ONVIF™
Access Rules Service Specification

Version 18.06
June 2018

CONTENTS

Contributors

ASSA ABLOY	Patrik Björling Rygert
ASSA ABLOY	Mattias Rengstedt
Axis Communications AB	Marcus Johansson
Axis Communications AB	Robert Rosengren
Axis Communications AB	Derek Wang
Axis Communications AB	Emil Selinder
Bosch	Mohane Caliaperoumal
Bosch	Dirk Schreiber
Honeywell	Uvaraj Thangarajan
Honeywell	Neelendra Bhandari
Honeywell	Mayur Salgar
Honeywell	Vinay Ghule
PACOM	Eugene Scully
Siemens AG	Lokeshwar K
Siemens AG	Suresh Raman
Siemens AG	Suresh Krishnamurthy

1 Scope

1.1 General

This specification defines the web service interface for integration with physical access control systems. This includes discovering components and support of the configuration of the access rules components.

The access rules service specification, which constitutes the access profiles and policies. The access rules service provides the access authorization for a credential.

Supplementary dedicated services such as credential services and schedule services will be defined in separate documents.

Web service usage and common ONVIF functionality are outside the scope of this document. Please refer to [Core Specification] for more information.

1.2 Conventions

The key words “shall”, “shall not”, “should”, “should not”, “may”, “need not”, “can”, “cannot” in this specification are to be interpreted as described in Annex H of [ISO/IEC Directives].

1.3 Namespaces

This document references the following namespaces:

Table 1 – Referenced namespaces (with prefix)

Prefix	Namespace URI
env	http://www.w3.org/2003/05/soap-envelope
ter	http://www.onvif.org/ver10/error
xs	http://www.w3.org/2001/XMLSchema
tt	http://www.onvif.org/ver10/schema
pt	http://www.onvif.org/ver10/pacs
tns1	http://www.onvif.org/ver10/topics
tac	http://www.onvif.org/ver10/accesscontrol/wsdl
tar	http://www.onvif.org/ver10/accessrules/wsdl

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ONVIF Core Specification
<http://www.onvif.org/specs/core/ONVIF-Core-Specification.pdf>

ONVIF PACS Architecture and Design Considerations
<https://www.onvif.org/specs/wp/ONVIF-PACS-Architecture-and-Design-Considerations.pdf>

ONVIF Access Control Service Specification
<http://www.onvif.org/specs/srv/access/ONVIF-AccessControl-Service-Spec.pdf>

ONVIF Schedule Service Specification
<http://www.onvif.org/specs/srv/sched/ONVIF-Scheduler-Service-Spec.pdf>

ISO/IEC Directives, *ISO/IEC Directives Part 2, Principles and rules for the structure and drafting of ISO and IEC documents, Edition 7.0, May 2016*
<http://www.iec.ch/members_experts/refdocs/iec/isoiecdir-2%7Bed7.0%7Den.pdf>

3 Terms, definitions and abbreviations

3.1 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

Access Policy — An association of an access point and a schedule. An access policy defines when an access point can be accessed using an access profile which contains this access policy.

Access Profile — A collection of access policies, used to define role based access.

Access Point — A logical composition of a physical door, reader(s) and/or a request-to-exit device controlling access in one direction.

Credential — A logical object holding related credential identifiers for a credential holder. E.g. if a PIN is associated with a specific credential number, then both of these identifiers are stored in one credential. Note that the PIN is normally not stored in the physical credential.

Validity Period — From a certain point in time, to a later point in time.

Schedule — A set of time periods, e.g. working hours (weekdays from 8 AM to 6 PM). It may also include one or more special days schedules.

Special Days — A set of dates that require the regular Schedule to be overridden, e.g. holidays, half-days or working Sundays.

3.2 Abbreviations

PACS — Physical Access Control System

4 Overview

The access rules service defines the access profile and its access policies. The credentials are associated to an access profiles for access authorization to a facility.

The access rules service defines WHEN and WHERE credentials have access. Each credential is associated with access profiles, where each access profile defines the access for a group of people. For example, employees will have access to office doors during office hours. Another example is access to an apartment by one family during all hours. Each access profile consists of a number of access policies, where each access policy defines when access is possible to an access point.

The service is flexible in such a way that it is possible to give access to something else than an access point (by setting EntityType to a QName other than AccessPointInfo).

The following picture shows the main data structures involved in the access rules service:

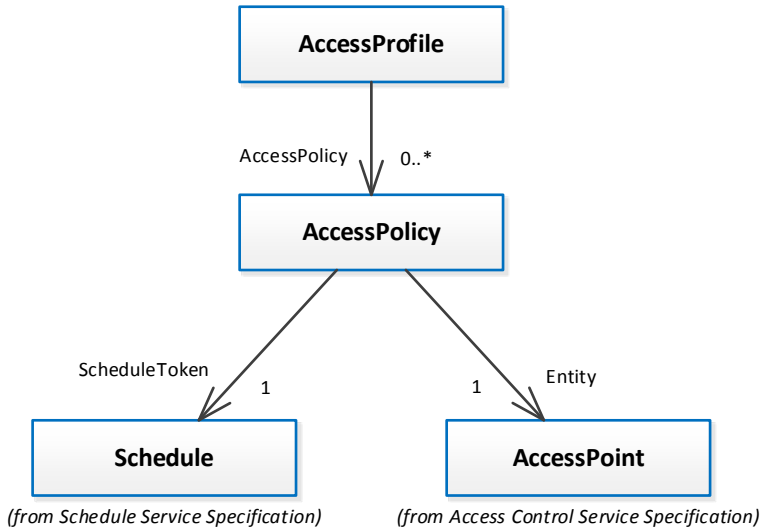

Figure 1: Main data structures in the access rules service

5 Access rules

5.1 General

The access rules service specification provides functionality for managing access authorization for an individual or a group of individuals. The service offers commands to manage the access rules and also determine WHEN and WHERE access is granted or denied.

5.2 Service capabilities

5.2.1 General

An ONVIF compliant device shall provide service capabilities in two ways:

1. With the GetServices method of Device service when IncludeCapability is true. Please refer to the ONVIF Core Specification for more details.

2. With the GetServiceCapabilities method.

5.2.2 Data structures

5.2.2.1 ServiceCapabilities

The service capabilities reflect optional functionality of a service. The information is static and does not change during device operation. The following capabilities are available:

- **MaxLimit**

  The maximum number of entries returned by a single Get<Entity>List or Get<Entity> request. The device shall never return more than this number of entities in a single response.

- **MaxAccessProfiles**

  Indicates the maximum number of access profiles supported by the device.

- **MaxAccessPoliciesPerAccessProfile**

  Indicates the maximum number of access policies per access profile supported by the device.

- **MultipleSchedulesPerAccessPointSupported**

  Indicates whether or not several access policies can refer to the same access point in an access profile.

- **ClientSuppliedTokenSupported**

  Indicates that the client is allowed to supply the token when creating access profiles. To enable the use of the command SetAccessProfile, the value must be set to true.

5.2.3 GetServiceCapabilities command

This operation returns the capabilities of the access rules service.

Table 2 GetServiceCapabilities command

GetServiceCapabilities	Access Class: PRE_AUTH
Message name	**Description**
GetServiceCapabilitiesRequest	*This message shall be empty*
GetServiceCapabilitiesResponse	*This message contains:* • *"Capabilities": The capability response message contains the requested access rules service capabilities using a hierarchical XML capability structure.* tar:ServiceCapabilities **Capabilities [1][1]**

5.3 Access Profile Information

5.3.1 General

Access profiles define who can access what and when.

5.3.2 Data Structures

5.3.2.1 AccessProfileInfo

The AccessProfileInfo structure contains basic information about an access profile. The device shall provide the following fields for each access profile instance.

- **token**

  A service unique identifier of the access profile.

- **Name**

  A user readable name. It shall be up to 64 characters.

To provide more information, the device may include the following optional fields:

- **Description**

  User readable description for the access profile. It shall be up to 1024 characters.

5.3.2.2 AccessProfile

The AccessProfile structure contains information about the collection of access policies. The device shall include all properties of the AccessProfileInfo structure and also a list of access policies.

- **AccessPolicy**

  A list of access policy structures, where each access policy defines during which schedule an access point can be accessed.

5.3.2.3 AccessPolicy

The access policy is an association of an access point and a schedule. It defines when an access point can be accessed using an access profile which contains this access policy. If an access profile contains several access policies specifying different schedules for the same access point will result in a union of the schedules.

The device shall provide the following fields for each access policy instance.

- **ScheduleToken**

  Reference to the schedule used by the access policy. Schedules are defined in [ONVIF Schedule Service Specification].

- **Entity**

  Reference to the entity used by the rule engine, the entity type may be specified by the optional EntityType field explained below but is typically an access point.

To provide more information, the device may include the following optional field:

- **EntityType**

  Optional entity type; if missing, an access point type as defined in [ONVIF Access Control Service Specification] should be assumed. This can also be represented by the QName value “tac:AccessPoint” where tac is the namespace of [ONVIF Access Control Service Specification]. This field is provided for future extensions; it will allow an access policy being extended to cover entity types other than access points as well.

5.3.3 GetAccessProfileInfo command

This operation requests a list of AccessProfileInfo items matching the given tokens.

The device shall ignore tokens it cannot resolve and shall return an empty list if there are no items matching the specified tokens. The device shall not return a fault in this case.

If the number of requested items is greater than MaxLimit, a TooManyItems fault shall be returned.

Table 3 GetAccessProfileInfo command

GetAccessProfileInfo	Access Class: READ_SYSTEM
Message name	**Description**
GetAccessProfileInfoRequest	*This message contains:* • *"Token": Tokens of AccessProfileInfo items to get.* pt:ReferenceToken **Token [1][unbounded]**
GetAccessProfileInfoResponse	*This message contains:* • *"AccessProfileInfo": List of AccessProfileInfo items.* tar:AccessProfileInfo **AccessProfileInfo [0][unbounded]**
Fault codes	**Description**
env:Sender ter:InvalidArgs ter:TooManyItems	*Too many items were requested, see MaxLimit capability.*

5.3.4 GetAccessProfileInfoList command

This operation requests a list of all AccessProfileInfo items provided by the device.

A call to this method shall return a StartReference when not all data is returned and more data is available. The reference shall be valid for retrieving the next set of data. Please refer to section 4.8.3 in [ONVIF PACS Architecture and Design Considerations] for more details.

The number of items returned shall not be greater than the Limit parameter.

Table 4 GetAccessProfileInfoList command

GetAccessProfileInfoList	Access Class: READ_SYSTEM
Message name	**Description**
GetAccessProfileInfoListRequest	*This message contains:* • *"Limit": Maximum number of entries to return. If not specified, less than one or higher than what the device supports, the number of items is determined by the device.* • *"StartReference": Start returning entries from this start reference. If not specified, entries shall start from the beginning of the dataset.* xs:int **Limit [0][1]** xs:string **StartReference [0][1]**
GetAccessProfileInfoListResponse	*This message contains:* • *"NextStartReference": StartReference to use in next call to get the following items. If absent, no more items to get.* • *"AccessProfileInfo": List of AccessProfileInfo items.* xs:string **NextStartReference [0][1]** tar:AccessProfileInfo **AccessProfileInfo [0][unbounded]**
Fault codes	**Description**
env:Sender ter:InvalidArgVal ter:InvalidStartReference	*StartReference is invalid or has timed out. Client needs to start fetching from the beginning.*

5.3.5 GetAccessProfiles command

This operation requests a list of AccessProfile items matching the given tokens.

The device shall ignore tokens it cannot resolve and shall return an empty list if there are no items matching the specified tokens. The device shall not return a fault in this case.

If the number of requested items is greater than MaxLimit, a TooManyItems fault shall be returned.

Table 5 GetAccessProfiles command

GetAccessProfiles	Access Class: READ_SYSTEM
Message name	**Description**
GetAccessProfileRequest	*This message contains:* • "*Token": Tokens of Access Profile items to get* pt:ReferenceToken **Token [1][unbounded]**
GetAccessProfileResponse	*This message contains:* • *"AccessProfile": List of Access Profile items.* tar:AccessProfile **AccessProfile [0][unbounded]**
Fault codes	**Description**
env:Sender ter:InvalidArgs ter:TooManyItems	*Too many items were requested, see MaxLimit capability.*

5.3.6 GetAccessProfileList command

This operation requests a list of all AccessProfile items provided by the device.

A call to this method shall return a StartReference when not all data is returned and more data is available. The reference shall be valid for retrieving the next set of data. Please refer to section 4.8.3 in [ONVIF PACS Architecture and Design Considerations] for more details.

The number of items returned shall not be greater than the Limit parameter.

Table 6 GetAccessProfileList command

GetAccessProfileList	Access Class: READ_SYSTEM
Message name	**Description**
GetAccessProfileListRequest	*This message contains:* • *"Limit": Maximum number of entries to return. If not specified, less than one or higher than what the device supports, the number of items is determined by the device.* • *"StartReference": Start returning entries from this start reference. If not specified, entries shall start from the beginning of the dataset.* xs:int **Limit [0][1]** xs:string **StartReference [0][1]**
GetAccessProfileListResponse	*This message contains:* • *"NextStartReference": StartReference to use in next call to get the following items. If absent, no more items to get.* • *"AccessProfile": List of Access Profile items.* xs:string **NextStartReference [0][1]** tar:AccessProfile **AccessProfile [0][unbounded]**
Fault codes	**Description**
env:Sender ter:InvalidArgVal ter:InvalidStartReference	*StartReference is invalid or has timed out. Client needs to start fetching from the beginning.*

5.3.7 CreateAccessProfile command

This operation creates the specified access profile in the device.

The token field of the AccessProfile structure shall be empty and the device shall allocate a token for the access profile. The allocated token shall be returned in the response.

If the client sends any value in the token field, the device shall return InvalidArgVal as a generic fault code.

If several access policies in one access profile are specifying different schedules for the same access point, then it will result in a union of the schedules.

The following figure shows an example of several schedules used at one access point. Each row in the figure below corresponds to one access policy. The first and second line are two access policies using different schedules for the same access point.

Time periods of the individual schedule					Access Policies	
08:00-09:00	09:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	**Schedule**	**Access Point**
					Schedule 1	Room A Entry
					Schedule 2	Room A Entry
					Schedule 1	Room B Entry

Figure 2: Multiple schedules per access point

Since both Schedule 1 and Schedule 2 defines when Room A Entry can be accessed, a credential holder will experience a union of Schedule 1 and Schedule 2, as seen in the first row in the figure below.

Time periods of the schedule union					Access Policies	
08:00-09:00	09:00-10:00	10:00-11:00	11:00-12:00	12:00-13:00	**Schedule**	**Access Point**
					Schedule 1 union Schedule 2	Room A Entry
					Schedule 1	Room B Entry

Figure 3: Result of schedule union

Table 7 CreateAccessProfile command

CreateAccessProfile	Access Class: WRITE_SYSTEM
Message name	**Description**
CreateAccessProfileRequest	*This message contains:* • *"AccessProfile": The AccessProfile to create* tar:AccessProfile **AccessProfile [1][1]**
CreateAccessProfileResponse	*This message contains:* • *"Token": The Token of created AccessProfile* pt:ReferenceToken **Token [1][1]**
Fault codes	**Description**
env:Receiver ter:CapabilityViolated ter:MaxAccessProfiles	*There is not enough space to add new AccessProfile, see the MaxAccessProfiles capability*
env:Sender ter:CapabilityViolated ter:MaxAccessPoliciesPerAccessProfile	*There are too many AccessPolicies in anAccessProfile, see MaxAccessPoliciesPerAccessProfile capability.*
env:Sender ter:CapabilityViolated ter:MultipleSchedulesPerAccessPoint-Supported	*Multiple AccessPoints are not supported for the same schedule, see MultipleSchedulesPerAccessPoint-Supported capability.*
env:Sender ter:InvalidArgVal ter:ReferenceNotFound	*A referred entity token is not found (some devices may not validate referred entities).*

5.3.8 SetAccessProfile command

This method is used to synchronize an access profile in a client with the device.

If an access profile with the specified token does not exist in the device, the access profile is created. If an access profile with the specified token exists, then the access profile is modified.

A call to this method takes an access profile structure as input parameter. The token field of the access profile must not be empty.

A device that signals support for the ClientSuppliedTokenSupported capability shall implement this command.

If no token was specified in the request, the device shall return InvalidArgs as a generic fault code.

Table 8 SetAccessProfile command

SetAccessProfile	Access Class: WRITE_SYSTEM
Message name	**Description**
SetAccessProfileRequest	*This message contains:* • *"AccessProfile": The AccessProfile item to create or modify* tar:AccessProfile **AccessProfile [1][1]**
SetAccessProfileResponse	*This message shall be empty*
Fault codes	**Description**
env:Receiver ter:CapabilityViolated ter:ClientSuppliedTokenSupported	*The device does not support that the client supplies the token*
env:Receiver ter:CapabilityViolated ter:MaxAccessProfiles	*There is not enough space to add new AccessProfile, see the MaxAccessProfiles capability*
env:Sender ter:CapabilityViolated ter:MaxAccessPoliciesPerAccessProfile	*There are too many AccessPolicies in anAccessProfile, see MaxAccessPoliciesPer-AccessProfile capability.*
env:Sender ter:CapabilityViolated ter:MultipleSchedulesPerAccessPointSupported	*Multiple AccessPoints are not supported for the same schedule, see MultipleSchedulesPer-AccessPointSupported capability.*
env:Sender ter:InvalidArgVal ter:ReferenceNotFound	*A referred entity token is not found (some devices may not validate referred entities).*

5.3.9 ModifyAccessProfile command

This operation modifies the specified access profile.

The token of the access profile to modify is specified in the token field of the AccessProfile structure and shall not be empty. All other fields in the structure shall overwrite the fields in the specified access profile.

If several access policies specifying different schedules for the same access point will result in a union of the schedules. See Figure 2 and Figure 3 above.

If no token was specified in the request, the device shall return InvalidArgs as a generic fault code.

Table 9 ModifyAccessProfile command

ModifyAccessProfile	Access Class: WRITE_SYSTEM
Message name	**Description**
ModifyAccessProfileRequest	*This message contains:* • *"AccessProfile": The details of Access Profile* tar:AccessProfile **AccessProfile[1][1]**
ModifyAccessProfileResponse	*This message shall be empty*
Fault codes	**Description**
env:Sender ter:InvalidArgVal ter:NotFound	*Access profile token is not found.*
env:Sender ter:CapabilityViolated ter:MaxAccessPoliciesPerAccessProfile	*There are too many AccessPolicies in an AccessProfile, see MaxAccessPoliciesPer-AccessProfile capability.*
env:Sender ter:CapabilityViolated ter:MultipleSchedulesPerAccessPointSupported	*Multiple AccessPoints are not supported for the same schedule, see MultipleSchedulesPer-AccessPointSupported capability.*
env:Sender ter:InvalidArgVal ter:ReferenceNotFound	*A referred entity token is not found (some devices may not validate referred entities).*

5.3.10 DeleteAccessProfile command

This operation will delete the specified access profile.

If the access profile is deleted, all access policies associated with the access profile will also be deleted.

If it is associated with one or more entities some devices may not be able to delete the access profile, and consequently a ReferenceInUse fault shall be generated.

If no token was specified in the request, the device shall return InvalidArgs as a generic fault code.

Table 10 DeleteAccessProfile command

DeleteAccessProfile	Access Class: WRITE_SYSTEM
Message name	**Description**
DeleteAccessProfileRequest	*This message contains:* • *"Token": The token of the access profile to delete.* pt:ReferenceToken **Token [1][1]**
DeleteAccessProfileResponse	*This message shall be empty*
Fault codes	**Description**
env:Sender ter:InvalidArgVal ter:NotFound	*Access profile token is not found.*
env:Sender ter:InvalidArgVal ter:ReferenceInUse	*Failed to delete, Access profile token is in use*

6 Notification topics

6.1 General

This section defines notification topics specific to the access rules service.

6.2 Event overview (informative)

The access rules service specifies events when access profiles are changed.

The main topics for configuration change notifications are:

- tns1:Configuration/AccessProfile/Changed
- tns1:Configuration/AccessProfile/Removed

6.3 Configuration changes

6.3.1 General

Whenever configuration data has been changed, added or been removed, the device shall provide these events to inform subscribed clients.

6.3.2 Access profile

Whenever configuration data for an access profile is changed or an access profile is added, the device shall provide the following event:

```
Topic: tns1:Configuration/AccessProfile/Changed

<tt:MessageDescription IsProperty="false">
  <tt:Source>
    <tt:SimpleItemDescription Name="AccessProfileToken"
                              Type="pt:ReferenceToken"/>
  </tt:Source>
</tt:MessageDescription>
```

Whenever an access profile is removed, the device shall provide the following event:

```
Topic: tns1:Configuration/AccessProfile/Removed

<tt:MessageDescription IsProperty="false">
  <tt:Source>
    <tt:SimpleItemDescription Name="AccessProfileToken"
                              Type="pt:ReferenceToken"/>
  </tt:Source>
</tt:MessageDescription>
```

Annex A. Revision History

Rev.	Date	Editor	Changes
1.0	Jun-2015	PACS WG	First release
18.06	Jun-2018	Patrik Björling Rygert	Added support for client-supplied tokens