replay	**-**	**Not Supported**
RTP/RTSP/TCP	**-**	**Not Supported**

5.6.17 Receiver Service Support

Receiver Service feature support is determined according to the following procedure.

Pre-requisite:

- This procedure assumes that GetCapabilitiesResponse has already been retrieved via preceding procedure described in Section 5.2.

Discovery Procedure:

1. ONVIF Client checks features support as defined in Table 5.113.

Table 5.113. Receiver Service (GetCapabilities)

Criterion Item	GetCapabilitiesResponse	
Feature	**Supported**	**Not Supported**
Receiver Service	**Includes Capabilities.Extension. Receiver element**	**Does not include Capabilities.Extension.Receiver element**

5.6.18 Door Control Service Support

Since the DUT does not support GetServices feature Door Control Service feature will be defined as not supported.

5.6.19 Access Control Service Support

Since the DUT does not support GetServices feature Access Control Service feature will be defined as not supported.

5.6.20 Security Configuration Service Support

Since the DUT does not support GetServices feature Security Configuration Service feature will be defined as not supported.

5.6.21 Credential Service Support

Since the DUT does not support GetServices feature Credential Service feature will be defined as not supported.

5.6.22 Access Rules Service Support

Since the DUT does not support GetServices feature Access Rules Service feature will be defined as not supported.

5.6.23 Schedule Service Support

Since the DUT does not support GetServices feature Schedule Service feature will be defined as not supported.

5.6.24 Provisioning Service Support

Since the DUT does not support GetServices feature Provisioning Service feature will be defined as not supported.

5.6.25 Thermal Service Support

Since the DUT does not support GetServices feature Thermal Service feature will be defined as not supported.

5.7 Devices Scopes Retrieval via GetDeviceScopes

Device scopes provided via GetDeviceScopes may indicate referenced Profiles by the DUT in case the DUT supports a certain profile. The following is the procedure to identify referenced Profile(s) by the DUT. For the details on how the retrieved scope will be used for Profile checking, refer to the specific document.

Discovery Procedure:

1. ONVIF Client invokes GetScopesRequest message to retrieve a device scope list.

2. The DUT responds with GetScopesResponse message with parameters
   - Scopes list =: *scopesList*
3. ONVIF Client preserves the device scope list to determine the referenced Profiles.

Note: If the DUT does not return GetScopesResponse, ONVIF Client regards that the device scope is defined as empty.

5.8 Devices Information Retrieval via GetDeviceInformation

General device information provided via GetDeviceScopes is required for report generation. The following is the procedure to identify DUT information.

Discovery Procedure:

1. ONVIF Client invokes GetDeviceInformationRequest message to retrieve device information.
2. ONVIF Client preserves the device information.

Note: If the DUT does not return GetDeviceInformationResponse, ONVIF Client regards the device information as undefined.

6 Pre-Configuration Procedure

This section describes procedure that is allowed to prepare the DUT for confoemance test execution.

6.1 General Policy

The results of pre-configuration procedure shall have no impact on conformance results, but in the case of the failure some test case execution will require more time or manual pre-configuration, before conformance will be started.

ONVIF Client goes frought each preconfiguration procedure described this document and runs it depending on DUT features according procedure pre-requisites.

6.2 IPv6 Enabling Pre-Configuration Procedure

ONVIF Client configures IPv6 address according to the following procedure.

Pre-requisite:

- IPv6 is supported by DUT.

Pre-Configuration Procedure:

1. ONVIF Client invokes **GetNetworkInterfaces** request.
2. The DUT responds with **GetNetworkInterfacesResponse** message with parameters
   - NetworkInterfaces list =: *networkInterfacesList*
3. Set *currentNetworkInterface* := network interface from *networkInterfacesList*, which is used for the DUT conformance.
4. If *currentNetworkInterface*.IPv6.Enabled = true, skip other steps of procedure.
5. ONVIF Client invokes **SetNetworkInterfaces** request with parameters
   - InterfaceToken := *currentNetworkInterface*.
   - NetworkInterface.Enabled := true
   - NetworkInterface.Link is skipped
   - NetworkInterface.MTU is skipped
   - NetworkInterface.IPv4 is skipped

- NetworkInterface.IPv6.Enabled := true
- NetworkInterface.Extension is skipped

6. The DUT responds with **SetNetworkInterfacesResponse** message with parameters
   - RebootNeeded =: *rebootNeededFlag*
7. If *rebootNeededFlag* = true:
   7.1. ONVIF Client invokes **SystemReboot** request.
   7.2. The DUT responds with **SystemRebootResponse** message with parameters
      - Message
8. ONVIF Client waits for HELLO message from the default network interface.

Note: In the case of failure on any step the procedure will skip other steps.

Annex A Helper Procedures and Additional Notes

A.1 Selection/Creation of Media Profile That Contains PTZ Configuration

Name: HelperSelectionCreationOfMediaProfileThatContainsPTZConfiguration

Procedure Purpose: Helper procedure to select or create Media Profile with PTZConfiguration.

Pre-requisite: Media Service is received from the DUT. PTZ Service is received from the DUT.

Input: Token of the PTZ Node, with which Media Profile should be configured (*ptzNodeToken*).

Returns: Media Profile (*profile*) with PTZ Configuration.

Procedure:

1. Retrieve media profiles by invoking **GetProfiles** request.
2. The DUT responds with **GetProfilesResponse** message with parameters
   - Profile list =: *profileList*
3. If *profileList* contains profile (*ptzProfile*) with PTZ configuration with NodeToken = *ptzNodeToken* and with Video Source Configuration and with Video Encoder Configuration:
   - Set *profile* := *ptzProfile*
   - Skip other steps.
4. If DUT does not support Get Compatible Configurations feature:
   - If no media profile contains identified PTZConfiguration, select one media profile whose fixed attribute is set to false and which already adds VideoSourceConfiguration and VideoEncoderConfiguration. Add PTZConfiguration to the media profile by invoking AddPTZConfiguration command.
   - If no media profile is present to meet the above condition, create new media profile with VideoSourceConfiguration and VideoEncoderConfiguration by invoking CreateProfile, AddVideoSourceConfiguration and AddVideoEncoderConfiguration command. After that, add PTZConfiguration to the media profile by invoking AddPTZConfiguration command.
5. If DUT supports Get Compatible Configurations feature:
   - ONVIF Client configures an empty Media Profile by following the procedure mentioned in Annex A.11 with the following input and output parameters

- in *profileList* - Media Profile List
- out *profile* - Media Profile

- ONVIF Client invokes **GetCompatibleVideoSourceConfigurations** request with parameters
  - ProfileToken := *profile*.@token
- DUT responds with **GetCompatibleVideoSourceConfigurationsResponse** message with parameters
  - Configurations list := *vscList*
- If *vscList* is empty, FAIL the test and skip other steps.
- For each Video Source Configuration *vsc* in *vscList* repeat the following steps:

5.1. ONVIF Client invokes **AddVideoSourceConfiguration** request with parameters
  - ProfileToken := *profile*.@token
  - ConfigurationToken := *vsc*.@token

5.2. DUT responds with **AddVideoSourceConfigurationResponse** message.

5.3. ONVIF Client invokes **GetCompatibleVideoEncoderConfigurations** request with parameters
  - ProfileToken := *profile*.@token

5.4. DUT responds with **GetCompatibleVideoEncoderConfigurationsResponse** message with parameters
  - Configurations list := *vecList*

5.5. If *vecList* is empty, FAIL the test and skip other steps.

5.6. ONVIF Client invokes **AddVideoEncoderConfiguration** request with parameters
  - ProfileToken := *profile*.@token
  - ConfigurationToken := *vecList*[0].@token

5.7. DUT responds with **AddVideoEncoderConfigurationResponse** message.

5.8. ONVIF Client invokes **GetCompatibleConfigurations** request with parameters

- ProfileToken := *profile*.@token

5.9. DUT responds with **GetCompatibleConfigurationsResponse** message with parameters

- PTZ Configurations list := *ptzConfigList*

5.10. If *ptzConfigList* contains PTZConfiguration item (*ptzConfig*) with NodeToken = *ptzNodeToken*:

- ONVIF Client invokes **AddPTZConfiguration** request with parameters
  - ProfileToken := *profile*.@token
  - ConfigurationToken := *ptzConfig*.@token
- DUT responds with **AddPTZConfigurationResponse** message.
- Skip other steps.

5.11. ONVIF Client invokes **RemoveVideoEncoderConfiguration** request with parameters

- ProfileToken := *profile*.@token

5.12. DUT responds with **RemoveVideoEncoderConfigurationResponse** message.

5.13. If *ptzConfigList* does not contain PTZConfiguration item with NodeToken = *ptzNodeToken* for all Video Source Configurations from *videoSourceConfigurationList1*, FAIL the test and skip other steps.

Procedure Result:

PASS –

- DUT passes all assertions.

FAIL –

- DUT did not send **GetProfilesResponse** message.
- DUT did not send **GetCompatibleConfigurationsResponse** message.
- DUT did not send **GetCompatibleVideoSourceConfigurationsResponse** message.
- DUT did not send **GetCompatibleVideoEncoderConfigurationsResponse** message.
- DUT did not send **AddVideoSourceConfigurationResponse** message.

- DUT did not send **AddVideoEncoderConfigurationResponse** message.
- DUT did not send **RemoveVideoEncoderConfigurationResponse** message.
- DUT did not send **AddPTZConfigurationResponse** message.

A.2 Get Complete Door Info List

The following algorithm will be used to get a complete list of Doors:

1. ONVIF Client will invoke GetDoorInfoListRequest message (no Limit, no StartReference) to retrieve the first part of Door Information list from the DUT.
2. Verify the GetDoorInfoListResponse message from the DUT.
3. If GetDoorInfoListResponse message contains NextStartReference, repeat steps 1-2 with StartReference = [current NextStartReference]. Otherwise, skip other steps and finalize getting complete door list.

The complete ordered list of doors with information will be made by the means of uniting all GetDoorInfoListResponse messages. Also, the total number of doors will be calculated.

A.3 Get Complete Access Point Info List

Name: HelperGetCompleteAccessPointInfoList

Procedure Purpose: Helper procedure to retrieve complete access points info list.

Pre-requisite: Access Control Service was received from the DUT.

Input: None

Returns: Complete access points info list (*accessPointInfoCompleteList*). Number of access points (*accessPointsNumber*).

Procedure:

1. ONVIF client invokes **GetAccessPointInfoList** with parameters
   - Limit is skipped
   - StartReference is skipped
2. The DUT responds with **GetAccessPointInfoListResponse** message with parameters
   - NextStartReference =: *nextStartReference*

- AccessPointInfo list =: *accessPointInfoCompleteList*

3. Until *nextStartReference* is not null, repeat the following steps:

   3.1. ONVIF client invokes **GetAccessPointInfoList** with parameters

   - Limit is skipped
   - StartReference := *nextStartReference*

   3.2. The DUT responds with **GetAccessPointInfoListResponse** message with parameters

   - NextStartReference =: *nextStartReference*
   - AccessPointInfo list =: *accessPointInfoListPart*

   3.3. Set *accessPointInfoCompleteList* := *accessPointInfoCompleteList* + *accessPointInfoListPart*.

4. Set *accessPointsNumber* := number of AccessPointInfo items in *accessPointInfoCompleteList*.

Procedure Result:

PASS –

- The DUT passed all assertions.

FAIL –

- The DUT did not send **GetAccessPointInfoListResponse** message.

A.4 Get Complete Area Info List

Name: HelperGetCompleteAreaInfoList

Procedure Purpose: Helper procedure to retrieve complete areas info list.

Pre-requisite: Access Control Service was received from the DUT.

Input: None

Returns: Complete areas info list (*areaInfoCompleteList*). Number of areas (*areaNumber*).

Procedure:

1. ONVIF client invokes **GetAreaInfoList** with parameters
   - Limit is skipped
   - StartReference is skipped
2. The DUT responds with **GetAreaInfoListResponse** message with parameters
   - NextStartReference =: *nextStartReference*
   - AreaInfo list =: *areaInfoCompleteList*
3. Until *nextStartReference* is not null, repeat the following steps:
   3.1. ONVIF client invokes **GetAreaInfoList** with parameters
      - Limit is skipped
      - StartReference := *nextStartReference*
   3.2. The DUT responds with **GetAreaInfoListResponse** message with parameters
      - NextStartReference =: *nextStartReference*
      - AreaInfo list =: *areaInfoListPart*
   3.3. Set *areaInfoCompleteList* := *areaInfoCompleteList* + *areaInfoListPart*.
4. Set *areasNumber* := number of AreaInfo items in *areaInfoCompleteList*.

Procedure Result:

PASS –

- The DUT passed all assertions.

FAIL –

- The DUT did not send **GetAreaInfoListResponse** message.

A.5 Get Analytics Configurations List

Name: HelperGetAnalyticsConfigurationsList

Procedure Purpose: Helper procedure to retrieve Analytics Configurations List.

Pre-requisite: Media2 Service is received from the DUT.

Input: None.

Returns: Analytics Configurations list (*analyticsConfList*).

Procedure:

1. ONVIF Client invokes **GetAnalyticsConfigurations** request with parameters
   - ConfigurationToken skipped
   - ProfileToken skipped
2. The DUT responds with **GetAnalyticsConfigurationsResponse** with parameters
   - Configurations list =: *analyticsConfList*
3. If *analyticsConfList* is empty, FAIL the test.

Procedure Result:

PASS –

- DUT passes all assertions.

FAIL –

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

A.6 Get Token List of Video Sources

Name: HelperGetVideoSourceTokensList

Procedure Purpose: Helper procedure to retrieve Video Sources List.

Pre-requisite: DeviceIO Service or Media2 Service is received from the DUT.

Input: None.

Returns: Video Source Token list (*videoSorceTokenList*).

1. If DUT supports DeviceIO Service:
   1.1. ONVIF Client invokes **GetVideoSources** request for DeviceIO service.
   1.2. The DUT responds with **GetVideoSourcesResponse** with parameters
      - Token list =: *videoSorceTokenList*
   1.3. If *videoSorceTokenList* is empty, FAIL the test.
   1.4. Skip other steps and return *videoSorceTokenList* in test procedure.

2. If DUT supports Media Service:

    2.1. ONVIF Client invokes **GetVideoSources** request for Media service.

    2.2. The DUT responds with **GetVideoSourcesResponse** with parameters

    - VideoSources list =: *videoSorceList*

    2.3. If *videoSorceList* is empty, FAIL the test.

    2.4. For each Video Source *videoSorce* in *videoSorceList*

        2.4.1. Set *videoSorceTokenList* := *videoSorceTokenList* + *videoSorce*.token

    2.5. Skip other steps and return *videoSorceTokenList* in test procedure.

Procedure Result:

PASS –

- DUT passes all assertions.

FAIL –

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

A.7 Get Video Source Configurations List

Name: HelperGetVideoSourceConfigurationsList

Procedure Purpose: Helper procedure to retrieve Video Source Configurations List.

Pre-requisite: Media2 Service is received from the DUT.

Input: None.

Returns: Video Source Configurations list (*videoSourceConfList*).

Procedure:

1. ONVIF Client invokes **GetVideoSourceConfigurations** request with parameters

    - ConfigurationToken skipped

    - ProfileToken skipped

2. The DUT responds with **GetVideoSourceConfigurationsResponse** with parameters

- Configurations list =: *videoSourceConfList*

3. If *videoSourceConfList* is empty, FAIL the test.

Procedure Result:

PASS –

- DUT passes all assertions.

FAIL –

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

A.8 Media Profile Configuration for PTZ Control

Name: HelperMediaProfileConfiguration

Procedure Purpose: Helper procedure to find, create or configure Media Profile with Video Source Configuration and PTZ Configuration.

Pre-requisite: Media2 Service is received from the DUT. PTZ Service is received from the DUT. GetCompatibleConfigurations is supported by Device as indicated by the GetCompatibleConfigurations = true capability.

Input: Token of the PTZ Node, with which Media Profile should be configured (*ptzNodeToken*).

Returns: Media Profile (*profile*) with Video Source Configuration and PTZ Configuration.

Procedure:

1. ONVIF Client invokes **GetProfiles** request with parameters
   - Token skipped
   - Type[0] := PTZ
   - Type[1] := VideoSource
2. The DUT responds with **GetProfilesResponse** message with parameters
   - Profiles list =: *profileList*
3. If *profileList* is empty, FAIL the test and skip other steps.
4. If *profileList* contains Media Profile, which includes Configurations.PTZ.NodeToken = *ptzNodeToken*:

4.1. Set *profile* := item from *profileList* list, which includes Configurations.PTZ.NodeToken = *ptzNodeToken*

4.2. If *profile* does not contain Configurations.VideoSource:

4.2.1. ONVIF Client adds Video Source to Media Profile by following the procedure mentioned in Annex A.9 with the following input and output parameters

- in *profile* - Media Profile
- out *profile* - Media Profile with Video Source Configuration

4.3. Skip other steps in the procedure.

5. For each profile (*profile*) with Video Source Configuration from *profileList*:

5.1. ONVIF Client invokes **GetCompatibleConfigurations** request with parameters:

- ProfileToken := *profile*.@token

5.2. The DUT responds with **GetCompatibleConfigurationsResponse** message with parameters

- PTZConfiguration list =: *ptzConfigurationList*

5.3. If *ptzConfigurationList* contains item with NodeToken = *ptzNodeToken*:

5.3.1. ONVIF Client invokes **AddConfiguration** request with parameters

- ProfileToken := *profile*.@token
- Name skipped
- Configuration[0].Type := PTZ
- Configuration[0].Token := *ptzConfiguration*.@token

5.3.2. The DUT responds with **AddConfigurationResponse** message.

5.3.3. Return *profile* and skip other steps.

6. FAIL the test and skip other steps.

Procedure Result:

PASS –

- DUT passes all assertions.

FAIL –

- DUT did not send **GetProfilesResponse** message.
- DUT did not send **GetCompatibleConfigurationsResponse** message.
- DUT did not send **AddConfigurationResponse** message.

A.9 Media Profile Configuration with Video Source Configuration

Name: HelperMediaProfileConfigurationVS

Procedure Purpose: Helper procedure to add Video Source Configuration to Media Profile.

Pre-requisite: Media2 Service is received from the DUT. PTZ Service is received from the DUT.

Input: Media Profile (*profile*).

Returns: Media Profile (*profile*) with Video Source Configuration.

Procedure:

1. ONVIF Client invokes **GetVideoSourceConfigurations** request with parameters
   - ConfigurationToken skipped
   - ProfileToken = *profile*.@token
2. The DUT responds with **GetVideoSourceConfigurationsResponse** with parameters
   - Configurations list =: *videoSourceConfigurationList*
3. If *videoSourceConfigurationList* is empty, FAIL the test and skip other steps.
4. ONVIF Client invokes **AddConfiguration** request with parameters
   - ProfileToken := *profile*.@token
   - Name skipped
   - Configuration[0].Type := VideoSource
   - Configuration[0].Token := *videoSourceConfigurationList*[0]
5. The DUT responds with **AddConfigurationResponse** message.

Procedure Result:

PASS –

- DUT passes all assertions.

FAIL –

- DUT did not send **GetVideoSourceConfigurationsResponse** message.
- DUT did not send **AddConfigurationResponse** message.

A.10 Recording Environment Pre-Requisite

The following pre-perquisite shall be filled before executing feature discovery for Replay Service, Recording Search Service, and Recording Control Service:

Procedure:

1. At least one recording shall be present at the DUT and specified for test execution on Management tab\Recording tab of ONVIF Device Test Tool.
2. The recording shall be stopped.
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.11 Configure Empty Media Profile

Name: HelperConfigureEmptyMediaProfile

Procedure Purpose: Helper procedure to create of configure an empty Media Profile (Media Service).

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

Input: Media Profile List (*profileList*).

Returns: Empty Media Profile (*profile*).

Procedure:

1. ONVIF Client invokes **CreateProfile** request with parameters
   - Name := TestName
2. DUT responds with **env:Receiver/ter:Action/ter:MaxNVTProfiles** SOAP 1.2 fault or with **CreateProfileResponse** message with parameters
   - Profile =: *profile*
3. If DUT returns **env:Receiver/ter:Action/ter:MaxNVTProfiles** SOAP 1.2 fault at step 2:
   3.1. Set *profile* := *profileList*[0]
   3.2. ONVIF Client removes all configurations from *profile*.

Procedure Result:

PASS –

- DUT passes all assertions.

FAIL –

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