# 流媒体协议介绍（rtp/rtcp/rtsp/rtmp/mms/hls）

[飞一样的成长](javascript:void(0);) 2021年09月02日 08:40

## **1、RTP**

Real-time Transport Protocol)是用于Internet上针对多媒体数据流的一种传输层协议。RTP协议详细说明了在互联网上传递音频和视频的标准数据包格式。RTP协议常用于流媒体系统（配合RTCP协议），视频会议和一键通（Push to Talk）系统（配合H.323或SIP），使它成为IP电话产业的技术基础。RTP协议和RTP控制协议RTCP一起使用，而且它是建立在UDP协议上的。

RTP 本身并没有提供按时发送机制或其它服务质量（QoS）保证，它依赖于低层服务去实现这一过程。RTP 并不保证传送或防止无序传送，也不确定底层网络的可靠性。RTP 实行有序传送， RTP 中的序列号允许接收方重组发送方的包序列，同时序列号也能用于决定适当的包位置，例如：在视频解码中，就不需要顺序解码。

RTP 由两个紧密链接部分组成：RTP ― 传送具有实时属性的数据；RTP 控制协议（RTCP） ― 监控服务质量并传送正在进行的会话参与者的相关信息。

## **2、RTCP**

实时传输控制协议（Real-time Transport Control Protocol或RTP Control Protocol或简写RTCP）是实时传输协议（RTP）的一个姐妹协议。RTCP为RTP媒体流提供信道外（out-of-band）控制。RTCP本身并不传输数据，但和RTP一起协作将多媒体数据打包和发送。RTCP定期在流多媒体会话参加者之间传输控制数据。RTCP的主要功能是为RTP所提供的服务质量（Quality of Service）提供反馈。

RTCP收集相关媒体连接的统计信息，例如：传输字节数，传输分组数，丢失分组数，jitter，单向和双向网络延迟等等。网络应用程序可以利用RTCP所提供的信息试图提高服务质量，比如限制信息流量或改用压缩比较小的编解码器。RTCP本身不提供数据加密或身份认证。SRTCP可以用于此类用途。

## **3、SRTP & SRTCP**

安全实时传输协议（Secure Real-time Transport Protocol或SRTP）是在实时传输协议（Real-time Transport Protocol或RTP）基础上所定义的一个协议，旨在为单播和多播应用程序中的实时传输协议的数据提供加密、消息认证、完整性保证和重放保护。它是由David Oran（思科）和Rolf Blom（爱立信）开发的，并最早由IETF于2004年3月作为RFC3711发布。

由于实时传输协议和可以被用来控制实时传输协议的会话的实时传输控制协议（RTP Control Protocol或RTCP）有着紧密的联系，安全实时传输协议同样也有一个伴生协议，它被称为安全实时传输控制协议（Secure RTCP或SRTCP）；安全实时传输控制协议为实时传输控制协议提供类似的与安全有关的特性，就像安全实时传输协议为实时传输协议提供的那些一样。

在使用实时传输协议或实时传输控制协议时，使不使用安全实时传输协议或安全实时传输控制协议是可选的；但即使使用了安全实时传输协议或安全实时传输控制协议，所有它们提供的特性（如加密和认证）也都是可选的，这些特性可以被独立地使用或禁用。唯一的例外是在使用安全实时传输控制协议时，必须要用到其消息认证特性。

## **4、RTSP**

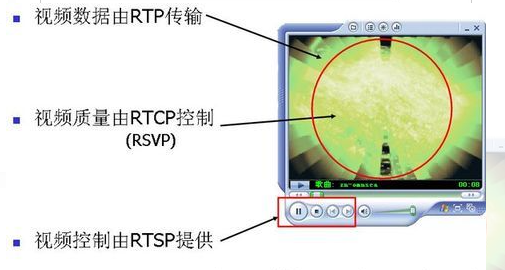
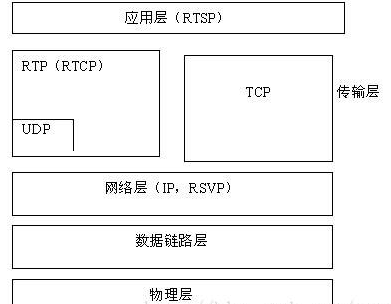
它是由Real Networks和Netscape共同提出的。该协议定义了一对多应用程序如何有效地通过IP网络传送多媒体数据。RTSP提供了一个可扩展框架，使实时数据，如音频与视频的受控、点播成为可能。数据源包括现场数据与存储在剪辑中的数据。该协议目的在于控制多个数据发送连接，为选择发送通道，如UDP、多播UDP与TCP提供途径，并为选择基于RTP上发送机制提供方法。

RTSP（Real Time Streaming Protocol）是用来控制声音或影像的多媒体串流协议，并允许同时多个串流需求控制，传输时所用的网络通讯协定并不在其定义的范围内，服务器端可以自行选择使用TCP或UDP来传送串流内容，它的语法和运作跟HTTP 1.1类似，但并不特别强调时间同步，所以比较能容忍网络延迟。而前面提到的允许同时多个串流需求控制（Multicast），除了可以降低服务器端的网络用量，更进而支持多方视讯会议（Video Conference）。因为与HTTP1.1的运作方式相似，所以代理服务器《Proxy》的快取功能《Cache》也同样适用于RTSP，并因RTSP具有重新导向功能，可视实际负载情况来转换提供服务的服务器，以避免过大的负载集中于同一服务器而造成延迟。

RTSP 和RTP的关系：

RTP不象http和ftp可完整的下载整个影视文件，它是以固定的数据率在网络上发送数据，客户端也是按照这种速度观看影视文件，当影视画面播放过后，就不可以再重复播放，除非重新向服务器端要求数据。

RTSP与RTP最大的区别在于：RTSP是一种双向实时数据传输协议，它允许客户端向服务器端发送请求，如回放、快进、倒退等操作。当然，RTSP可基于RTP来传送数据，还可以选择TCP、UDP、组播UDP等通道来发送数据，具有很好的扩展性。它时一种类似与http协议的网络应用层协议。目前碰到的一个应用：服务器端实时采集、编码并发送两路视频，客户端接收并显示两路视频。由于客户端不必对视频数据做任何回放、倒退等操作，可直接采用UDP+RTP+组播实现。



RTP：实时传输协议（Real-time Transport Protocol）

RTP/RTCP是实际传输数据的协议

RTP传输音频/视频数据，如果是PLAY，Server发送到Client端，如果是RECORD，可以由Client发送到Server

整个RTP协议由两个密切相关的部分组成：RTP数据协议和RTP控制协议（即RTCP）

RTSP：实时流协议（Real Time Streaming Protocol，RTSP）

RTSP的请求主要有DESCRIBE,SETUP,PLAY,PAUSE,TEARDOWN,OPTIONS等，顾名思义可以知道起对话和控制作用

RTSP的对话过程中SETUP可以确定RTP/RTCP使用的端口，PLAY/PAUSE/TEARDOWN可以开始或者停止RTP的发送，等等

RTP/RTCP是实际传输数据的协议

RTCP包括Sender Report和Receiver Report，用来进行音频/视频的同步以及其他用途，是一种控制协议

SDP：会话描述协议（SDP）为会话通知、会话邀请和其它形式的多媒体会话初始化等目的提供了多媒体会话描述。会话目录用于协助多媒体会议的通告，并为会话参与者传送相关设置信息。SDP 即用于将这种信息传输到接收端。SDP 完全是一种会话描述格式 ― 它不属于传输协议 ― 它只使用不同的适当的传输协议，包括会话通知协议（SAP）、会话初始协议（SIP）、实时流协议（RTSP）、MIME 扩展协议的电子邮件以及超文本传输协议（HTTP）。

SDP 的设计宗旨是通用性，它可以应用于大范围的网络环境和应用程序，而不仅仅局限于组播会话目录，但 SDP 不支持会话内容或媒体编码的协商。

在因特网组播骨干网（Mbone）中，会话目录工具被用于通告多媒体会议，并为参与者传送会议地址和参与者所需的会议特定工具信息，这由 SDP 完成。SDP 连接好会话后，传送足够的信息给会话参与者。SDP 信息发送利用了会话通知协议（SAP），它周期性地组播通知数据包到已知组播地址和端口处。这些信息是 UDP 数据包，其中包含 SAP 协议头和文本有效载荷（text payload）。这里文本有效载荷指的是 SDP 会话描述。此外信息也可以通过电子邮件或 WWW （World Wide Web） 进行发送。

* SDP 文本信息包括：

会话名称和意图；

会话持续时间；

构成会话的媒体；

有关接收媒体的信息（地址等）。

* 协议结构

SDP 信息是文本信息，采用 UTF-8 编 码中的 ISO 10646 字符集。SDP 会话描述如下：（标注 \* 符号的表示可选字段）：

v = （协议版本）

o = （所有者/创建者和会话标识符）

s = （会话名称）

i = \* （会话信息）

u = \* （URI 描述）

e = \* （Email 地址）

p = \* （电话号码）

c = \* （连接信息 ― 如果包含在所有媒体中，则不需要该字段）

b = \* （带宽信息）

一个或更多时间描述（如下所示）：

z = \* （时间区域调整）

k = \* （加密密钥）

a = \* （0 个或多个会话属性行）

0个或多个媒体描述（如下所示）

* 时间描述：

t = （会话活动时间）

r = \* （0或多次重复次数）

* 媒体描述：

m = （媒体名称和传输地址）

i = \* （媒体标题）

c = \* （连接信息 — 如果包含在会话层则该字段可选）

b = \* （带宽信息）

k = \* （加密密钥）

a = \* （0 个或多个会话属性行）

## **5、RTMP/RTMPS**

RTMP(Real Time Messaging Protocol)实时消息传送协议是Adobe Systems公司为Flash播放器和服务器之间音频、视频和数据传输 开发的开放协议。

它有三种变种：

1)工作在TCP之上的明文协议，使用端口1935；

2)RTMPT封装在HTTP请求之中，可穿越防火墙；

3)RTMPS类似RTMPT，但使用的是HTTPS连接；

RTMP协议(Real Time Messaging Protocol)是被Flash用于对象,视频,音频的传输.这个协议建立在TCP协议或者轮询HTTP协议之上.

RTMP协议就像一个用来装数据包的容器,这些数据既可以是AMF格式的数据,也可以是FLV中的视/音频数据.一个单一的连接可以通过不同的通道传输多路网络流.这些通道中的包都是按照固定大小的包传输的.

## **6、mms**

MMS (Microsoft Media Server Protocol)，中文“微软媒体服务器协议”，用来访问并流式接收 Windows Media 服务器中 .asf 文件的一种协议。MMS 协议用于访问 Windows Media 发布点上的单播内容。MMS 是连接 Windows Media 单播服务的默认方法。若观众在 Windows Media Player 中键入一个 URL 以连接内容，而不是通过超级链接访问内容，则他们必须使用MMS 协议引用该流。MMS的预设埠（端口）是1755

当使用 MMS 协议连接到发布点时，使用协议翻转以获得最佳连接。“协议翻转”始于试图通过 MMSU 连接客户端。MMSU 是 MMS 协议结合 UDP 数据传送。如果 MMSU 连接不成功，则服务器试图使用 MMST。MMST 是 MMS 协议结合 TCP 数据传送。

如果连接到编入索引的 .asf 文件，想要快进、后退、暂停、开始和停止流，则必须使用 MMS。不能用 UNC 路径快进或后退。若您从独立的 Windows Media Player 连接到发布点，则必须指定单播内容的 URL。若内容在主发布点点播发布，则 URL 由服务器名和 .asf 文件名组成。

例如：mms://windows\_media\_server/sample.asf。

其中 windows\_media\_server 是 Windows Media 服务器名，sample.asf 是您想要使之转化为流的 .asf 文件名。

若您有实时内容要通过广播单播发布，则该 URL 由服务器名和发布点别名组成。

例如：mms://windows\_media\_server/LiveEvents。

这里 windows\_media\_server 是 Windows Media 服务器名，而 LiveEvents 是发布点名

## **7、HLS**

HTTP Live Streaming（HLS）是苹果公司(Apple Inc.)实现的基于HTTP的流媒体传输协议，可实现流媒体的直播和点播，主要应用在iOS系统，为iOS设备（如iPhone、iPad）提供音视频直播和点播方案。HLS点播，基本上就是常见的分段HTTP点播，不同在于，它的分段非常小。

相对于常见的流媒体直播协议，例如RTMP协议、RTSP协议、MMS协议等，HLS直播最大的不同在于，直播客户端获取到的，并不是一个完整的数据流。HLS协议在服务器端将直播数据流存储为连续的、很短时长的媒体文件（MPEG-TS格式），而客户端则不断的下载并播放这些小文件，因为服务器端总是会将最新的直播数据生成新的小文件，这样客户端只要不停的按顺序播放从服务器获取到的文件，就实现了直播。由此可见，基本上可以认为，HLS是以点播的技术方式来实现直播。由于数据通过HTTP协议传输，所以完全不用考虑防火墙或者代理的问题，而且分段文件的时长很短，客户端可以很快的选择和切换码率，以适应不同带宽条件下的播放。不过HLS的这种技术特点，决定了它的延迟一般总是会高于普通的流媒体直播协议。

根据以上的了解要实现HTTP Live Streaming直播，需要研究并实现以下技术关键点：

采集视频源和音频源的数据

对原始数据进行H264编码和AAC编码

视频和音频数据封装为MPEG-TS包

HLS分段生成策略及m3u8索引文件

HTTP传输协议

我是txp，我们下期见！

文章来源：

<https://blog.csdn.net/tttyd/article/details/12032357?utm_medium=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlogCommendFromMachineLearnPai2%7Edefault-10.essearch_pc_relevant&depth_1-utm_source=distribute.pc_relevant.none-task-blog-2%7Edefault%7EBlo>