RTSP: NAT traversal for media streams

draft-ietf-mmusic-rtsp-nat-06

Magnus Westerlund / Ericsson
Jeff Goldberg / Cisco
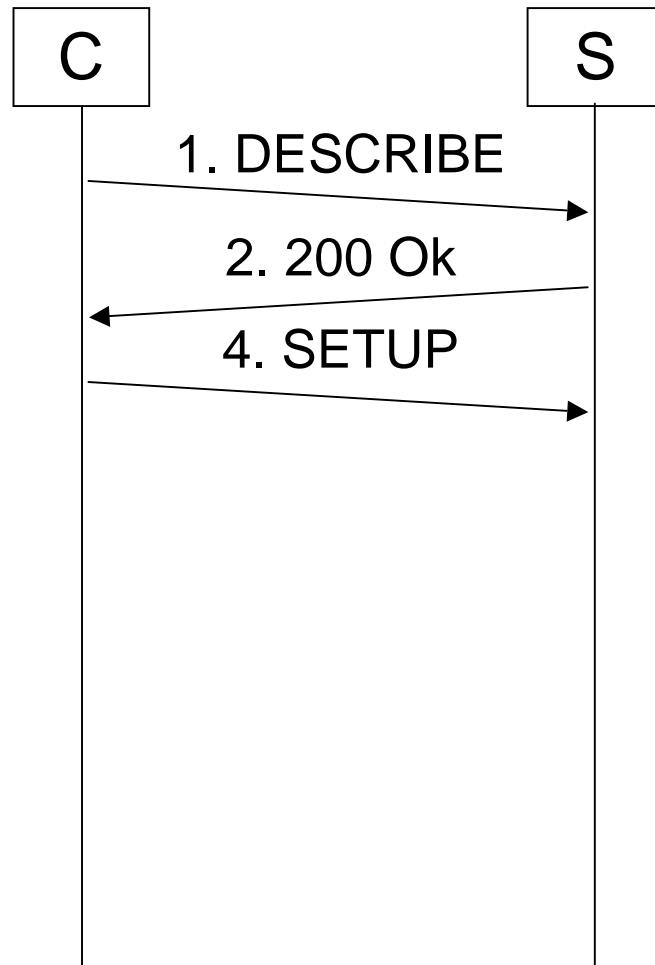Thomas Zeng / Nextwave Wireless

The background

- There is a clear need for a NAT traversal for the media controlled by RTSP

- ICE has previously been chosen because:
  - Prevents server being used as denial of service tools
  - Reuses well fleshed out solution

- Discussion around the possible choices are documented in draft-ietf-mmusic-rtsp-nat-evaluation

Fleshed out proposal

- ICE becomes a lower layer for the higher layer transport
  - This document defines one that supports datagrams (D-ICE)
  - RTP/AVP/D-ICE will now be the transport protocol specification in the Transport header.
  - Candidates are listed in a transport parameter header
- Optimization for servers with public addresses to reduce processing load
  - Normally single candidate per address family
  - Do triggered checks
- Proposal is in complexity between Full ICE and NICE

Example

```
C                          S
│                          │
│      1. DESCRIBE         │
│ ───────────────────────► │
│                          │
│      2. 200 Ok           │
│ ◄─────────────────────── │
│      4. SETUP            │
│ ───────────────────────► │
│                          │
│                          │
```

- Describe request including "Supported: setup.ice-d-m"
- Describe response: "Supported: setup.ice-d-m" and SDP also contains "a=rtsp-ice-d-m"
- Client gather candidates
- Client requests to setup a media stream:
  Transport: RTP/AVP/D-ICE; unicast; candidates = "
  1 1 UDP 2130706431 10.0.1.1 8998 typ host;
  2 1 UDP 1694498815 192.0.2.3 45664 typ srflx raddr
  10.0.1.1 rport 9002",
  RTP/AVP/UDP; unicast; dest_addr=":6970"/":6971",
  RTP/AVP/TCP;unicast;interleaved=0-1

Example
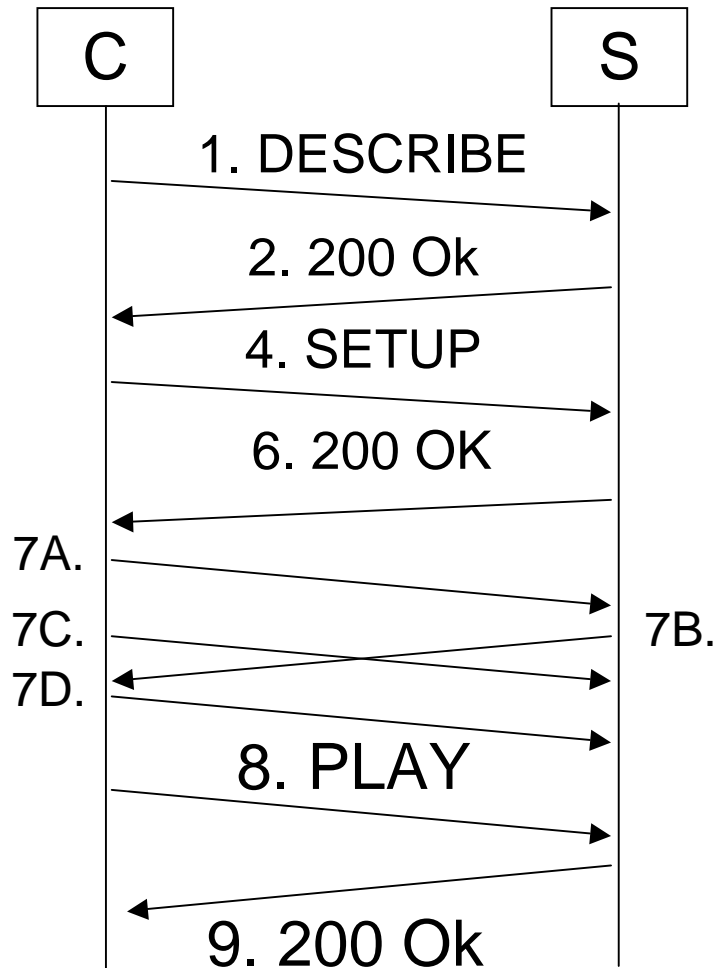
- Server Gather Candidates
- SETUP Response:
  Transport: RTP/AVP/D-ICE; unicast; candidates = " 1 1 UDP 2130706431 192.0.2.56 50234 typ host"
- Connectivity Checks (aggresive nomination):
  A. Client high priority pair
  B. Triggered Response
  C. Client next priority pair
  D. STUN response
- Client has completed Checks and issues PLAY request
- If Server has completed its check its sends 200 OK, else 150 (ICE connectivity checks in progress). If server checks fails (they shouldn't) it responds with 480 (ICE Processing Failed)

Further Properties

- Re-Setup creates a completely new ICE transaction:
  - Requires new username password pairs for the new connectivity checks
  - Simplifies the handling of corner cases

- Re-setup also uses regular nomination to avoid inconsistencies when switching from any established candiate pair to the promoted one.
  - Could be changed if requiring this to always happen in ready state.

Open Issues

- Is the approach taken a good one?

- We recommend that one uses RTP and RTCP Mux but current version does support multiple components
  - Is the support for multiple components needed?
  - In other words can we guarantee that no media protocol ever will need multiple streams per media?
  - Advantage would be to simplify processing of checkboards as each candidate pair only have a single component

- A few editorials:
  - General ICE description is missing

More Open Issues

- Server initiated ICE restart
  - To make RTSP ICE robust in failure cases Server maybe needs to be able to initiate ICE restart?
  - If so we need to decide method for doing that:
    - Server intitiated SETUP?
    - Asynchronous notification to client?

- ICE TCP
  - Seems quite straightforward to allow also TCP type of candidates for D-ICE
  - For Stream based media transports a new transport lower layer "S-ICE" would be analogous to D-ICE

NAT Traversal for the signalling connection

- Discussion about the lack of this functionality

- Not intended in this draft

- But will be needed for servers behind NATs

- Probably need some type of Proxy/Rendevouz service

- Suggest interested go ahead and make individual draft

RTSP Core

draft-ietf-mmusic-rfc2326bis-17

Magnus Westerlund, Martin Stiemeriling, Henning Schulzrinne, Rob Lanphier, Anup Rao

Status Report

- Primarily editorial changes in this version:
  - Some restructuring
- Still need to implement a number of consensus from last IETF's WG and off-line session.
- Some open questions on what to include in the Core:
  - Strong suggestion to include END_OF_STREAM semantics in some way
- End of Session:
  - Discussion seems to be needed over what semantics are needed.
  - Bring out and clarify that functionality.