

Real Time Streaming Protocol (RTSP)

Status of this Memo

This document is an Internet-Draft. Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as “work in progress”.

To learn the current status of any Internet-Draft, please check the “lid-abstracts.txt” listing contained in the Internet-Drafts Shadow Directories on ftp.is.co.za (Africa), nic.nordu.net (Europe), munnari.oz.au (Pacific Rim), ds.internic.net (US East Coast), or ftp.isi.edu (US West Coast).

Distribution of this document is unlimited.

Copyright Notice

Copyright (c) The Internet Society (1998). All Rights Reserved.

Abstract

The Real Time Streaming Protocol, or RTSP, is an application-level protocol for control over the delivery of data with real-time properties. RTSP provides an extensible framework to enable controlled, on-demand delivery of real-time data, such as audio and video. Sources of data can include both live data feeds and stored clips. This protocol is intended to control multiple data delivery sessions, provide a means for choosing delivery channels such as UDP, multicast UDP and TCP, and provide a means for choosing delivery mechanisms based upon RTP (RFC 1889).

Contents

1	Introduction	5
1.1	Purpose	5
1.2	Requirements	6
1.3	Terminology	6
1.4	Protocol Properties	8
1.5	Extending RTSP	9
1.6	Overall Operation	10
1.7	RTSP States	10
1.8	Relationship with Other Protocols	11
2	Notational Conventions	11
3	Protocol Parameters	11
3.1	RTSP Version	11
3.2	RTSP URL	12

3.3	Conference Identifiers	13
3.4	Session Identifiers	13
3.5	SMPTE Relative Timestamps	13
3.6	Normal Play Time	13
3.7	Absolute Time	14
3.8	Option Tags	14
3.8.1	Registering New Option Tags with IANA	15
4	RTSP Message	15
4.1	Message Types	15
4.2	Message Headers	15
4.3	Message Body	16
4.4	Message Length	16
5	General Header Fields	16
6	Request	16
6.1	Request Line	17
6.2	Request Header Fields	17
7	Response	18
7.1	Status-Line	18
7.1.1	Status Code and Reason Phrase	18
7.1.2	Response Header Fields	20
8	Entity	20
8.1	Entity Header Fields	20
8.2	Entity Body	22
9	Connections	22
9.1	Pipelining	22
9.2	Reliability and Acknowledgements	22
10	Method Definitions	23
10.1	OPTIONS	23
10.2	DESCRIBE	24
10.3	ANNOUNCE	25
10.4	SETUP	26
10.5	PLAY	27
10.6	PAUSE	28
10.7	TEARDOWN	29
10.8	GET_PARAMETER	30
10.9	SET_PARAMETER	30
10.10	REDIRECT	31
10.11	RECORD	31
10.12	Embedded (Interleaved) Binary Data	32

11	Status Code Definitions	33
11.1	Success 2xx	33
11.1.1	250 Low on Storage Space	33
11.2	Redirection 3xx	33
11.3	Client Error 4xx	33
11.3.1	405 Method Not Allowed	33
11.3.2	451 Parameter Not Understood	33
11.3.3	452 Conference Not Found	33
11.3.4	453 Not Enough Bandwidth	33
11.3.5	454 Session Not Found	33
11.3.6	455 Method Not Valid in This State	34
11.3.7	456 Header Field Not Valid for Resource	34
11.3.8	457 Invalid Range	34
11.3.9	458 Parameter Is Read-Only	34
11.3.10	459 Aggregate Operation Not Allowed	34
11.3.11	460 Only Aggregate Operation Allowed	34
11.3.12	461 Unsupported Transport	34
11.3.13	462 Destination Unreachable	34
11.3.14	551 Option not supported	34
12	Header Field Definitions	34
12.1	Accept	35
12.2	Accept-Encoding	35
12.3	Accept-Language	35
12.4	Allow	35
12.5	Authorization	35
12.6	Bandwidth	37
12.7	Blocksize	37
12.8	Cache-Control	37
12.9	Conference	39
12.10	Connection	39
12.11	Content-Base	39
12.12	Content-Encoding	39
12.13	Content-Language	39
12.14	Content-Length	39
12.15	Content-Location	40
12.16	Content-Type	40
12.17	CSeq	40
12.18	Date	40
12.19	Expires	40
12.20	From	41
12.21	Host	41
12.22	If-Match	41
12.23	If-Modified-Since	41
12.24	Last-Modified	41

12.25	Location	41
12.26	Proxy-Authenticate	42
12.27	Proxy-Require	42
12.28	Public	42
12.29	Range	42
12.30	Referer	42
12.31	Retry-After	43
12.32	Require	43
12.33	RTP-Info	43
12.34	Scale	44
12.35	Speed	45
12.36	Server	45
12.37	Session	45
12.38	Timestamp	46
12.39	Transport	46
12.40	Unsupported	48
12.41	User-Agent	48
12.42	Vary	49
12.43	Via	49
12.44	WWW-Authenticate	49
13	Caching	49
14	Examples	49
14.1	Media on Demand (Unicast)	50
14.2	Streaming of a Container file	51
14.3	Single Stream Container Files	54
14.4	Live Media Presentation Using Multicast	55
14.5	Playing media into an existing session	56
14.6	Recording	57
15	Syntax	59
15.1	Base Syntax	60
16	Security Considerations	61
A	RTSP Protocol State Machines	62
A.1	Client State Machine	63
A.2	Server State Machine	63
B	Interaction with RTP	64
C	Use of SDP for RTSP Session Descriptions	65
C.1	Definitions	65
C.1.1	Control URL	65
C.1.2	Media streams	66

C.1.3	Payload type(s)	66
C.1.4	Format-specific parameters	66
C.1.5	Range of presentation	66
C.1.6	Time of availability	67
C.1.7	Connection Information	67
C.1.8	Entity Tag	67
C.2	Aggregate Control Not Available	67
C.3	Aggregate Control Available	68
D	Minimal RTSP implementation	68
D.1	Client	68
D.1.1	Basic Playback	69
D.1.2	Authentication-enabled	70
D.2	Server	70
D.2.1	Basic Playback	70
D.2.2	Authentication-enabled	71
E	Author Addresses	71
F	Acknowledgements	72

1 Introduction

1.1 Purpose

The Real-Time Streaming Protocol (RTSP) establishes and controls either a single or several time-synchronized streams of continuous media such as audio and video. It does not typically deliver the continuous streams itself, although interleaving of the continuous media stream with the control stream is possible (see Section 10.12). In other words, RTSP acts as a “network remote control” for multimedia servers.

The set of streams to be controlled is defined by a presentation description. This memorandum does not define a format for a presentation description.

There is no notion of an RTSP connection; instead, a server maintains a session labeled by an identifier. An RTSP session is in no way tied to a transport-level connection such as a TCP connection. During an RTSP session, an RTSP client may open and close many reliable transport connections to the server to issue RTSP requests. Alternatively, it may use a connectionless transport protocol such as UDP.

The streams controlled by RTSP may use RTP [1], but the operation of RTSP does not depend on the transport mechanism used to carry continuous media.

The protocol is intentionally similar in syntax and operation to HTTP/1.1 [2] so that extension mechanisms to HTTP can in most cases also be added to RTSP. However, RTSP differs in a number of important aspects from HTTP:

- RTSP introduces a number of new methods and has a different protocol identifier.
- An RTSP server needs to maintain state by default in almost all cases, as opposed to the stateless nature of HTTP.
- Both an RTSP server and client can issue requests.

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.