

流中继到每个参与者 [Zhang X 2012]。你可能想知道为什么每个参与者向服务器而不是向每个其他  $N-1$  其他参与者直接发送其视频流的副本呢？的确，对两种方法而言， $N(N-1)$  个视频流正由会议中的  $N$  个参与者共同接收。其原因是，在大多数接入链路中上行链路带宽比下行链路带宽要低得多，上行链路可能不能支持使用 P2P 方法的  $N-1$  条流。

诸如 Skype、QQ 和 Google Talk 等 VoIP 系统引入了新的隐私性关注。具体而言，当 Alice 和 Bob 经过 VoIP 通信，Alice 能够嗅探到 Bob 的 IP 地址，进而使用地理定位服务 [MaxMind 2012; Quova 2012] 来确定 Bob 的当前位置和 ISP（例如他的工作或家庭 ISP）。事实上，Alice 使用 Skype 能够在呼叫创建期间阻挡特定的分组传输，这样她获得 Bob 当前（比如说每小时）的 IP 地址，而 Bob 并不知道他正被跟踪，并且 Alice 不在 Bob 的接触列表上。此外，从 Skype 发现的 IP 地址能被关联到在 BitTorrent 中发现的 IP 地址，因此 Alice 能够确定 Bob 正在下载的文件 [LeBlond 2011]。此外，通过进行流中分组长度的流量分析，可能部分解密一个 Skype 呼叫 [White 2011]。

## 7.4 实时会话式应用的协议

实时会话式应用（包括 VoIP 和视频会议）引人入胜并且非常流行。因此标准机构如 IETF 和 ITU 多年来一直忙于（而且要继续忙下去！）苦心推敲这类应用的标准是毫不奇怪的。借助于实时会话式应用的适当标准，各个独立公司正在创造新的能够互相操作的产品。在本节中，我们探讨用于实时会话式应用的 RTP 和 SIP。这两个标准正广泛地应用于工业产品中。

### 7.4.1 RTP

在前一节中，我们知道 VoIP 应用的发送端在将块传递给运输层之前为它们附加上首部字段。这些首部字段包括了序号和时间戳。因为大多数多媒体网络应用能够利用序号和时间戳，因此有一个包括音频/视频数据、序号、时间戳以及其他潜在有用字段的标准分组结构是方便的。定义在 RFC 3550 中的 RTP 就是这样一个标准。RTP 能够用于传输通用格式，如用于声音的 PCM、ACC 和 MP3，用于视频的 MPEG 和 H. 263。它也可以用于传输专用的声音和视频格式。目前，RTP 在许多产品和研究原型中得到广泛实现。它也是其他重要的实时交互协议（如 STP）的补充。

本节我们介绍 RTP。我们也鼓励读者去访问 Henning Schulzrinne 的 RTP 站点 [Schulzrinne-RTP 2012]，该网站提供有关这个主题的很多信息。读者也可以访问 RAT 站点 [RAT 2012]，它记载了使用 RTP 的 VoIP 应用。

#### 1. RTP 基础

RTP 通常运行在 UDP 之上。发送端在 RTP 分组中封装媒体块，然后在 UDP 报文段中封装该分组，然后将该报文段递交给 IP。接收端从 UDP 报文段中提取出这个 RTP 分组，然后从 RTP 分组中提取出媒体块，并将这个块传递给媒体播放器来解码和呈现。

举例来说，考虑使用 RTP 来传输语音。假设语音源采用了 64kbps 的 PCM 编码（也就是采样、量化和数字化）。再假设应用程序在 20ms 块中收集这些编码数据，也就是一个块中有 160 字节。发送端在每个语音数据块的前面加上一个 RTP 首部（RTP header），这个首部包括音频编码的类型、序号和时间戳。RTP 首部通常是 12 字节。音频块和 RTP 首部一起形成 RTP 分组（RTP packet）。然后向 UDP 套接字接口发送该 RTP 分组。在接收端，