Advanced Networking

Voice over IP:
RTP/RTCP – The transport layer

Renato Lo Cigno
Renato.LoCigno@disi.unitn.it
Requirements For Real-Time Transmission

• Need to emulate conventional telephone system
  - *Isochronous* - output timing same with input timing
• IP Internet is not isochronous!
• Additional protocol support is required:
  - *sequence information* that allows detection of duplicate or reordered packets
  - each packet (indeed sub-part of it) must carry a separate *timestamp* that tells the receiver the exact time at which the data in the packet should be played
• Playback
  - Internet introduces burstiness
  - *Jitter buffer* used to smooth bursts
  - Protocol support needed
Illustration Of Jitter Buffer

- Data arrives in bursts
- Receiver *delays* playback until certain threshold, \( k \) – playback point
  - \( k \) too small, still have jitter and freeze of info flow
  - \( k \) too large, extra delay noticeable to users
- Data leaves at steady rate
RTP: characteristics and functionalities

- Independent from the PHY (obvious!!!)
- Scalable
  - Unicast e multicast
- Defines separate logical channels for data and control
  - indeed a “pair” of protocols RTP-RTCP
- Packet reordering at destination
- Delay jitter equalization with buffers (in addition to the playout buffer of the application)
- Sender identification
- Intra-media synchronization
- No predefined Port, but must be even
### RTP: header format

<table>
<thead>
<tr>
<th>Ver.</th>
<th>P</th>
<th>X</th>
<th>CSRC ct.</th>
<th>M</th>
<th>payload type</th>
<th>sequence number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- **Timestamp**
- synchronization source identifier (SSRC)
- content source identifiers (CSRC)
- Extension Headers (optional)

**Data**
The RTP header (12 bytes)

- **Ver. (2 bits):** Version of the protocol. Current is 2
- **P (1 bit):** Indicate if there are extra padding bytes at the end of the RTP packet.
- **X (1 bit):** Extensions to the protocol used (ELH present)
- **CC (4 bits):** Number of CSRC identifiers that follow the fixed header
- **M (1 bit):** If set means that the current data has some special relevance for the application defined in a profile (external to the protocol)
- **PT (7 bits):** Format of the payload and its interpretation by the application
- **SSRC:** Indicates the synchronization source and timing
- **Extension header:** Length of the extension (EHL=extension header length) in 32bit units, excluding the 32bits of the extension header
RTCP

- **Real Time Control Protocol**
- **Functionalities:**
  - Data Distribution Control
  - Session information advertisement (during the session, not for setup)
  - Distribute information about participants
    - Who is participating in session
    - Who is speaking now
  - QoS feedback
  - Error reporting
  - ...

- **RTCP messages are sent on RTP-port+1**
RTP Application Programming

Application Specific

OS API

IP
Integrated Layer Processing
RTP sending process

• Sending a packet, RTP adds:
  - A timestamp per Application Data Unit (not per packet)
  - A Synchronisation Source Identifier (SSRC)
    • Randomly generated
    • Conflicts resolved
    • Uniquely identifies flows to receivers
Why is a flow identifier needed?

Multiple flows may come from the same process (audio and video), the receiver has to separate them (same IP and port)
Translation & Mixing

• Supports changing stream encoding during a session - translation
• RTP can coordinate multiple data streams
• Up to 15 sources (c.f. 4-bit CC field)
• Header specifies mixing
  - Contributing source id - SSRC of streams that were mixed
• Performed through relays, which also provides for multicast
RTP Relays

• relay (or application level routing)
  - operate on intermediary system
  - acts as both destination and source
  - relay data between systems

• mixer
  - combines streams from multiple sources
  - forwards new stream to one or more dests (multicast)
  - may change data format if needed

• translator
  - simpler, sends 1+ RTP packets for each one received
RTP Relays: Further functions

- **Mixer**
  - New Synchronisation Source Identifier
  - Add old SSRC as Contributing Source Identifiers

- **Translator**, manipulates the content, e.g. a transcoder
RTCP Packet Types

- have multiple RTCP packets in datagram
- Sender Report (SR)
- Receiver Report (RR)
- Source Description (SDES)
- Goodbye (BYE)
- Application Specific
RTP Packets

(a) RTCP Sender Report
- VH RC PT = 200 Length
- SSRC of sender
- NTP timestamp (most sig. word)
- NTP timestamp (least sig. word)
- RTP timestamp
- Sender's packet count
- Sender's octet count
- SSRC_1 (SSRC of first source)
- Fraction lost
- Cumulative number of packets lost
- Extended highest sequence number received
- Interarrival jitter
- Time of last sender report
- Delay since last sender report

(b) RTCP Receiver Report
- VH RC PT = 201 Length
- SSRC of sender
- SSRC_1 (SSRC of first source)
- Fraction lost
- Cumulative number of packets lost
- Interarrival jitter
- Time of last sender report
- Delay since last sender report

(c) RTCP Application-defined packet
- VH sub-type PT = 204 Length
- SSRC/CSRC name (ASCII)

(d) RTCP Source Description
- VH SC PT = 202 Length
- SSRC/CSRC_1
- SDES items

(c) RTCP Application-defined packet
- VH SC PT = 203 Length
- SSRC/CSRC_1
- SDES items

(c) RTCP Application-defined packet
- VH SC PT = 204 Length
- SSRC/CSRC
- Reason for leaving
RTCP: generation of reports

• Frequency inv. Proportional to the number of members
  - Excellent scalability (BW limited)
  - Initial “storm”
Report calculations

- Estimate total bandwidth of session
- Sender period T:
- Receiver period T:
- Next packet = last packet + Max(5s, T) + random (0.5-1.5)
- Random prevents synchronization and storms

\[
T = \frac{\#\text{Participants}}{0.25 \times 0.05 \times \text{SessionBW}} \times \text{avg.}_\text{RTCP\_packet\_size}
\]
RTCP Packets

Receiver report packets:
• fraction of packets lost, last sequence number, average interarrival jitter

Sender report packets:
• SSRC of the RTP stream, the current time, the number of packets sent, and the number of bytes sent
RTCP Bandwidth Scaling

- RTCP attempts to limit its traffic to 5% of the session bandwidth.

**Example**
- Suppose one sender, sending video at a rate of 2 Mbps. Then RTCP attempts to limit its traffic to 100 Kbps.
- RTCP gives 75% of this rate to the receivers; remaining 25% to the sender

- The 75 Kbps is equally shared among receivers:
  - With R receivers, each receiver gets to send RTCP traffic at 75/R Kbps
- Sender gets to send RTCP traffic at 25 Kbps
- Participant determines RTCP packet transmission period by calculating avg RTCP packet size (across the entire session) and dividing by allocated rate