

ARUBA WLAN 设置语音流量最高优先级，避免因网络阻塞造成 SIP、VoIP 体验过差

系统预定义的 **net service**

```
net service svc-rtsp tcp 554 alg rtsp
net service svc-sip-tcp tcp 5060 alg sip
net service svc-sip-udp udp 5060 alg sip
```

设置 **session ACL**

```
ip access-list session abc
user any svc-rtsp permit log dot1p-priority 7
user any svc-sip-udp permit log dot1p-priority 7
user any svc-sip-tcp permit log dot1p-priority 7
```

设置 **role**

```
user-role xyz
access-list session abc
```

aaa profile 调用 **user role**

```
aaa profile myprofile
initial-role/dot1x-default-role/mac-default-role xyz
```

RTSP 的应用

实时流协议（Real Time Streaming Protocol, RTSP）是一种网络应用协议，专为娱乐和通信系统的使用，以控制流媒体服务器。该协议用于创建和控制终端之间的媒体会话。媒体服务器的客户端发布 VCR 命令，例如播放，录制和暂停，以便于实时控制从服务器到客户端（视频点播）或从客户端到服务器（语音录音）的媒体流。

流数据本身的传输不是 RTSP 的任务。大多数 RTSP 服务器使用实时传输协议（RTP）和实时传输控制协议（RTCP）结合媒体流传输。然而，一些供应商实现专有传输协议。例如，RealNetworks 公司的 RTSP 服务器软件也使用 RealNetworks 的专有实时数据传输（RDT）

协议简介

RTSP 是 TCP/IP 协议体系中的一个应用层协议，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 在体系结构上位于 RTP 和 RTCP 之上，它使用 TCP 或 UDP 完成数据传输。HTTP 与 RTSP 相比，HTTP 传送 HTML，而 RTSP 传送的是多媒体数据。

应用层	SDP	
	RTSP	
传输层		RTP
	TCP	UDP
网络层	IP	

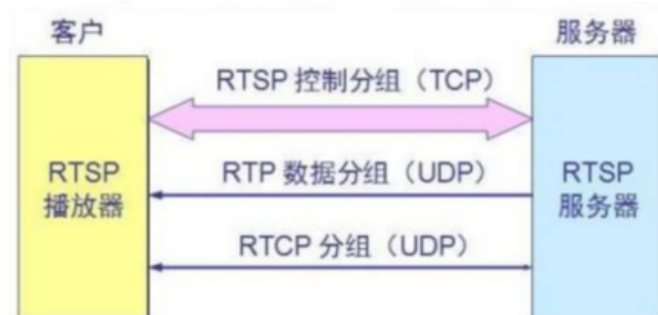
RTSP 是基于文本的协议，采用 ISO10646 字符集，使用 UTF-8 编码方案。行以 CRLF 中断，包括消息类型、消息头、消息体和消息长。但接收者本身可将 CR 和 LF 解释成行终止符。基于文本的协议使其以自描述方式增加可选参数更容易，接口中采用 SDP 作为描述语言。

RTSP 是应用级协议，控制实时数据的发送。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、组播 UDP 与 TCP，提供途径，并为选择基于 RTP 上发送机制提供方法。

RTSP 建立并控制一个或几个时间同步的连续流媒体。尽管连续媒体流与控制流交换是可能的，通常它本身并不发送连续流。换言之，RTSP 充当多媒体服务器的网络远程控制。RTSP 连接没有绑定到传输层连接，如 TCP。在 RTSP 连接期间，RTSP 用户可打开或关闭多个对服务器的可传输连接以发出 RTSP 请求。此外，可使用无连接传输协议，如 UDP。RTSP 流控制的流可能用到 RTP，但 RTSP 操作并不依赖于携带连续媒体的传输机制

协议支持的操作

如下：



RTSP 协议支持

(1) 从媒体服务器上检索媒体：用户可通过 HTTP 或其它方法提交一个演示描述。如演示是组播，演示式就包含用于连续媒体的组播地址和端口。如演示仅通过单播发送给用户，用户为了安全应提供目的地址。

(2) 媒体服务器邀请进入会议：媒体服务器可被邀请参加正进行的会议，或回放媒体，或记录其中一部分，或全部。这种模式在分布式教育应用上很有用，会议中几方可轮流按远程控制按钮。

(3) 将媒体加到现成讲座中：如服务器告诉用户可获得附加媒体内容，对现场讲座显得尤其有用。如 HTTP/1.1 中类似，RTSP 请求可由代理、通道与缓存处理。

概念

集合控制

对多个流的同时控制。对音频/视频来讲，客户端仅需发送一条播放或者暂停消息就可同时控制音频流和视频流。

实体 (Entity)

作为请求或者回应的有效负荷传输的信息。由以实体标题域 (entity-header field) 形式存在的元信息和以实体主体 (entity body) 形式存在的内容组成。

如不受请求方法或响应状态编码限制，请求和响应信息可传输实体，实体则由实体头文件和实体体组成，有些响应仅包括实体头。在此，根据谁发送实体、谁接收实体，发送者和接收者可分别指用户和服务器。

实体头定义实体体可选元信息，如没有实体体，指请求标识的资源。扩展头机制允许定义附加实体头段，而不用改变协议，但这些段不能假定接收者能识别。不可识别头段应被接收者忽略，而让代理转发。

容器文件 (Containerfile)

可以容纳多个媒体流的文件。RTSP 服务器可以为这些容器文件提供集合控制。

RTSP 会话

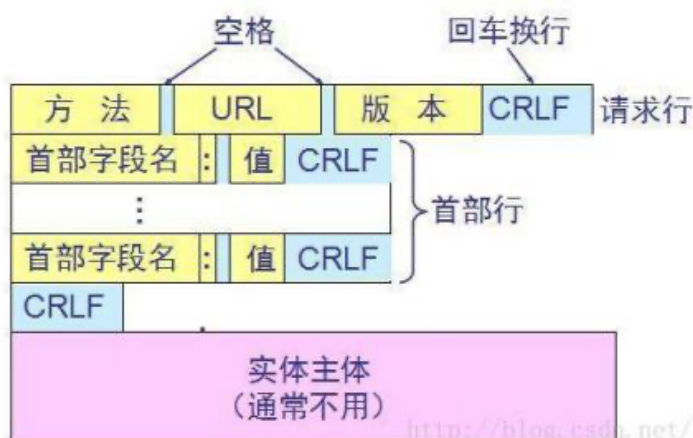
RTSP 交互的全过程。对一个电影的观看过程, 会话 (session) 包括由客户端建立媒体流传输机制 (SETUP), 使用播放 (PLAY) 或录制 (RECORD) 开始传送流, 用停止 (TEARDOWN) 关闭流。

协议格式

RTSP 中所有的操作都是通过服务器和客户端的消息应答机制完成的, 其中消息包括请求和应答两种, RTSP 是对称的协议, 客户机和服务器都可以发送和回应请求。RTSP 是一个基于文本的协议, 它使用 UTF -8 编码 (RFC2279) 和 ISO10646 字符序列, 采用 RFC882 定义的通用消息格式, 每个语句行由 CRLF 结束。RTSP 的消息包括请求和应答两类。

请求消息

请求消息由请求行、标题行中的各种标题域和主体实体组成。请求行和标题行由 ASCII 字符组成。请求消息的格式如下图所示



其中方法包括 OPTIONS、DESCRIBE、SETUP、PLAY、TEARDOWN 等。URL 是接收方的地址，例如:rtsp://192.168.0.1/video.264。RTSP 版本一般都是 RTSP/1.0。每行后面的 CRLF 表示回车换行，需要接收端有相应的解析，最后一个消息头需要有两个 CRLF。消息体是可选的，有的请求消息并不带消息体

应答消息

RTSP 应答消息的格式如下图所示

