CLUE Signaling
(draft-kyzivat-clue-signaling-04)

July 29-Aug 1, 2012

Editor: Paul Kyzivat

Major Open Issues

- Relationship between CLUE messages & SDP
- Representation of encodings: SDP or ADV
- Approach to message responses (ack/nak/error)
- Elaboration of message sequencing
- Legacy mode

- CLUE channel
- Message Syntax Details
- Approach to versioning/options/extensions
- Approach to message encoding: stand-alone or deltas

- Examples/call flows

Relationship between CLUE messages & SDP

- Discussed in Section 5
  - Largely Rob's work so far
- Approach: independent negotiation of each
  - Media to be consistent with both
- Is more complete than most other sections
- IMO is "good enough" for now

Representation of Encodings: SDP or ADV

- Rob has proposed putting encoding representation in SDP, not in the ADV
  - This has not been agreed
- Current doc assumes this approach
  - Just a trial, subject to change
- Needs careful evaluation:
  - It seems to require a lot of O/As
  - Moving encodings to ADV might save O/As
  - Maybe we need to try both ways and compare
- Choice impacts details of msg encoding & data model

Approach to message responses (ack/nak/error)

- There has been very little work or discussion
  - I put a couple of alternatives in the text
  - draft-presta-clue-protocol adds another
  - Work in other sections not necessarily compatible
- Interacts with other sections
  - We need to nail this down so can make progress elsewhere

Message response Alternatives

- ADV & CFG get an explicit response msg.
  - CFG is not the response to ADV.
- Each ADV/CFG implicitly ACKs previously received msgs
  - An explicit NAK msg is used to report an error
  - Work in other sections not necessarily compatible
- presta-clue-protocol:
  - CFG acks ADV
  - RE-ADV nacks ADV
  - RESPONSE acks/naks CFG
- Any others we should be considering?
- We need to pick one
  - And then work it out in detail

Elaboration of message sequencing

- This needs to be worked out at multiple levels
  - – Sequencing of CLUE messages
    - State machine for provider
    - State machine for consumer
    - Coupling between ***peer*** provider & consumer
    - Coupling between ***collocated*** provider & consumer
  - – Sequencing/dependency between CLUE messages and SDP offers/answers
  - – Transition between CLUE and legacy modes
- This needs a lot of work!
  - – Nothing useful in doc, a start in presta doc

Legacy Mode

- Tradeoff:
  - maximize possibility of success with legacy peer
  - minimize call setup with clue peer
- Priority: make key decisions that impact clue mode:
  - Offer clue channel before clue support known?
  - May clue-controlled media be offered before choice of legacy/clue mode is decided?
  - Can the legacy mode 5-tuples and RTP media streams be reused in clue mode?
- Other issues:
  - May/should legacy streams be dropped once clue in use?
  - Is it possible to revert to legacy mode from clue mode?

CLUE channel

- Current proposal is to use:
  - SCTP over DTLS over UDP
  - draft-ietf-mmusic-sctp-sdp-nn for setup
  - Single SCTP stream in each direction (equivalent to an RTCWEB Data Channel)
  - Single SCTP message per CLUE message
  - No additional message framing other than what is included in message syntax
- Is there any desire to revisit this?

CLUE channel lifetime & error handling

- (Re)Establishment of SCTP association signaled by SDP per RFC 4145 (comedia)
  - O/A to request a new association
- What state is tied to CLUE channel?
- How does CLUE channel state relate to SCTP association state?
- What happens when an error is detected on CLUE channel of SCTP association?
  - Do most recent ADV & CFG remain in effect?
  - Attempt to reestablish SCTP?
- May CLUE channel be intentionally dropped if further use is not expected?
  - Should other side then attempt to reestablish if it wants to send ADV or CFG?
  - How to prevent continued attempts at reestablishment?

Message Syntax Details

- Current proposal is to use XML:
  - Basic message syntax defined in this draft
  - Significant content by reference to data model syntax
  - Any objection to this approach?
- Current details are just a straw horse
  - Just a way to try out the approach
  - No attempt to be consistent with anything
  - Needs to be done over after other decisions are made

Approach to versioning/options/extensions

- For all of these, do we distinguish between the CLUE channel and overall CLUE behavior?
- Do we handle options/extensions via versioning, or separate from versioning?
  - Do we have any examples of options to work with?
- Do we establish once for a session?
  - Or permit changes during a session?
- Do we negotiate in SIP, SDP, or in the CLUE channel?
- What naming/numbering scheme?
- What requirements for backward compatibility?

Approach to message encoding stand-alone or deltas

- Proposal:
  - Assume stand-alone to start
  - Revisit after we have enough detail to understand how big and frequent the messages are

Examples & call flows

- Examples in sections 6 & 7 aren't based on the rest of the document
- We need different kinds:
  - Sequence diagrams
  - Message content
  - Clue-channel-only, uni-directional
  - Clue-channel-only, bi-directional
  - Clue-channel + SDP O/A
  - SIP + SDP + CLUE-channel
  - Ordering flexibility means we can't show all possibilities
    - Focus on common cases and problem cases
- We need ***some*** examples as we go to evaluate the decisions being made

END