ONVIF®

Profile Q Specification

Version 1.2

December 2018

REVISION HISTORY

Vers.	Date	Description
1.0	July 2016	Original release version 1.0
1.1	July 2018	[EDITORIAL UPDATE, NO TECHNICAL CHANGE] Fix of typos, clarification on relevance for clients (Clause 7 &10), caution statement on use of Factory Default State (Clause 6) and definition for 'full anonymous access' (Clause3.1) added. New ONVIF document template applied.
1.2	December 2018	[EDITORIAL UPDATE, NO TECHNICAL CHANGE] 'Advanced Security Specification' was renamed with Network Interface Specifications v18.12 to 'Security Configuration Service Specification' and respective name changes have been reflected in clause 8.3. Editorial clarification in clauses 6.1.4 and 8.1 on consistent use of term firmware upgrade (instead of update).

CONTRIBUTORS

Version 1.0

Company	Contributor
Siemens	Gero Bäse – chair (until 2015-08)
Oncam	Steven Dillingham – chair (as of 2015-09)
Pelco by Schneider Electric	Scott Hudson – editor Steve Wolf
Axis	Baldvin Gislason Bern Stefan Andersson
Bosch	Hans Busch Dirk Stegemann
Canon	Takahiro Iwasaki
Panasonic	Hirokazu Kitaoka Hasan Timucin Ozdemir
Sony	Michio Hirai Hiroyuki Sano Norio Ishibashi
VideoTec	Ottavio Campana
Vivotek	Jerry Kuo

Version 1.1

Company	Contributor
Anixter	Bob Dolan
Axis Communications AB	Johan Svensk
Bosch	André Eichhorn Hans Busch
Honeywell	Giri Guntipalli
Pelco by Schneider Electric	Marwan Obeidat
Sony Corporation	Andreas Schneider (Ed.) Hiroyuki Sano
VideoTec	Ottavio Campana

Version 1.2

Company	Contributor
Pelco by Schneider Electric	Steve Wolf
Sony Corporation	Andreas Schneider (Ed.) Hiroyuki Sano

Table of Contents

1 Scope

This document defines the mandatory and conditional features required by an ONVIF device and ONVIF client that support the Profile Q.

2 Normative references

This section defines the normative references applicable to this specification.

2.1 Normative references

- **ONVIF Profile Policy**

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

- **ONVIF Network Interface Specifications**

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

3 Terms and definitions

This section provides common terms and definitions used in this specification.

3.1 Definitions

profile	See [ONVIF Profile Policy].
ONVIF device	networked hardware appliance or software program that exposes one or multiple ONVIF Web Services
ONVIF client	networked hardware appliance or software program that uses ONVIF Web Services
out-of-the-box	unmodified configuration of a device as shipped from the manufacturer
Factory Default State	state of the Profile Q device prior to setting an administrator password NOTE: In this state, the device accepts any commands without authentication.
Operational State	state of the Profile Q device after setting an administrator password NOTE: The device requires an authentication according to its access policy to accept commands
full anonymous access	device does not require any kind of client authentication EXAMPLES for client authentication: http digest authentication, TLS client authentication
tns1	prefix for the ONVIF topic namespace NOTE: This prefix is not part of the standard and an implementation can use any prefix. See [ONVIF Network Interface Specifications] Core Specification description of Namespaces for details.

4 Technical specification version requirement

Implementation of ONVIF Network Interface Specifications, version 2.5 or later is required for conformance to Profile Q.

5 Requirement levels

Each feature in this document has a requirement level for device and client that claims conformance to Profile Q and contains a function list that states the function's requirement level for device and client that implement that feature.

The requirement levels for features are:

- **Mandatory = Feature that shall be implemented by an ONVIF device or ONVIF client.**
- **Conditional = Feature that shall be implemented by an ONVIF device or ONVIF client if it supports that functionality in any way, including any proprietary way. Features that are conditional are marked with "if supported" in a profile specification.**

The requirement levels for functions are:

- **Mandatory = Function that shall be implemented by an ONVIF device or ONVIF client.**
- **Conditional = Function that shall be implemented by an ONVIF device or ONVIF client if it supports that functionality.**
- **Optional = Function that may be implemented by an ONVIF device or ONVIF client.**

Function lists use the following abbreviations:

- **M = Mandatory**
- **C = Conditional**
- **O = Optional**

All functions shall be implemented as described in the corresponding [ONVIF Network Interface Specifications].

6 Overview

This section explains the concept of the profile.

Out-of-the-box interoperability is a key market demand for IP security devices. ONVIF is addressing this need by providing standardized interfaces to network security products. In the context of security devices ONVIF translates “out-of-the-box” interoperability to a “quick install” profile called Profile Q.

> ***Special attention when deploying an ONVIF Profile Q device:***
> *When a Profile Q conformant device is connected to an IP network for the first time, it is in the Factory Default State (see clause 9), intended for initial set-up / configuration. In this state, the device will accept any supported ONVIF commands without requiring authentication. For cybersecurity and data protection/privacy reasons, implementing protective measures is recommended while in Factory Default State. The device remains in this state until an administrator password is set. Only then, the device changes to the Profile Q Operational State requiring authentication for all ONVIF commands. Profile Q devices should be deployed in Operational State only (see clause 10).*

An ONVIF device compliant to Profile Q is an ONVIF device that can be discovered and configured by an ONVIF client.

An ONVIF client compliant to Profile Q is an ONVIF client that can discover, configure and control an ONVIF device compliant to Profile Q over an IP network.

6.1 Use cases

This section defines the anticipated typical usage of the profile

6.1.1 Installation and initial configuration

An installer has installed one or more devices on a network and wishes to discover them on the network. The installer runs an ONVIF Profile Q client that lists the devices it has found, and allows the installer to set the common settings such as IP Address, NTP, Time, authentication and security settings. It does not matter what type of device the client is communicating to, as these settings are common to all device types.

- **Easy set-up facilitates the commissioning of IP security devices, by allowing a single mode of discovery and basic configuration. (Multi-vendor / Multi-disciplines)**
- **Unified configuration tool for basic set-up across devices from different vendors and different disciplines**

6.1.2 Network discovery

A new Profile Q network switch is deployed in a local network. There are Profile Q conformant devices plugged into this switch. The switch discovers each device and records its port, IP address and its supported profiles. This is exposed in the administration User Interface of the Switch.

6.1.3 Monitoring

A Profile Q network monitor is installed on the local network and is used to discover all devices on the local network. Once the application has appropriate credentials, it then can pro-actively monitor and report any issues intelligently based on the type of device, and its profile capability. Monitoring examples include events (e.g., Fan Failure, Power Supply Failure, Storage Failure, Temperature Critical and Last Backup) and devices added to or removed from the network.

6.1.4 Firmware upgrade

A system administrator wishes to find all devices on a deployed system and generate a report of the firmware version for review. The system administrator connects a Profile Q client which discovers every device on the local network and queries for the firmware version. The resulting report is used to determine if any upgrades should be applied. The Profile Q client could then be used to selectively apply the appropriate firmware upgrades.

6.1.5 Backup and restore

A system administrator would like to either create a system configuration backup or restore an existing backup. The system administrator connects a Profile Q client that discovers every device on the local network. The Profile Q client could then be used for either creating a backup or restoring an existing backup. Several scenarios exist:

1. **Restore a backup to the same device in order to restore a previous configuration**
    a. **the device firmware version is unchanged**
    b. **the firmware version differs between backup and restore**
2. **Restore a backup to a replacement device of same type in order to have a full replacement for a broken device. Security related information may not be expected to be backed up or restored.**
3. **Restore a backup to a device of same type in order to have exactly the same configuration running on two devices. Security related information may not be expected to be backed up or restored.**

The format of the backup configuration data is vendor specific. It is expected that after completion of the restore operation, the device is working on the same configuration as that of the time the configuration was backed up. Note that the configuration of static IP addresses may differ.

Device vendors may put restrictions on the functionality to be restored. The detailed behavior is outside the scope of this specification.

6.1.6 Preparation for untrusted network

A trusted communication channel is available between a trusted security operator and the device at some point in time before the device is connected to an untrusted network. This trusted communication channel can be used to initialize the security features of the device.

After the device has been connected to the untrusted network, the security features of the device can be used to communicate to it securely for further configuration.

7 Profile mandatory features (normative)

The profile mandatory features section lists all the features that are guaranteed to be supported between a device and client that are both conformant to the profile.–

7.1 Capabilities

- **GetServices and GetServiceCapabilities are used to query a device for its capabilities.**

7.1.1 Device requirements

- **Device shall support Services and Capabilities operations as detailed in [ONVIF Network Interface Specifications].**
- **Device shall signal maximum user name and password length using MaxUsernameLength and MaxPasswordLength.**

7.1.2 Client requirements

- **Client shall determine the available Services using the GetServices operation.**

7.1.3 Capabilities function list for devices

Capabilities	**Device MANDATORY**	
Function	**Service**	**Requirement**
GetServices	Device	M

7.1.4 Capabilities function list for clients

Capabilities	**Client MANDATORY**	
Function	**Service**	**Requirement**
GetServices	Device	M

7.2 Network configuration

- **Configuration of network settings on the device**

7.2.1 Device requirements

- **Device shall support hostname, DNS, network interface, network protocol and network default gateway operations as covered by the device service.**

- **Device shall support Zero Configuration operations as covered by the device service.**
- **Device shall return Device->Network->ZeroConfiguration capability set to “true” in GetCapabilities response.**

7.2.2 Client requirements (if supported)

- **If configuring a device’s network configuration is supported in any way by the client, the client shall be able to list and configure the device network interface using the GetNetworkInterfaces and SetNetworkInterfaces operations.**
- **If configuring a device’s network configuration is supported in any way by the client, the client shall be able to list and set the default gateway of the device using the GetNetworkDefaultGateway and SetNetworkDefaultGateway operations.**
- **If configuring a device's network hostname is supported in any way by the client, the client shall be able to list and set the network hostname of the device using the GetHostname and SetHostname operations.**
- **If configuring a device's domain name server is supported in any way by the client, the client shall be able to list and set the domain name server of the device using the GetDNS and SetDNS operations.**
- **If enabling or disabling a device's HTTP, HTTPS or RTSP protocols of configuring the ports for those protocols is supported in any way by the client, the client shall be able configure those properties of the device using the GetNetworkProtocols and SetNetworkProtocols operations.**
- **If configuring a device's zero configuration is supported in any way by the client, the client shall be able to list and set zero configuration of the device using the GetZeroConfiguration and SetZeroConfiguration operations.**

7.2.3 Network configuration function list for devices

Network Configuration		**Device MANDATORY**
Function	**Service**	**Requirement**
GetHostname	Device	M
SetHostname	Device	M
GetDNS	Device	M
SetDNS	Device	M
GetNetworkInterfaces	Device	M
SetNetworkInterfaces	Device	M
GetNetworkProtocols	Device	M
SetNetworkProtocols	Device	M
GetNetworkDefaultGateway	Device	M
SetNetworkDefaultGateway	Device	M
GetZeroConfiguration	Device	M
SetZeroConfiguration	Device	M

7.2.4 Network configuration function list for clients

Network Configuration		Client CONDITIONAL
Function	**Service**	**Requirement**
GetHostname	Device	C
SetHostname	Device	C
GetDNS	Device	C
SetDNS	Device	C
GetNetworkInterfaces	Device	M
SetNetworkInterfaces	Device	M
GetNetworkProtocols	Device	C
SetNetworkProtocols	Device	C
GetNetworkDefaultGateway	Device	M
SetNetworkDefaultGateway	Device	M
GetZeroConfiguration	Device	C
SetZeroConfiguration	Device	C

7.3 System

- **Configuration of system settings.**
- **Device information.**
- **Synchronization of time using manual methods or NTP servers.**

7.3.1 Device requirements

- **Device shall support get information, date and time, NTP, factory defaults and reboot operations as covered by the device service.**

7.3.2 Client requirements (if supported)

- **Client shall be able to get device information such as manufacturer, model and firmware version using the GetDeviceInformation operation.**
- **Client shall be able to get the time of the device using the GetSystemDateAndTime operation.**
- **Client shall be able to configure time of the device using either SetNTP or SetSystemDateAndTime operations.**
- **A client that supports SetNTP shall also support GetNTP.**

7.3.3 System function list for devices

System		Device MANDATORY
Function	**Service**	**Requirement**
GetDeviceInformation	Device	M
GetSystemDateAndTime	Device	M
SetSystemDateAndTime	Device	M
GetNTP	Device	M
SetNTP	Device	M
SetSystemFactoryDefault	Device	M
Reboot	Device	M

7.3.4 System function list for clients

System		Client CONDITIONAL
Function	**Service**	**Requirement**
GetDeviceInformation	Device	M
GetSystemDateAndTime	Device	M
SetSystemDateAndTime	Device	C
GetNTP	Device	C
SetNTP	Device	C
SetSystemFactoryDefault	Device	O
Reboot	Device	O

7.4 User handling

- **Manage users on the device.**

7.4.1 Device requirements

- **Device shall support user handling operations as covered by the device service.**
- **Device shall signal the maximum number of users supported via its capabilities.**

7.4.2 Client requirements

- **Client shall be able to create, list, modify and delete users from the device using the CreateUsers, GetUsers, SetUser and DeleteUsers operations.**

7.4.3 User handling function list for devices

User Handling		Device MANDATORY
Function	**Service**	**Requirement**
GetUsers	Device	M
CreateUsers	Device	M
DeleteUsers	Device	M
SetUser	Device	M

7.4.4 User handling function list for clients

User Handling		Client MANDATORY
Function	**Service**	**Requirement**
GetUsers	Device	M
CreateUsers	Device	M
DeleteUsers	Device	M
SetUser	Device	M

7.5 Standard events for monitoring

- **Standard events enable monitoring for device operations.**

7.5.1 Device requirements

- **Device shall provide the monitoring events listed below.**

7.5.2 Client requirements (if supported)

- **If the client supports monitoring any of the events listed below, the client shall support monitoring each of the relevant topics listed below.**

7.5.3 Standard monitoring events list for devices

Standard Events for Monitoring		Device MANDATORY
Event	**Topic**	**Requirement**
Processor Usage	tns1:Monitoring/ProcessorUsage	M
Last Reset	tns1:Monitoring/OperatingTime/LastReset	M
Last Reboot	tns1:Monitoring/OperatingTime/LastReboot	M
Last Clock Synchronization	tns1:Monitoring/OperatingTime/LastClockSynchronization	M

7.5.4 Standard monitoring events list for clients

Standard Events for Monitoring		Client CONDITIONAL
Event	**Topic**	**Requirement**
Processor Usage	tns1:Monitoring/ProcessorUsage	C
Last Reset	tns1:Monitoring/OperatingTime/LastReset	C
Last Reboot	tns1:Monitoring/OperatingTime/LastReboot	C
Last Clock Synchronization	tns1:Monitoring/OperatingTime/LastClockSynchronization	C

7.6 Event handling

- **Retrieving and filtering of events from a device**

7.6.1 General requirements

- **The Real-time Pull-Point Notification Interface described in [ONVIF Network Interface Specifications] is mandatory for Profile Q conformance. The Base Notification Interface of the WS-BaseNotification as described in [ONVIF Network Interface Specifications] is not mandatory for Profile Q conformance.**

7.6.2 Device requirements

- **Device shall support at least two pull point subscriptions as described in [ONVIF Network Interface Specifications] (Event Service) by returning MaxPullPoints set to no less than two in the GetServiceCapabilities response.**

7.6.3 Client requirements (if supported)

- **Client shall implement event handling with a pull point using the CreatePullPointSubscription and PullMessage operations if any of the specific events described in this specification are supported.**

7.6.4 Event handling function list for devices

Event Handling		Device MANDATORY
Function	**Service**	**Requirement**
SetSynchronizationPoint	Event	M
CreatePullPointSubscription	Event	M
PullMessages	Event	M
GetEventProperties	Event	M
Renew	Event	M
Unsubscribe	Event	M
TopicFilter parameter of CreatePullPointSubscriptionRequest	Event	M

7.6.5 Event handling function list for clients

Event Handling		Client CONDITIONAL
Function	**Service**	**Requirement**
SetSynchronizationPoint	Event	O
CreatePullPointSubscription	Event	M
PullMessages	Event	M
GetEventProperties	Event	O
Renew	Event	O
Unsubscribe	Event	O
TopicFilter parameter of CreatePullPointSubscriptionRequest	Event	O

8 Profile conditional features (normative)

The profile conditional features section lists the features that shall be implemented if the device or client supports the feature. For instance, a device implementing firmware upgrade in the native API shall also implement the ONVIF firmware upgrade interface as specified in the [ONVIF Network Interface Specifications]. The requirements represent the minimum functionality that must be implemented for conformance.

8.1 Firmware upgrade (if supported)

- **Upgrade device firmware via HTTP.**

8.1.1 Device requirements (if supported)

- **Device shall support firmware upgrade through HTTP using the StartFirmwareUpgrade command.**

8.1.2 Client requirements (if supported)

- **Client shall be able to initiate a device firmware upgrade using the firmware upgrade operations. Client sends the StartFirmwareUpgrade command to instruct the device to prepare for upgrade, then it sends the firmware image using HTTP POST.**

8.1.3 Firmware upgrade function list for devices

Firmware Upgrade		**Device CONDITIONAL**
Function	**Service**	**Requirement**
StartFirmwareUpgrade	Device	M

8.1.4 Firmware upgrade function list for clients

Firmware Upgrade		**Client CONDITIONAL**
Function	**Service**	**Requirement**
StartFirmwareUpgrade	Device	M

8.2 Backup and restore (if supported)

- **Backup and restore configuration files.**
- **The behavior of restoring a backup made of a device in Factory Default State to a device in Operational State is not defined in this specification.**

8.2.1 Device requirements (if supported)

- **Device shall be able to backup system configurations using GetSystemUris.**
- **Device shall be able to restore system configurations using StartSystemRestore.**

8.2.2 Client requirements (if supported)

- **Client shall be able to backup system configurations on a device using GetSystemUris.**
- **Client shall be able to restore system configurations on a device using StartSystemRestore.**

8.2.3 Backup and restore function list for devices

Backup and Restore		**Device CONDITIONAL**
Function	**Service**	**Requirement**
GetSystemUris	Device	M
StartSystemRestore	Device	M

8.2.4 Backup and restore function list for clients

Backup and Restore		**Client CONDITIONAL**
Function	**Service**	**Requirement**
GetSystemUris	Device	M
StartSystemRestore	Device	M

8.3 TLS configuration (if supported)

- **Manage TLS server, certification path, certificate, and key configurations.**

8.3.1 Device requirements (if supported)

Device shall support the following Security Configuration Service capabilities:

- **Device shall support key configuration as defined by both the "RSAKeyPairGeneration" and the "PKCS12CertificateWithRSAPrivateKeyUpload" capability.**
- **Device shall support key generation status with both the “GetKeyStatus” function and with “tns1:Advancedsecurity/Keystore/KeyStatus” event notification.**
- **Device shall support “MaximumNumberOfKeys” capability of at least 16 to allow flexibility in certificate configuration.**

- **Device shall support certificate configuration as defined by each of the "SelfSignedCertificateCreationWithRSA", the "PKCS10ExternalCertificationWithRSA", and the “PKCS12CertificateWithRSAPrivateKeyUpload” capabilities.**
- **Device shall support “MaximumNumberOfCertificates” capability of at least 16 to allow flexibility in certification path configuration.**
- **Device shall support certification path configuration as defined by both the “PKCS12CertificateWithRSAPrivateKeyUpload” and the “TLSServerSupported” capabilities.**
- **Device shall support TLS server certificate assignment as defined by the “TLSServerSupported” capability.**
- **Device shall support the “SetNetworkProtocols” function to enable/disable TLS.**

8.3.2 Client requirements (if supported)

Client shall support the following Security Configuration Service capabilities:

- **Client shall support key configuration as defin ed by both the "RSAKeyPairGeneration" and the "PKCS12CertificateWithPrivateKeyUpload" capabilities.**
- **Client shall support retrieval of key generation status with either the “GetKeyStatus” function or with “tns1:Advancedsecurity/Keystore/KeyStatus” event notification.Client shall support certificate configuration as defined by both the "PKCS10ExternalCertificationWithRSA" and the “PKCS12CertificateWithRSAPrivateKeyUpload” capabilities.**
- **Client may choose to use either passphrase IDs or an actual passphrase in the “UploadCertificateWithPrivateKeyInPKCS12” function request.**
- **Client shall support certification path configuration as defined by both the “PKCS12CertificateWithRSAPrivateKeyUpload” and “TLSServerSupported” capabilities.**
- **Client shall support TLS server certificate assignment as defined by the “TLSServerSupported” capability.**
- **Client shall support the “SetNetworkProtocols” function to enable/disable TLS.**

8.3.3 TLS server configuration function list for devices

TLS Server Configuration		Device CONDITIONAL
Function	**Service**	**Requirement**
AddServerCertificateAssignment	Security	M
CreateCertificationPath	Security	M
CreatePKCS10CSR	Security	M
CreateRSAKeyPair	Security	M
CreateSelfSignedCertificate	Security	M
DeleteCertificate	Security	M
DeleteCertificationPath	Security	M
DeleteKey	Security	M
DeletePassphrase	Security	M
GetAllCertificates	Security	M
GetAllCertificationPaths	Security	M
GetAllKeys	Security	M
GetAllPassphrases	Security	M
GetAssignedServerCertificates	Security	M
GetCertificate	Security	M
GetCertificationPath	Security	M
GetKeyStatus	Security	M
tns1:Advancedsecurity/Keystore/KeyStatus	Event	M
RemoveServerCertificateAssignment	Security	M
ReplaceServerCertificateAssignment	Security	M
SetNetworkProtocols	Device	M
UploadCertificate	Security	M
UploadCertificateWithPrivateKeyInPKCS12	Security	M
UploadPassphrase	Security	M

8.3.4 TLS server configuration function list for clients

TLS Server Configuration		Client CONDITIONAL
Function	**Service**	**Requirement**
AddServerCertificateAssignment	Security	M
CreateCertificationPath	Security	M
CreatePKCS10CSR	Security	M
CreateRSAKeyPair	Security	M
CreateSelfSignedCertificate	Security	O
DeleteCertificate	Security	M
DeleteCertificationPath	Security	M
DeleteKey	Security	M
DeletePassphrase	Security	O
GetAllCertificates	Security	O
GetAllCertificationPaths	Security	O
GetAllKeys	Security	O
GetAllPassphrases	Security	O
GetAssignedServerCertificates	Security	O
GetCertificate	Security	O
GetCertificationPath	Security	O
GetKeyStatus	Security	O
tns1:Advancedsecurity/Keystore/KeyStatus	Event	O
RemoveServerCertificateAssignment	Security	M
ReplaceServerCertificateAssignment	Security	M
SetNetworkProtocols	Device	M
UploadCertificate	Security	M
UploadCertificateWithPrivateKeyInPKCS12	Security	M
UploadPassphrase	Security	O

8.4 Standard events for device management (if supported)

- **Standard events enable monitoring for critical device conditions.**

8.4.1 Device requirements (if supported)

- **Device shall provide device management events listed below if the components are available on the device.**

8.4.2 Client requirements (if supported)

- **Client shall use device managent events listed below if the client supports montitoring of the respective system condition.**

8.4.3 Standard device management events for devices

Standard Events for Device Management		Device CONDITIONAL
Event	**Topic**	**Requirement**
Fan Failure	tns1:Device/HardwareFailure/FanFailure	C
Power Supply Failure	tns1:Device/HardwareFailure/PowerSupplyFailure	C
Storage Failure	tns1:Device/HardwareFailure/StorageFailure	C
Temperature Critical	tns1:Device/HardwareFailure/TemperatureCritical	C
Last Backup	tns1:Monitoring/Backup/Last	C

8.4.4 Standard device management events for clients

Standard Events for Device Management		Client CONDITIONAL
Event	**Topic**	**Requirement**
Fan Failure	tns1:Device/HardwareFailure/FanFailure	C
Power Supply Failure	tns1:Device/HardwareFailure/PowerSupplyFailure	C
Storage Failure	tns1:Device/HardwareFailure/StorageFailure	C
Temperature Critical	tns1:Device/HardwareFailure/TemperatureCritical	C
Last Backup	tns1:Monitoring/Backup/Last	C

9 Factory Default State (Mandatory)

Devices conformant to Profile Q shall be in Factory Default State out-of-the-box and after hard factory reset. Factory Default State requires WS-Discovery, DHCP and IPv4 Link Local Address (defined as ZeroConfiguration capability) to be enabled by default as detailed below.

Factory Default State is signaled by the scope value:

```
onvif://www.onvif.org/Profile/Q/FactoryDefault
```

The FactoryDefault scope shall be included in the WS-Discovery ProbeMatch and Hello response. A device shall provide full anonymous access to all ONVIF commands while the device operates in Factory Default State. Full anonymous access shall only be terminated when the device exits from the Factory Default State and enters into Operational State.

The Factory Default State shall be terminated by the device when one of the following actions takes place:

- **the operator modifies the password of an existing admin user by invoking SetUser if an admin user exists**
- **the operator creates a new admin user by successfully invoking CreateUsers with a non-empty password if an admin user does not exist**

When switching from Factory Default State to Operational State, the device may reboot if necessary.

Clients conformant to Profile Q shall support corresponding commands as detailed below.

9.1 Device discovery

- **Devices on a local network shall be discoverable using device type and scope definitions according to the WS-Discovery standard as described in [ONVIF Network Interface Specifications]. A successful discovery provides the device service transport address to the client.**
- **The Factory Default State for devices shall be the Discoverable mode.**
- **Clients shall be able to discover a device using WS-Discovery as specified in [ONVIF Network Interface Specifications].**

9.1.1 Device requirements

- **Device shall support discovery operations.**

9.1.2 Client requirements

- **Client shall use discovery operations.**

9.1.3 Discovery function list for devices

Discovery		Device MANDATORY
Function	**Service**	**Requirement**
WS-Discovery	Core	M
GetScopes	Device	M

9.1.4 Discovery function list for clients

Discovery		Client MANDATORY
Function	**Service**	**Requirement**
WS-Discovery	Core	M
GetScopes	Device	O

9.2 ZeroConfiguration network configuration

The factory default device configuration shall have dynamic IP configuration enabled according to [RFC3927].

9.3 Automatic IP assignment

- **The factory default device configuration shall have IPv4 DHCP enabled.**
- **If the device has multiple network interfaces, this section shall apply to all interface(s) that expose the ONVIF API.**
- **If IPv6 is supported, the device shall have stateless autoconfiguration enabled.**

10 Operational State (Mandatory)

The Operational State shall be identified using the scope value:
`onvif://www.onvif.org/Profile/Q/Operational`

A device that is not in Factory Default State shall be in the Operational State. Transition from Factory Default to Operational State is defined in section 9.

A client shall support the corresponding functions.

10.1 Authentication

In Operational State, the device shall protect all services using HTTP digest authentication according to [RFC 2617] as described in [ONVIF Network Interface Specifications].

10.2 Default access policy

- **In the Operational State, the Profile Q device shall conform to the access policy which is specified by Default Access Policy and the SetAccessPolicy command.**
- **The factory default for Default Access Policy shall include user levels as described in [ONVIF Network Interface Specifications].**