RTSP 2.0

draft-ietf-mmusic-rfc2326bis-12
draft-ietf-mmusic-rtsp-nat-04

ERICSSON
TAKING YOU FORWARD

Open Issues in Base Spec

- Definition of non-interleaved TCP/RTP/AVP?
- Inclusion of SMPTE 50 and 60 fps formats
- Format of Error Message bodies
- Format for URI list in 300 (Multiple Choices)
- Should dest_addr contain used address?
- Should there be a scope address for IPv6 multicast?
- Usage of unregistered Media Types in examples
- Expires also provide cach-control instructions
- Proxy handling of Accept-Credentials

1. Definition of non-interleaved RTP/AVP/TCP?

- Needs definition if anyone wants to use a separate TCP connection for media transport
- Needs text similar to B.1.2 for RTP/AVP/UDP
- If none is voluteering to draft intial text this will be dropped.

2. SMPTE 50 and 60 FPS

- To my knowledge SMPTE has defined 50 and 60 frame per second formats for their timestamps.
- If those format is desired to be used in RTSP range headers they need to be defined.
- Unless someone drafts text they will be excluded and for further extension work.

3. Format of Error Message bodies

- RTSP 2.0 allows the inclusion of a message body in all error responses (4xx or 5xx).
- The format of that message body is not defined.
- If it is desirable to have a standardized way of providing the requesting party with a human readable error message response then we should define a format.
- A proposal would be to use either HTML or UTF-8 encoded text.

4. Format for URI list in 300 Response (Multiple Choices)

- The 300 response code indicates that their are multiple choices for the resourece. The user or user-agent needs to select the most suitable resource location.
- If it is desired to have a interoperable functionality for letting the user agent select from multiple choices some kind of format would be needed.
- Alternative 1: Define a format for the URI-List
- Alternative 2: remove 300 as supported response code
- Alternative 3: Keep 300 but not specify a format, thus creating an interoperability issue

5. Content of dest_addr in SETUP responses

- The transport header parameter dest_addr may include only ports in requests.
- Should it be mandatory for server to include the used IP address in SETUP responses?
- Would be inline with the notion of keeping thing explicit for Firewalls and proxies.
- Does also provide a way for RTSP agents to verify that the intended operation has happened before sending PLAY.

6. Scope for IPv6 Multicast Addresses

- For IPv4 multicast addresses there is the TTL transport header parameter.
- For IPv6 the scope is part of the address itself, thus no need for a parameter.

7. Usage of unregistered Media Types in examples

- In the current version there is a few cases where there is usage of ”application/rtsl”. This is not a registered type.
- Because of that I would like to avoid using it as it could cause issues in the future if it would be registered.
- But at the same time, multiple media types would be beneficial in the examples.
- Tom Taylor suggested to use application/example.
- Discussion has resulted in that we will look into registering some type of example version of a media type.

8. Expires and Cache-Controll

- The Expires header definition has the following text:
  - Expires header field with a date value of some time in the future on a media stream that otherwise would by default be non-cacheable indicates that the media stream is cacheable, unless indicated otherwise by a Cache-Control header field (Section 14.10).
- Missaligned clocks shouldn't be a major issue as long as the "Date" header is used in the response.
- Is overloading the expiry time with also having cache-control meaning good?
- I guess this is due to simplified default behavior in caches.
- Should we simply leave it as it is?

9. Proxies and the Accept-Credentials header

- The Accept-Credentials header is forward with the request.
- Each entry within the Accept-Credentials headers has a intended proxy.
- Should that proxy remove the entry intened for itself before forwarding the request?
- Doing the above procedure rather then having them go end to end would:
  - Reduce bandwidth in requests
  - Slightly increase processing load
  - Hide earlier TLS hops from later RTSP agents in this header
  - Via shows route, however it allows for a proxy to hide topology
- What are the security implications?

Way Forward with RTSP 2.0

- Hope to resolve these issues quickly.
- Has requested further security review of TLS solution in the TLS WG.
- We are getting close to WGLC – Finally!
- Needs review and help to resolve issues.
- Draft text is very much appreciated.

RTSP 2.0 and NAT traversal

- There has been no update of the draft since last meeting.
- ICE seems to be possible to MAP on RTSP in a nice way.
- Needs to develop the actual solution description.
- Need reinforcements in the author team to speed up progress.