VAPIX® version 3

RTSP API

VAPIX LICENSE AGREEMENT

This is a legal agreement (the "License Agreement") between you (either individual or an entity) and Axis Communications AB (hereinafter referred to as "Axis").

1. GRANT OF LICENSE

   Axis hereby grants to you the right to use the INTERFACE and the INTERFACE DESCRIPTION for the sole and limited purpose of creating, manufacturing and developing a solution that integrates any unit or portion included in the product range of Axis network cameras, Axis video encoders, Axis video encoders and Axis video decoders (as defined by Axis at its discretion) and to market, sell and distribute any such solution.

2. COPYRIGHT

   The INTERFACE and the INTERFACE DESCRIPTION are owned by Axis and are protected by copyright laws and international treaty provisions. Any use of the INTERFACE and/or the INTERFACE DESCRIPTION outside the limited purpose set forth in Section 1 above is strictly prohibited.

3. NO REVERSE ENGINEERING

   You may not reverse engineer, decompile, or disassemble the INTERFACE except to the extent required to obtain interoperability with other independently created computer programs as permitted by mandatory law.

4. TERMINATION

   This License is effective until terminated. Your rights under this License will terminate automatically without notice from Axis if you fail to comply with any term(s) of this License. Upon the termination of this License, you shall cease all use and disposition of the INTERFACE and/or THE INTERFACE DESCRIPTION whether for the purpose set forth in Section 1 above or not.

5. GOVERNING LAW

   This agreement shall be deemed performed in and shall be construed by the laws of Sweden. All disputes in connection with this agreement shall be finally settled by arbitration in accordance with the Rules of the Arbitration Institute of the Stockholm Chamber of

Commerce. The place of arbitration shall be Malmö, Sweden. The language of the proceedings, documentation and the award shall be English.

6. DISCLAIMER

6.1. THE INTERFACE AND THE INTERFACE DESCRIPTION ARE DELIVERED FREE OF CHARGE AND "AS IS" WITHOUT WARRANTY OF ANY KIND. THE ENTIRE RISK AS TO THE USE, RESULTS AND PERFORMANCE OF THE INTERFACE AND THE INTERFACE DESCRIPTION IS ASSUMED BY THE USER/YOU. AXIS DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, NON-INFRINGEMENT AND PRODUCT LIABILITY, OR ANY WARRANTY ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE WITH RESPECT TO THE INTERFACE AND THE INTERFACE DESCRIPTION.

6.2. YOU ARE YOURSELF RESPONSIBLE FOR EXAMINING WHETHER THE INTERFACE AND THE INTERFACE DESCRIPTION ARE ENCUMBERED BY OR INFRINGES UPON A RIGHT HELD BY A THIRD PARTY. AXIS, WHO HAS NOT UNDERTAKEN ANY SUCH INVESTIGATIONS, HAS NO KNOWLEDGE OF NOR DOES AXIS ACCEPT ANY LIABILITY FOR ANY SUCH ENCUMBRANCES OR INFRINGEMENTS.

6.3. YOU UNDERTAKE NOT TO PURSUE ANY CLAIMS WHATSOEVER AGAINST AXIS OR ITS AFFILIATES RELATING TO OR EMANATING FROM THE INTERFACE AND THE INTERFACE DESCRIPTION.

6.4. AXIS SHALL NOT BE LIABLE FOR LOSS OF DATA, LOSS OF PRODUCTION, LOSS OF PROFIT, LOSS OF USE, LOSS OF CONTRACTS OR FOR ANY OTHER CONSEQUENTIAL, ECONOMIC OR INDIRECT LOSS WHATSOEVER IN RESPECT OF USE OR DISPOSITION OF THE INTERFACE AND THE INTERFACE DESCRIPTION.

6.5. AXIS TOTAL LIABILITY FOR ALL CLAIMS IN ACCORDANCE WITH THE USE OF THE INTERFACE AND THE INTERFACE DESCRIPTION SHALL NOT EXCEED THE PRICE PAID FOR THE INTERFACE AND THE INTERFACE DESCRIPTION.

6.6. YOU SHALL INDEMNIFY AND HOLD AXIS AND ITS AFFILIATES HARMLESS FROM ANY CLAIMS WHATSOEVER FROM ANY THIRD PARTY AGAINST AXIS OR ITS AFFILIATES RELATING TO OR EMANATING FROM YOUR USE OF THE INTERFACE AND THE INTERFACE DESCRIPTION UNDER THIS LICENSE AGREEMENT. THE FOREGOING INDEMNIFICATION INCLUDES BUT IS NOT LIMITED TO ANY AND ALL DAMAGES, COSTS AND EXPENSES (INCLUDING REASONABLE ATTORNEYS' FEES).

Table of Contents

1 Overview

1.1 Description

This document describes the external RTSP-based application programming interface (API) to Axis cameras and video encoders with firmware 5.00 and above.

RTSP (Real Time Streaming Protocol) is a control protocol for media streams delivered by a media server. RTSP can be considered a “remote control” providing commands such as play and pause. In addition, Axis RTSP API provides parameters controlling media stream properties such as resolution, compression, video bit rate and audio as well as parameters controlling the image appearance including, for example, text and image overlay settings.

In contrast to HTTP, RTSP supports UTF-8. Hence UTF-8 encoded characters can be used directly, for example in the textstring parameter.

Support for the RTSP API is product and firmware dependent. Please refer to the Release Notes for the actual product for compliance information.

Note: For readability, lines have been broken in several of the examples in this document. There is no indication that a certain line is broken.

1.2 History

Version	Date	Comment
2.00	2008-Aug-27	Initial version
2.01	2008-Sep-3	Added note about line breaks. Corrected typos.

2 Prerequisites

2.1 Identification

Property: Properties.API.Version=3

Firmware: 5.00 and above

3 RTSP Commands

The RTSP API provides several commands for media stream control.

3.1 Request syntax

Syntax:

```
COMMAND rtsp://<servername>/axis-media/media.amp
[?<parameter>=<value>[&<parameter>=<value>...]] RTSP/1.0<CRLF>
Headerfield1: val1<CRLF>
Headerfield2: val2<CRLF>
...
<CRLF>

[Body]
```

COMMAND is any of DESCRIBE, SETUP, OPTIONS, PLAY, PAUSE, TEARDOWN or SET_PARAMETER.* Lines are separated with Carriage Return and Line Feed (CRLF).

`<servername>` is replaced by the host name or IP address of the product. Note that RTSP requests always contain the absolute URL.

Supported parameters and their values are listed in section 5.

The following header fields are accepted by all commands. Some commands accept or require additional header fields.

Header Field	Description
`Authorization`	Authorization information from the client.
`CSeq`	Request sequence number.
`Session`	Session identifier (returned by server in SETUP response).
`Content-Length`	Length of content.
`Content-Type`	The media type of the content.
`User-Agent`	Information about the client that initiates the request.
`Require`	Query whether an option is supported. Unsupported features are listed in the Unsupported header field.

* The GET_PARAMETER command is also supported; there are however no parameters to retrieve so this command is not very useful.

3.2 Response syntax

Syntax:

```
RTSP/1.0 <Status Code> <Reason Phrase> <CRLF>
Headerfield3: val3<CRLF>
Headerfield4: val4<CRLF>
...
<CRLF>

[Body]
```

The first response line contains a status code and a reason phrase indicating the success or failure of the request. The status codes are described in RFC 2326.

The following header fields can be included in all RTSP response messages.

Header Field	Description
CSeq	Response sequence number (matches the sequence number of the request).
Session	Session identifier.
WWW-Authenticate	Authentication from client requested.
Date	Date and time of the response.
Unsupported	Features not supported by the server.

3.3 RTSP DESCRIBE

The DESCRIBE command is used to request an SDP description of the media stream(s). The Session Description Protocol (SDP) is described in RFC 2327.

The DESCRIBE request accepts the additional header field:

Header Field	Description
Accept	List of content types that client supports (application/sdp is the only supported type).

The response to the DESCRIBE command contains the additional header fields:

Header Field	Description
Content-Type	Type of content (application/sdp).
Content-Length	Length of SDP description.
Content-Base	If relative URLs are used in the SDP description, this is the base URL.

Example:

Request

```
DESCRIBE rtsp://myserver/axis-media/media.amp
?videocodec=h264&resolution=640x480 RTSP/1.0
CSeq: 0
User-Agent: Axis AMC
Accept: application/sdp
Authorization: Basic cm9vdDpwYXNz
```

Response

```
RTSP/1.0 200 OK
CSeq: 0
Content-Type: application/sdp
Content-Base: rtsp://myserver/axis-media/media.amp/
Date: Wed, 16 Jul 2008 12:48:47 GMT
Content-Length: 847

v=0
o=- 1216212527554872 1216212527554872 IN IP4 myserver
s=Media Presentation
e=NONE
c=IN IP4 0.0.0.0
b=AS:50064
t=0 0
a=control:rtsp://myserver/axis-media/media.amp?videocodec=h264
&resolution=640x480
a=range:npt=0.000000-
m=video 0 RTP/AVP 96
b=AS:50000
a=framerate:30.0
a=control:rtsp://myserver/axis-media/media.amp/trackID=1
?videocodec=h264&resolution=640x480
a=rtpmap:96 H264/90000
a=fmtp:96 packetization-mode=1; profile-level-id=420029; sprop-
parameter-sets=Z0IAKeKQFAe2AtwEBAaQeJEV,aM48gA==
m=audio 0 RTP/AVP 97
b=AS:64
a=control:rtsp://myserver/axis-media/media.amp/trackID=2
?videocodec=h264&resolution=640x480
a=rtpmap:97 mpeg4-generic/16000/1
a=fmtp:97 profile-level-id=15; mode=AAC-hbr;config=1408; SizeLength=13;
IndexLength=3;IndexDeltaLength=3; Profile=1; bitrate=64000;
```

3.4 RTSP OPTIONS

The OPTIONS request returns a list of supported RTSP commands. The command can be used to keep RTSP sessions alive by repeating the OPTIONS request at regular intervals. The session timeout time is specified by the `timeout` parameter returned from the SETUP command (see 3.5).

The response to the OPTIONS command contains the additional header field:

Header field	Description
`Public`	Specify the supported RTSP commands.

Example: List supported commands. The asterisk (*) makes the request apply to the server and not to a particular URL.

Request

```
OPTIONS * RTSP/1.0
CSeq: 1
User-Agent: Axis AMC
Session: 12345678
Authorization: Basic cm9vdDpwYXNz
```

Response

```
RTSP/1.0 200 OK
CSeq: 1
Session: 12345678
Public: DESCRIBE, GET_PARAMETER, PAUSE, PLAY, SETUP, SET_PARAMETER,
TEARDOWN
Date: Wed, 16 Jul 2008 12:48:48 GMT
```

As indicated in the response, the `GET_PARAMETER` command is supported; there are however no parameters to retrieve so this command is not very useful.

3.5 RTSP SETUP

The SETUP command is used to configure the data delivery method.

The SETUP request requires an additional header field which is also included in the response:

Header field	Description
`Transport`	Specify how the data stream is transported. Supported variants are RTP/AVP;unicast;client_port=port1-port2 RTP/AVP;multicast;client_port=port1-port2 RTP/AVP/TCP;unicast

If using unicast in combination with TCP, it is recommended to increase the size of the RTP packets (from the standard 1500 bytes), provided that the client can accept larger packets. Also for unicast streaming over RTP/UDP it might be beneficial to increase the packet size if no packets are dropped. The packet size is changed using the following header field in the SETUP request:

Header field	Description
`Blocksize`	Request a specific media packet size. The packet size should be a positive decimal number measured in octets.

The response returns a session identifier that should be used together with the stream control commands (e.g. PLAY, PAUSE and TEARDOWN). If the session header includes the `timeout` parameter, the session will close after the timeout time unless explicitly kept alive. Session can be kept alive by sending RTSP requests to the server containing the session identifier (e.g. OPTIONS, see 3.4) within the timeout time or by using RTCP messages. Reconfiguration of transport parameters is not supported.

Example: The response to the first SETUP request returns the session identifier (`Session`) which is used in subsequent requests.

Request

```
SETUP rtsp://myserver/axis-media/media.amp/
trackID=1?videocodec=h264&resolution=640x480 RTSP/1.0
CSeq: 2
User-Agent: Axis AMC
Transport: RTP/AVP;unicast;client_port=20000-20001
Authorization: Basic cm9vdDpwYXNz
```

Response

```
RTSP/1.0 200 OK
CSeq: 2
Session: 12345678; timeout=60
Transport: RTP/AVP;unicast;client_port=20000-20001;server_port=50000-
50001;ssrc=B0BA7855;mode="PLAY"
Date: Wed, 16 Jul 2008 12:48:47 GMT
```

Example:

Request

```
SETUP rtsp:///myserver//axis-media/media.amp/
trackID=2?videocodec=h264&resolution=640x480 RTSP/1.0
CSeq: 3
User-Agent: Axis AMC
Transport: RTP/AVP;unicast;client_port=20002-20003
Session: 12345678
Authorization: Basic cm9vdDpwYXNz
```

Response

```
RTSP/1.0 200 OK
CSeq: 3
Session: 12345678; timeout=60
Transport: RTP/AVP;unicast;client_port=20002-20003;server_port=50002-
50003;ssrc=D7EB59C0;mode="PLAY"
Date: Wed, 16 Jul 2008 12:48:48 GMT
```

3.6 RTSP PLAY

The PLAY request starts (or restarts if paused) the data delivery to the client.

The response to the PLAY command contains the additional header fields:

Header field	Description
`Range`	The play time period.
`RTP-Info`	Information about the RTP stream, including the sequence number of the first packet of the stream.

Example:

Request

```
PLAY rtsp://myserver/axis-media/media.amp
?videocodec=h264&resolution=640x480 RTSP/1.0
CSeq: 4
User-Agent: Axis AMC
Session: 12345678
Authorization: Basic cm9vdDpwYXNz
```

Response

```
RTSP/1.0 200 OK
CSeq: 4
Session: 12345678
Range: npt=0.645272-
RTP-Info: url=rtsp://myserver/axis-media/media.amp/
trackID=1?videocodec=h264&resolution=640x480;seq=46932;
rtptime=1027887748, url=rtsp://myserver/axis-media/media.amp/
trackID=2?videocodec=h264&resolution=640x480;seq=3322;
rtptime=611053482
Date: Wed, 16 Jul 2008 12:48:48 GMT
```

Example: Play back the recording “myrecording”.

Request

```
PLAY rtsp://myserver/axis-media/media.amp
?recordingid=”myrecording” RTSP/1.0
CSeq: 4
User-Agent: Axis AMC
Session: 12345678
Authorization: Basic cm9vdDpwYXNz
```

3.7 RTSP PAUSE

The PAUSE request is used to temporarily stop data delivery from the server. Use PLAY to restart data delivery.

Example:

Request

```
PAUSE rtsp://myserver/axis-media/media.amp
?videocodec=h264&resolution=640x480 RTSP/1.0
CSeq: 5
User-Agent: Axis AMC
Session: 12345678
Authorization: Basic cm9vdDpwYXNz
```

Response

```
RTSP/1.0 200 OK
CSeq: 5
Session: 12345678
Date: Wed, 16 Jul 2008 12:48:49 GMT
```

3.8 RTSP TEARDOWN

The TEARDOWN request is used to terminate the data delivery from the server.

Example:

Request

```
TEARDOWN rtsp://myserver/axis-media/media.amp
?videocodec=h264&resolution=640x480 RTSP/1.0
CSeq: 6
User-Agent: Axis AMC
Session: 12345678
Authorization: Basic cm9vdDpwYXNz
```

Response

```
RTSP/1.0 200 OK
CSeq: 6
Session: 12345678
Date: Wed, 16 Jul 2008 12:49:01 GMT
```

3.9 RTSP SET_PARAMETER

The SET_PARAMETER command is used to request an I-frame, for example when starting a recording. The command sets the value of the Renew-Stream parameter to yes. Note that Renew-Stream must be sent in the body.*

* The corresponding RenewStream parameter in some firmware 4.xx products had to be sent in the header. See second example below.

Example: Use of SET_PARAMETER in firmware 5.xx products. Renew-Stream is sent in the body.

Request

```
SET_PARAMETER rtsp://myserver/axis-media/media.amp RTSP/1.0
CSeq: 7
Session: 12345678
Authorization: Basic cm9vdDpwYXNz
Content-Type: text/parameters
Content-Length: 19

Renew-Stream: yes
```

Response

```
RTSP/1.0 200 OK
CSeq: 7
Session: 12345678
Date: Wed, 16 Jul 2008 13:01:25 GMT
```

Example: In some of Axis' older products, I-frames were requested using RenewStream:yes in the *header*. To find out whether Renew-Stream should be sent in the header or the body, the following method is recommended.

Send a request with Require and RenewStream:yes in the header.

```
SET_PARAMETER rtsp://myserver/axis-media/media.amp RTSP/1.0
CSeq: 7
Session: 12345678
Authorization: Basic cm9vdDpwYXNz
Require: com.axis.parameters-in-header
RenewStream: yes
```

If the request is successful (response "200 OK"), the stream is renewed. Else, the server responds with "551 Option not supported" (below) and Renew-Stream should be sent in the body.

```
RTSP/1.0 551 Option not supported
CSeq: 7
Session: 12345678
Unsupported: com.axis.parameters-in-header
Date: Wed, 16 Jul 2008 13:01:24 GMT
```

Send a second request with Renew-Stream: yes in the body.

```
SET_PARAMETER rtsp://myserver/axis-media/media.amp RTSP/1.0
CSeq: 8
Session: 12345678
Authorization: Basic cm9vdDpwYXNz
Content-Type: text/parameters
Content-Length: 19

Renew-Stream: yes
```

Successful response.

```
RTSP/1.0 200 OK
CSeq: 8
Session: 12345678
Date: Wed, 16 Jul 2008 13:01:25 GMT
```

4 RTSP over HTTP

RTSP can be tunneled through HTTP. This might prove necessary in order to pass firewalls etc. To tunnel RTSP over HTTP, two sessions are set up; one GET (for command replies and stream data) and one POST (for commands). RTSP commands sent on the POST connection are base64 encoded, but the replies on the GET connection are in plain text. To bind the two sessions together the server needs a unique ID (conveyed in the `x-sessioncookie` header). The GET and POST requests are accepted on both the HTTP port (default 80) and the RTSP server port (default 554).

Syntax: The RTSP/HTTP URL

```
http://<servername>/axis-media/media.amp
```

Supported methods are GET and POST.

Example: GET example

Request

```
GET axis-media/media.amp?videocodec=h264&audio=0 HTTP/1.0
x-sessioncookie: 123456789
```

Response

```
HTTP/1.0 200 OK
Content-Type: application/x-rtsp-tunnelled
```

Example: POST request. There is no response from the server.

Request

```
POST axis-media/media.amp?videocodec=h264&audio=0 HTTP/1.0
x-sessioncookie: 123456789
Content-Length: 32767
Content-Type: application/x-rtsp-tunnelled
```

5 Parameter Specification

Axis RTSP API provides parameters for requesting media streams with specific properties and for setting the image appearance. The parameters are entered in the RTSP URL.

Syntax:

```
rtsp://<servername>/axis-media/media.amp
[?<parameter>=<value>[&<parameter>=<value>...]]
```

The following parameters are supported for H.264, MPEG-4 Part 2 and MJPEG streams.

Parameter	Valid values	Description
`videocodec`	h264, mpeg4, jpeg [1]	The selected video codec. *Default*: Product dependent; in order of priority: h264, mpeg4, jpeg
`streamprofile`	string	Name of a saved stream profile.[2]
`recordingid`	string	Name of a saved recording.
`resolution`	1280x1024, 1280x960, 1280x720, 768x576, 4CIF, 704x576, 704x480, VGA, 640x480, 640x360, 2CIFEXP, 2CIF, 704x288, 704x240, 480x360, CIF, 384x288, 352x288, 352x240, 320x240,	Specify the resolution of the returned image.

	240x180, QCIF, 192x144, 176x144, 176x120, 160x120 [1]	
audio	0, 1	Specify whether audio shall be available in the stream (for compatibility with applications without audio control). 0=no audio, 1=audio *Default:* 1
camera	1,2,3,4 or quad [1]	Select the source camera. Applies only to video encoders with more than one video input.
compression	0 ... 100 [1]	Adjust the compression level of the image. Higher values correspond to higher compression, i.e. lower image quality and smaller image size. **Note:** This value is internally mapped and therefore product-dependent.
colorlevel[2]	0 ... 100 [1]	Set the level of color or grey scale. 0=grey scale, 100= full color **Note:** This value is internally mapped and therefore product-dependent.
color	0, 1	Enable/disable color. 0=black and white, 1=color
clock	0, 1	Show/hide the time stamp. 0=hide, 1=show
date	0, 1	Show/hide the date. 0=hide, 1=show
text	0, 1	Show/hide the text. 0=hide, 1=show
textstring	An URL-encoded string	Set the text shown in the image.
textcolor	black, white	Set the color of the text shown in the image.
textbackgroundcolor	black,	Set the color of the text

	white, transparent, semitransparent	background shown in the image.
`rotation`	0, 90, 180, 270 [1]	Rotate the image clockwise.
`textpos`	0, 1	The position of the string shown in the image. 0=top, 1=bottom
`overlayimage`	0, 1	Enable/disable overlay image. 0=disable, 1=enable
`overlaypos`	<xoffset>[1]x<yoffset>[1]	Set the position of the overlay image.
`duration`	unsigned integer	Set the number of seconds the video will generate and push. 0=unlimited
`nbroffframes`	unsigned integer	Set the number of frames the server will generate and push. 0=unlimited
`fps`	unsigned integer	Set the frame rate from the server. 0=unlimited

[1]Values are product dependent. Please refer to the product specification.

[2]A stream profile is a set of video stream parameters (including videocodec) and is defined in the HTTP API or the web GUI. Supported stream profile names are stored in the StreamProfile.S#.Name parameters. It is possible override parameter values saved in a stream profile by specifying new values after the stream profile. See VAPIX® Stream Profile API for details.

H.264 and MPEG-4 Part 2 streams support the following additional parameters.

Parameter	Valid values	Description
`videobitrate`	integer	The rate (in kbits/s) at which video is requested. 0=Variable bit rate >0=Constant bit rate with the given target bit rate *Default*: 0
`videomaxbitrate`	integer	Maximum bit rate (in kbits/s) for bit rate control. *Default*: 0
`videobitratepriority`	none, framerate	The priority when rate control is used.

`videokeyframeinterval`	integer	Corresponds to the GOV length setting in the web GUI. *Default*: 32, but product and codec dependent

MJPEG streams support the additional parameter

Parameter	Valid values	Description
`squarepixel`	0, 1	Enable/disable square pixel correction. Applies only to video encoders.

6 References

RTSP Protocol

- Real Time Streaming Protocol – RFC 2326

SDP Protocol

- Session Description Protocol – RFC 2327

HTTP Protocol

- Hypertext Transfer Protocol – HTTP/1.0

External application programming interfaces (Client side)

VAPIX®: http://www.axis.com/techsup/cam_servers/dev/cam_http_api_index.php

- VAPIX®, HTTP API
- VAPIX®, Parameter specification
- Axis Video Product Release Notes
- VAPIX® Stream Profile API