MPRTTP

draft-singh-avt-mprtp-00.txt

Varun Singh
Teemu Kärkkäinen
Jörg Ott
Saba Ahsan
Simple Basic Idea

- Exploit multiple (at least partly disjoint) paths between two unicast RTP endpoints

- Similar to the idea of the Multipath-TCP WG
Where Do Paths Come From?

- **Multiple host interfaces**
  - Learned from the OS
  - Learned from ICE
  - Probed in another way

- **Overlay paths**
  - STUN servers (learned from ICE)
  - Other transport overlays

- **Network paths**
  - Learned by some routing magic
Goals: What to use them for?

• Aggregate data rate
  – Higher or more stable

• Load balancing

• Fault tolerance

• Interface optimization
  – E.g., power or performance
Goals: Compatibility

• Keep the “application interface” of RTP
  – I.e., do not require applications to do anything else then create/accept RTP/RTCP

• Keep the RTP/RTCP flow properties
  – I.e., do not confuse middleboxes, monitors, ... that see packets on a single path only
# Simple MPRTP Architecture

<table>
<thead>
<tr>
<th>Application</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>MPRTP</td>
<td>MPRTP</td>
</tr>
<tr>
<td>MPRTP Subflow</td>
<td>MPRTP Subflow</td>
</tr>
<tr>
<td>UDP</td>
<td>UDP</td>
</tr>
<tr>
<td>IP</td>
<td>IP</td>
</tr>
</tbody>
</table>
Non-Goal

• Solve the rate adaptation problem for RTP
Functional Block of MPRTP

- Session Setup
- Expanding RTP
- Adding Interfaces
- Expanding RTCP
- Maintenance and Failure Handling
- Teardown
Path Discovery (e.g., ICE)

MPRTP Capability Discovery (in-band with RTP)

- Assume startup signaling for multipath discovery (e.g., ICE)
- Initial session backwards compatible with RTP
  - MPRTP capability discovery in-band
- Subsessions added
  - Discover MPRTP support
  - Learn about new paths
A Prospective MPRTP Spec

• Interaction with session setup management
  – SDP (caps) and SIP, RTSP, etc.
  – ICE (interfaces)
  – In-band mechanisms

• RTP/RTCP extensions for subflows
  – Monitoring for individual paths
  – Packet allocation to paths and scheduling (sender)
  – Aggregation of subflows (receiver)

• Mechanisms: leave the algorithms open
Questions

• Many open ones... 😊

• Does it make sense to think about this in AVT?

• What other use cases come to mind?
• Which further constraints to worry about?
• Further architectural implications wrt RTP?
  – Mixers/translators...