12

Media Transport

Establishing media sessions is one of the most important applications of SIP in Internet communications. An understanding of the issues relating to media transport of voice, video, DTMF, and text helps motivate the media negotiation capabilities of SIP. In this chapter, the Real-Time Transport Protocol (RTP) will be introduced as the protocol that transports actual media samples. The basic steps in audio and video media encoding and decoding are discussed, along with the effects of common Internet impairments. The RTP header format is covered along with common RTP topologies. The RTP Control Protocol (RTCP) is introduced as a way to monitor call quality. RTP profiles and common codes are discussed—both PSTN codecs and Internet codecs. Common audio and video codecs are discussed. Finally, DTMF transport and conversational text are covered.

12.1 Real-Time Transport Protocol (RTP)

Real-Time Transport Protocol [1] was developed to enable the transport of real-time datagrams containing voice, video, or other information over IP. RTP was not the first VoIP protocol used on the Internet. Network Voice Protocol (NVP) [2] was implemented in 1973 to carry real-time voice communications over the Internet. Early versions of RTP, first implemented in 1992, were used to transport voice over the Internet's multicast backbone (MBONE). Both H.323 and SIP use RTP for media transport, making it the most common standard for Internet communications.

RTP is defined by the IETF proposed standard RFC 3550 (which updates the original RFC 1889). RTP does not provide any quality of service over the IP network—RTP packets are handled the same as all other packets in an IP

network. However, RTP allows for the detection of some of the impairments introduced by an IP network, such as:

- Packet loss;
- Variable transport delay;
- Out of sequence packet arrival;
- Asymmetric routing.

Here is how RTP fits into the common media processing steps.

1. *Coding.* The coding step involves analog to digital conversion (A/D), which is implemented by low pass filtering, followed by sampling. The determination of how many bits per sample is specific to a particular codec (coder/decoder) algorithm. The particular codec used is transported by RTP in the payload type field. The sampling rate is carried in the offer/answer exchange in SDP, which negotiates the media session.
2. *Packetization.* The packetization step involves breaking the codec sample data into individual datagrams for transport. The determination of packet size is based on a tradeoff between packetization delay (how many sampling intervals must pass before enough data is ready for the datagram) and transport efficiency (each datagram has the fixed overhead of the RTP header and lower layer headers). Typically, packet sizes are chosen to be small so that packetization time is around 20 ms to 30 ms. Packetization involves adding the RTP header to the codec payload.
3. *Tranport.* RTP, as the name suggests, has a real-time nature, which requires a minimum latency (delay) across the Internet. There is never time to detect a missing packet, signal the loss, and wait for a retransmission. This might be possible for nonreal-time streaming media, but not real-time media. As a result, RTP does not usually use TCP transport but instead uses UDP transport. As a result, datagrams may be lost or may arrive out of sequence. Various fields in the RTP header field allow the detection of this.
4. *Depacketization.* The depacketization step involves removing the RTP header from the codec payload.
5. *Buffering.* The buffering step involves storing or buffering the codec samples before beginning playback. The choice of the buffer size for this step is critical for media quality. Too short a buffer will result in the buffer emptying and gaps in the media playback, while too long a buffer will introduce unpleasant latency. Adapting the size of the play-

back buffer when jitter or delay variation is occurring is best for media quality.

6. *Decoding*. The decoding step involves sending the codec packets to the codec algorithm. The right codec is chosen based on the received payload type in the RTP header.
7. *Playback*. The playback step involves rendering the media to the user as audio, video, or perhaps text (as we shall see in real-time text or text over IP, ToIP).

In terms of media quality, the two most important factors are the packet loss rate and the end-to-end latency. Lost packets mean gaps in the playback stream that the codec algorithm must try to compensate for. Different codecs use different techniques for packet loss concealment (PLC). For example, interpolation can be used to try to predict the missing samples based on received samples either side of the lost ones. A simple replay algorithm can be useful for some media types. Silence or comfort noise insertion can be used to prevent users noticing the dead air of lost samples. Some codecs employ forward error correction (FEC), which allows partial reconstruction of missing packets under low loss conditions. Note that packets aren't really "lost" on the Internet, instead they are discarded by routers in the Internet due to congestion, or discarded by the RTP stack due to out of order arrival or late arrival resulting in missing their playback interval.

The end-to-end latency of real-time communications in general must be kept less than 150 ms. Longer latency than this affects the perceived quality of the call, resulting in users interrupting each other and starting and stopping when both parties speak at the same time. There are many sources of delay in the media path. The codec itself introduces delay as it gathers at least one sample or frame before beginning coding and decoding. The packetization step introduces a delay of around 20 ms, the time it takes to gather a full packet's worth of data before sending it over the Internet. Transport delays are added by the routers and switches that forward and process the IP packets across the Internet. And finally, the buffering delay of the receiver to deal with jitter or delay variation also introduces latency. Some of these sources are shown in Figure 12.1.

Real-Time Transport Protocol (RTP) is an application layer protocol that uses UDP for transport over IP. RTP is not text encoded, but uses a bit-oriented header similar to UDP and IP. RTP version 0 is only used by the vat audio tool for MBONE broadcasts. Version 1 was a pre-RFC implementation and is not in use. The current RTP version 2 packet header has 12 octets. RTP was designed to be very general; most of the headers are only loosely defined in the standard; the details are left to profile documents. The header contains:

- Version (***v***): This 2-bit field is set to 2, the current version of RTP.

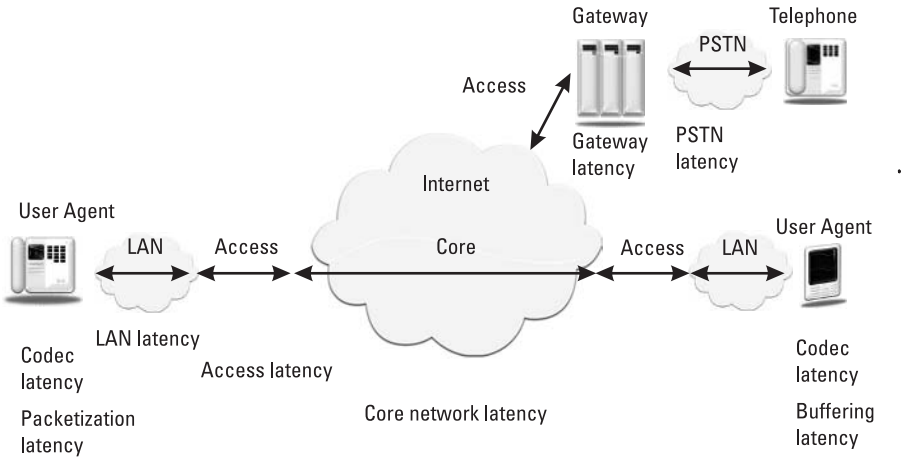

Figure 12.1 Sources of latency and packet loss on the Internet.

- Padding (`P`): If this bit is set, there are padding octets added to the end of the packet to make the packet a fixed length. This is most commonly used if the media stream is encrypted.
- Extension (`X`): If this bit is set, there is one additional extension following the header (giving a total header length of 14 octets). Extensions are defined by certain payload types.
- CSRC count (`CC`): This 4-bit field contains the number of content source identifiers (CSRC) that are present following the header. This field is only used by mixers that take multiple RTP streams and output a single RTP stream.
- Marker (`M`): This single bit is used to indicate the end of a complete frame in video, or the start of a talk-spurt in silence-suppressed speech.
- Payload type (`PT`): This 7-bit field defines the codec in use. The value of this field matches the profile number listed in the SDP.
- Sequence Number: This 16-bit field is incremented for each RTP packet sent and is used to detect missing/out of sequence packets.
- Timestamp: This 32-bit field indicates in relative terms the time when the payload was sampled. This field allows the receiver to remove jitter and to play back the packets at the right interval assuming sufficient buffering.
- Synchronization source identifier (`SSRCI`): This 32-bit field identifies the sender of the RTP packet. At the start of a session, each participant chooses an SSRC number randomly. Should two participants choose the same number, they each choose again until each party is unique.

- Contributing source identifier (***CSRC***): There can be none or up to 15 instances of this 32-bit field in the header. The number is set by the CSRC count (***CC***) header field. This field is only present if the RTP packet is being sent by a mixer, which has received RTP packets from a number of sources and sends out combined packets. A nonmulticast conference bridge would utilize this header.

RTP allows detection of a lost packet by a gap in the Sequence Number. Packets received out of sequence can be detected by out-of-sequence Sequence Numbers. Note that RTP allows detection of these transport-related problems but leaves it up to the codec to deal with the problem. For example, a video codec may compensate for the loss of a packet by repeating the last video frame, while an audio codec may play background noise for the interval. Variable delay or jitter can be detected by the Timestamp field. A continuous bit rate codec such as PCM will have a linearly increasing Timestamp. A variable bit rate codec, however, which sends packets at irregular intervals, will have an irregularly increasing Timestamp, which can be used to play back the packets at the correct interval.

RTP media sessions are unidirectional—they define how media is sent from the media source to the media sink. As such, a normal bidirectional media session is actually two RTP sessions, one in each direction.

In a multimedia session established with SIP, the information needed to select codecs and send the RTP packets to the right location is carried in the SDP message body. Under some scenarios, it can be desirable to change codecs during an RTP session. An example of this relates to the transport of dual tone multiple frequency (DTMF) digits. A low bit rate codec that is optimized for transmitting vocal sounds will not transport the superimposed sine waves of a DTMF signal without introducing significant noise, which may cause the DTMF digit receiver to fail to detect the digit. As a result, it is useful to switch to another codec when the sender detects a DTMF tone. Because an RTP packet contains the payload type, it is possible to change codecs on the fly without any signaling information being exchanged between the UAs. On the other hand, switching codecs in general should probably not be done without a SIP signaling exchange (re-***INVITE***) because the call could fail if one side switches to a codec that the other does not support. The SIP re-***INVITE*** message exchange allows this change in media session parameters to fail without causing the established session to fail.

The use of random numbers for SSRC provides a minimal amount of security against "media spamming" where a literally *uninvited* third party tries to break into a media session by sending RTP packets during an established call. Unless the third party can guess the SSRC of the intended sender, the receiver will detect a change in SSRC number and either ignore the packets or inform the user that something is going on. This behavior for RTP clients, however, is not

universally accepted, because in some scenarios (wireless hand-off, announcement server, call center, and so forth) it might be desirable to send media from multiple sources during the progress of a call.

RTP supports encryption of the media through the secure RTP (SRTP) profile discussed in Chapter 14. RTP supports a number of different topologies [3] including unicast (point-to-point) and multicast (point-to-multipoint). They are summarized in Table 12.1. In RTP, a translator is an element that converts the codec or sampling rate of an RTP stream. An RTP mixer is an element that combines multiple RTP streams into a single RTP stream in a media specific way.

At the start of an RTP session, the sender randomly chooses an initial value of the timestamp and SSRC. If both the sender and receiver happen to choose the same SSRC, both sides choose again to ensure each have a different SSRC. Media samples are encoded by the codec. Based on the packetization interval, once a complete frame of media data is available, the RTP header is populated and the packet sent. The sampling instant is used to update the timestamp. The sequence number is updated for each RTP packet sent. The receiver first validates the RTP header, using the sequence number to determine if any packets have been lost or received out of sequence. The timestamp is used to playout the media sample by the codec.

12.2 RTP Control Protocol (RTCP)

The RTP Control Protocol (RTCP) is a related protocol also defined in RFC 3550 that allows participants in an RTP session to send each other quality reports and statistics, and exchange some basic identity information. The five types of RTCP packets are shown in Table 12.2. RTCP has been designed to scale for very large conferences. Because RTCP traffic is all overhead, the bandwidth allocated to these messages remains fixed regardless of the number of participants. That is, the more participants in a conference, the less frequently RTCP packets

Table 12.1
RTP Topologies

Point to point
Point to multipoint using multicast
Point to multipoint using an RTP translator
Point to multipoint using an RTP mixer
Point to multipoint using video switching MCUs
Point to multipoint using RTCP-terminating MCU
Nonsymmetric mixer/translators
Combined topologies

Table 12.2
RTCP Reports

Sender report (**SR**)
Receiver report (**RR**)
Source description (**SDES**)
Bye (**BYE**)
Application specific (**APP**)

are sent. For example, in a basic two-participant audio RTP session, the RTP/AVP profile states that RTCP packets are to be sent about every 5 seconds; for four participants, RTCP packets can be sent every 10 seconds. Sender reports (SR) or receiver reports (RR) packets are sent the most frequently, with the other packet types being sent less frequently. The use of reports allows feedback on the quality of the connection including information such as:

- Number of packets sent and received;
- Number of packets lost;
- Packet jitter.

By default, RTCP uses the next highest port from the RTP port, although this can be changed in the offer/answer exchange as discussed in Chapter 13.

12.2.1 RTCP Reports

RTCP is always sent as a compound packet. This means that every RTCP packet starts with a sender report (**SR**) or receiver report (**RR**), then any additional packets. As their name suggests, sender reports are sent by media senders while receiver reports are sent by media receivers. Since RTP is unidirectional, a bidirection media session will have two RTP sessions and two RTCP sessions. A source description (**SDES**) packet is used to exchange information about the sender or receiver. A bye (**BYE**) packet is used to leave a multicast session. An application specific (**APP**) packet is used for RTCP extensions. An important RTCP extension is described in the next section.

12.2.2 RTCP Extended Reports

RTCP extended reports (RTCP-XR) [4] defines seven additional report blocks. They were defined due to limitation of the basic SR and RR. For example, the receiver report contains information about the average packet loss rate. However, for call quality, information about burst packet loss is much more important than average packet loss, since a good codec can cope with individual lost packets but not a long sequence of lost packets. In addition, RTCP-XR defines a way to

estimate actual voice call quality and exchange this information. Deriving this information from existing receiver reports is not possible. As a result, the definition of RTCP extended reports has driven additional implementation of RTCP.

12.3 Compression

RTP does not provide very efficient transport of media. For example, consider the iLBC codec used in the 12.2 kb/s mode with 20 ms packetization time (ptime) transported over RTP, UDP, IPv4, and Ethernet. The size in octets of each frame of codec data can be calculated using the formula:

$$frame = \frac{bw * ptime}{8}$$

where *frame* is the frame size in octets (8 bits, or a byte), *bw* is the codec bandwidth, and *ptime* is the packetization time. For this example, each frame would contain 38 octets of codec data. RTP has a 12 octet header, UDP adds a 16 octet header, while IPv4 with no options adds 20 octets. Ethernet (IEEE 802.3) adds a 13-octet header and a 3-octet footer. As a result, the header overhead for this example is 60 octets! Overhead makes up over 60% of the total packet size. For normal Internet communications, this is how RTP is utilized. However, for some applications where RTP is to be used over low bit rate or wireless links, compression is performed. Note that saving bandwidth is only one reason to do compression. Compression can also reduce serialization delays when sending packets over very low speed links.

One method of compression is compressed RTP (CRTP) [5]. This method only compresses the RTP header fields, using the fact that many parts of the header are identical in every packet. For example, V, P, X, CC, PT, and SSRC typically do not change once a session has been setup, so they do not need to be sent every packet. Other parts of the header such as sequence numbers and timestamps can be sent as deltas, resulting in saved bandwidth.

Another method that compresses the entire RTP/UDP/IP stack is robust header compression (ROHC) [6]. ROHC can compress 40 octets of overhead into 2 octets. To do this, codebooks are used to encode and decode common elements. Codebooks can be either static—predefined for a given protocol—or dynamic—constructed and used during a given session. ROHC can also be used to compress SIP and the stack below SIP.

Although UDP is normally used, it is possible to transport RTP over a stream transport such as TCP. To do this, a framing method is used, which is defined in [7]. If TCP is used, the retransmissions of TCP must be carefully managed or the latency of the session will increase with every retransmission, resulting in very poor performance. Also, when using TCP for media, the roles

of each endpoint must be negotiated. One endpoint will be active, and initiate opening the TCP connection, while the other endpoint will be passive, listening on a port for an open request.

12.4 RTP Audio Video Profiles

The use of profiles enables RTP to be an extremely general media transport protocol. The current audio video profiles defined by RFC 3551 [8] and others are listed in Table 12.3. Four are defined, although only the first one is widely implemented. As secure Internet communications are deployed, the use of the secure audio and video profile (SAVP) is increasing, as described in Chapter 14. The most common profile is the RTP profile for audio and video conferences with minimal control, also known as the RTP/AVP profile. RTP/AVP makes the following specifications for RTP:

- UDP is used for underlying transport.
- RTP port numbers are always even—the corresponding RTCP port number is the next highest port, which is always an odd number.
- No header extensions are used.

Some common audio and video codecs are listed in Tables 12.4 and 12.5. The codecs listed with a payload number in the tables use a static payload number. The RTP/AVP profile document lists details of these codecs, or a reference for the details is provided. Codecs shown with a payload number of dynamic must use a dynamic payload in the range 96–127. Dynamic payloads must be defined dynamically during a session. The minimum payload support is defined as 0 (PCMU) and 5 (DVI4) (although in practice, most only support PCM). The document recommends dynamically assigned UDP port numbers, although ports 5004 and 5005 have been registered for use of the profile and can be used instead. The standard also describes the process of registering new payload types with IANA.

Table 12.3
Defined RTP Profiles

Profile	Name	Specification
RTP profile for audio and video Conferences with minimal control	RTP/AVP	RFC 3551
The secure real-time transport protocol	RTP/SAVP	RFC 3711
RTP audio-visual profile with feedback	RTP/AVPF	RFC 4585
Extended secure RTP profile for RTCP-based feedback	RTP/SAVPF	RFC 5124

Table 12.4
Common RTP/AVP Audio Payload Types

Payload	Codec	Bit Rate
0	PCMU	64 kb/s
3	GSM	13 kb/s
4	G.723	5.3 or 6.3 kb/s
5	DVI4	32 kb/s
8	PCMA	64 kb/s
9	G.722	128 kb/s
18	G.729	8 kb/s
Dynamic	iLBC	13.33 or 15.2 kb/s
Dynamic	AMR	1.8–12.2 kb/s
Dynamic	AMR-WB	6.6–23.85 kb/s
Dynamic	SPEEX	2–44 kb/s
Dynamic	MP3	8–320 kb/s

Table 12.5
Common RTP/AVP Video Payload Types

Payload	Codec	Type
26	JPEG	JPEG video
31	H261	H.261
32	MPV	MPEG-I and MPEG-II
34	H263	H.263
Dynamic	H264	MPEG-4

12.4.1 Audio Codecs

There are two main types of audio codecs—PSTN codecs and Internet codecs. PSTN codecs were developed for the circuit-switched world of the PSTN. They have been designed to minimize bandwidth but were not designed to function over a packet switched network such as the Internet. In particular, their quality rapidly degrades under conditions of packet loss, delay variation (jitter), and other common Internet impairments. Typically, these codecs are only usable when packet loss is less than 1%. Some examples include G.711 (PCM), G.721, G.723, and G.729A. G.711 is also known as pulse coded modulation (PCM) which has two variants, μ-law companding used mainly in the United States and Japan or A-law companding used in the rest of the world. G.711 is uncompressed, with 8 bits samples at 8,000 samples per second resulting in a 64 kb/s data stream. The others implement compression or linear prediction to reduce the bandwidth requirement. However, this compression makes their performance more sensitive to packet loss. For example, a single RTP packet of G.711 lost only affects that sampling interval while a single packet of G.729A can affect

audio quality for a number of sampling intervals. These codecs typically require less than 1% packet loss. PSTN codecs are designed based on 8 kHz sampling due a design limitation of the PSTN network which is not present on the Internet. Many of these codecs also have significant intellectual property (IPR) fees and licensing associated with them. Some modern PSTN codecs overcome some of these problems. For example, the adaptive multirate codec (AMR) [9] was developed by the mobile phone industry with packet transport in mind. As a result, it has reasonable performance under packet loss. There is also a wideband version known as AMR-WB. However, AMR still has significant intellectual property and licensing costs.

In contrast, Internet audio codecs were designed with the Internet in mind. They are designed to give good performance even under conditions of packet loss and delay variation. Also, many provide better-than-PSTN quality by ignoring the 8 kHz sampling limitation. Internet codecs that provide this higher quality are often known as wideband codecs. Examples of Internet codecs include the Internet low bit rate codec (iLBC) [10] and SPEEX [11]. Both of these codecs have no intellectual property or licensing costs, and open source implementations can be found on the Internet. These codecs still have reasonable quality even under conditions of up to 10% packet loss.

12.4.2 Video Codecs

Many of the considerations that apply to audio media transport also apply to video transport. However, there are some key differences. For example, the large amount of information present in every video frame, and the frequency of frame updates means that video requires very high bandwidth. Sending uncompressed video is essentially impractical over the Internet. There are two main techniques in video compression. One is intraframe compression, where information in a single frame is compressed. These frames are called I-frames or key frames. For example, a lossy compression technique such as JPEG could be used. A frame will often be transported in multiple RTP packets. In this case, the marker (M) bit is set on the last packet of a frame to indicate to the codec that the frame is ready for processing and rendering. The other compression technique is interframe compression, where successive frames are compared and differences and predictions made. Predicted frames are known as P-frames and are made relative to a key frame or I-frame. Since often only a small amount of the entire screen changes between each frame, this can result in very compressed, relatively static images. For example, in a telepresence video conference, the I-frame would encode the background image and the face of the person while P-frames could carry their moving lips, blinking eyes, and waving hands. In addition bipredicted frames which are based on multiple P-frames can be used to increase compression. For moving objects in an image, motion vectors of macroblocks can be used

to achieve excellent compression. As a result, a typical video media stream will consist of combinations of I-frames, P-frames, and B-frames.

Since a frame is typically sent over a number of packets, a single lost packet may cause an entire frame to be discarded by the codec. The effect on the quality of the picture depends on the type of frame lost. If it was an I-frame, the loss will have a major impact on quality, and future P-frames and B-frames will result in an incomplete picture until another I-frame is sent. If the lost frame was a P-frame or B-frame, the impact will be less and for a shorter duration. Video codecs employ a number of loss concealment techniques. For example, some use repetition of previous frames, which can work for stationary or slowly moving images. Spatial and frequency interpolation can be used to try to generate lost frames. Also, sending frames using interleaving can provide protection against burst errors. In this approach, parts of different frames are sent out of sequence.

The most common standard video codecs are the H.26x series. H.261 was a very early codec used for video conferencing. H.262 is essentially the same as MPEG-2 which is used in DVDs and HDTV broadcasts. H.263 is commonly used on the Internet today through the Flash Player plugin used by video sharing sites. Many video systems are moving to H.264, which uses MPEG-4 encoding. H.264 is the recommended codec by YouTube [12]. Motion JPEG is a high quality video codec that only uses I-frames with JPEG compression within each frame. It uses much more bandwidth than H.26x codecs but provides a high quality picture even during fast action and motion sequences. In addition, there are many proprietary video codecs in use over the Internet.

12.5 Conferencing

Audio conferencing and videoconferencing are important applications that utilize media transport. Each of these applications has their own media requirements. The details of SIP conferencing are covered in Section 9.7. Audio conferencing requires an audio mixer: a device which combines multiple RTP audio streams into a single stream. A mixer in RTP synchronizes the input media streams then combines them together. The SSRC of each media stream, which was included (mixed) into the resulting stream, will be copied into the contributing SSRC (CSSRC) field of the header. This allows speaker identification during the conference. A typical mixing strategy uses $N - 1$ mixing—that is, the N loudest speakers will have their media combined and shared, but each speaker will not hear themselves—they get the $N - 1$ mix. Thus for $N = 3$, the mixer will produce four distinct mixes, one with all three speakers that is received by nonspeakers, and three with only two of the speakers. Each speaker will get a version of this mix. An audio mixer is sometimes called a multipoint control unit (MCU).

Video mixing can involve combining multiple video streams into a single stream known as tiling (sometimes called "Hollywood squares" if they are presented in a checkerboard arrangement), or by just selecting a video streaming. If video follows audio is used, the video will switch to the loudest speaker. In other cases, users in a videoconference can select which video stream or streams they view, sometimes from a set of thumbnail images. When video switching is occurring, the new video stream needs an I-frame or key frame to be sent immediately, otherwise, the sequence of P-frames and B-frames being sent will not provide a complete image without the I-frame they reference. This is accomplished using fast update signaling between the video mixer and the video source. One method uses an XML message [13] to convey this signaling. Another method uses a special RTCP message [14] and the audio video profile with feedback (AVPF).

12.6 ToIP—Conversational Text

Conversational text, or text over IP (ToIP), is a bidirectional real-time exchange of text characters. Unlike e-mail where the message is only sent when the user hits send, or instant messaging where the message is sent when the user presses enter or return, conversational text messages are sent character by character, usually in full duplex (i.e., both sides can type at the same time). Devices in the PSTN to accomplish this are known as telecommunications devices for the deaf (TDD). Sometimes they are used only for one direction of the call; a human relay operator receives the conversational text messages from one party and speaks the words to the other party. The PSTN uses many different standards and devices for this communication. T.140 [15] is an International Telecommunications Union (ITU) format for encoding conversational text. RTP has a payload for transporting T.140 information [16] over UDP. This payload can use redundant transmission so that individual lost RTP packets will not result in dropped characters. For conversational text to be truly conversational, the end-to-end latency must be less than 300 ms. An industry group known as the Real-Time Text Taskforce (R3TF) [17] has been formed to help the adoption of this technology to the Internet.

12.7 DTMF Transport

Dual tone multifrequency (DTMF) tones are commonly used on the PSTN for dialing telephone numbers. Although Internet communications do not utilize dialing, DTMF still must be transported and supported for user signaling—for example, when entering a personal identification number (PIN) or password to access voicemail or interactive voice response (IVR) systems. Calling card, telephone banking, and many other systems use DTMF tones for signaling. DTMF,

as the name suggests, generates two superimposed sine waves at particular frequencies to send a particular digit (0–9, *, #, or, less commonly, A–F). In the PSTN, DTMF is typically encoded the same way as voice. However, low bit rate codecs, which are optimized for encoding voice, often do not reliably encode DTMF tones. As a result, there is a need to transport DTMF not as sine waves but as actual digits. This is especially appealing for devices such as SIP phones and mobile phones which only need to generate DTMF. A payload known as telephone-events [18] has been defined for transport over RTP. This approach is commonly known in the industry as RFC 2833 tones, where RFC 2833 [19] was the original RFC specification for telephone-events.

The payload contains:

- Event: an octet used to encode the event such as the DTMF key pressed;
- End (E) bit: a bit used to indicate the end of the event;
- Reserved (R) bit: a bit reserved for future use, set to zero and ignored;
- Volume: 6 bits for the level of the tone in dBm0;
- Duration: 16 bits used for a timestamp for the event duration.

When a user presses a DTMF key, or a gateway detects a DTMF tone in band, an RTP telephone-events packet is created and sent. The marker (M) bit in the RTP header is set to indicate that this is the first packet sent. If the key is still being pressed or detected, the duration field will not be valid but should be set to a value higher than the update time. Update RTP telephone-events are sent typically every 50 ms. The RTP timestamp for these update packets will be the same as the first RTP packet but the duration will increase for each. When the key is released or the DTMF tone is no longer detected, a final RTP telephone-event packet is created. The end (E) bit will be set and the duration field will contain the actual tone duration. This final RTP packet will be resent two more times for redundancy. If the DTMF keypress or tone duration is less than the update time, only three RTP telephone events will be sent. The first will have the M bit set, all will have the E bit sent and the duration field will indicate the duration.

12.8 Questions

Q12.1 List the purpose of packet loss concealment. List some methods for packet loss concealment in audio codecs.

Q12.2 Why does RTP usually use UDP transport?

Q12.3 Explain the purpose of the sequence and SSRC fields in an RTP packet.

Q12.4 Calculate the bandwidth required for the SPEEX codec operating at 7.5 kb/s, 25 ms packetization time, assuming transport over UDP, IPv4 (no extensions), and 100BaseT Ethernet.

Q12.5 Explain the differences between RTCP receiver reports and RTCP extended reports.

Q12.6 Describe the three different types of video frames. Explain the need for fast update in a video conferencing system. Which types of frames does motion JPEG use?

Q12.7 How many telephone events packets will be sent if the DTMF key # is pressed and held for 185 ms? Assume the recommended update interval. How many bits in total will be sent, assuming transport over UDP, IPv4 (no extensions), and 1000BaseT Ethernet?

Q12.8 Explain the difference between instant messaging and conversational text.

Q12.9 Describe common audio and video mixing techniques.

Q12.10 Explain the need for the payload type field in the RTP header. Use an example with the iLBC codec and telephone-events to make your point.

References

[1] Schulzrinne, H., et al., "RTP: A Transport Protocol for Real-Time Applications," STD 64, RFC 3550, July 2003.

[2] Cohen, D., "Specifications for the Network Voice Protocol (NVP)," RFC 741, November 1976.

[3] Westerlund, M., and S. Wenger, "RTP Topologies," RFC 5117, January 2008.

[4] Friedman, T., R. Caceres, and A. Clark, "RTP Control Protocol Extended Reports (RTCP XR)," RFC 3611, November 2003.

[5] Casner, S., and V. Jacobson, "Compressing IP/UDP/RTP Headers for Low-Speed Serial Links," RFC 2508, February 1999.

[6] Bormann, C., et al., "Robust Header Compression (ROHC): Framework and Four Profiles: RTP, UDP, ESP, and Uncompressed," RFC 3095, July 2001.

[7] Lazzaro, J., "Framing Real-Time Transport Protocol (RTP) and RTP Control Protocol (RTCP) Packets over Connection-Oriented Transport," RFC 4571, July 2006.

[8] Schulzrinne, H., and S. Casner, "RTP Profile for Audio and Video Conferences with Minimal Control," STD 65, RFC 3551, July 2003.

[9] Sjoberg, J., et al., "Real-Time Transport Protocol (RTP) Payload Format and File Storage Format for the Adaptive Multi-Rate (AMR) and Adaptive Multi-Rate Wideband (AMR-WB) Audio Codecs," RFC 3267, June 2002.

[10] Andersen, S., et al., "Internet Low Bit Rate Codec (iLBC)," RFC 3951, December 2004.

[11] Herlein, G., et al., "RTP Payload Format for the Speex Codec," draft-ietf-avt-rtp-speex-05 (work in progress), February 2008.

[12] YouTube, http://www.youtube.com.

[13] Levin, O., R. Even, and P. Hagendorf, "XML Schema for Media Control," RFC 5168, March 2008.

[14] Wenger, S., et al., "Codec Control Messages in the RTP Audio-Visual Profile with Feedback (AVPF)," RFC 5104, February 2008.

[15] ITU-T Recommendation T.140 (1998)—Text Conversation Protocol for Multimedia Application, with Amendment 1, (2000).

[16] Hellstrom, G., and P. Jones, "RTP Payload for Text Conversation," RFC 4103, June 2005.

[17] http://www.realtimetext.org/.

[18] Schulzrinne, H., and T. Taylor, "RTP Payload for DTMF Digits, Telephony Tones, and Telephony Signals," RFC 4733, December 2006.

[19] Schulzrinne, H., and S. Petrack, "RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals," RFC 2833, May 2000.