Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: April 26, 2012

S. Wenger
Vidyo
M. Eubanks
AmericaFree.TV
R. Even
Huawei
G. Camarillo
Ericsson

October 24, 2011

Transport Options for Clue
draft-wenger-clue-transport-01

Abstract

This memo describes the assumption and the proposed options for the coding and transport of CLUE messages as outlined in version 01 of the framework draft.

Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [RFC2119].

Status of this Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on April 26, 2012.

Copyright Notice

Table of Contents

1. Introduction

The CLUE WG is chartered to design a protocol working in conjunction with the IETFOs protocol suites of choiceNnamely SIP for basic call setup and control and RTP for media transport. This document describes options for the coding and transport of CLUE messages in a SIP / RTP environment. Specifically, three issues are addressed.

First, while the framework draft conceptually describes message flows, it does not specify how those messages are actually transferred Oon the wireO and how they relate to the SIP offer/answer [rfc3264]. This document lists the options that have been proposed in CLUE to date, plus some new ones derived by the authors.

Second, the framework-01 draft describes three messages between the producer and the consumer in an abstract form, without specifying the details of the representation of those messages. This memo lists (some of) the options for the OcodingO of the abstract messages of the framework draft.

Third, before any CLUE messages can be meaningfully exchanged, it is necessary to discover whether the involved systems are actually CLUE-capable.

In version -00 of this document we deliberately list all options we could come up with, however likely they may be to find consensus in the group. Deciding on the appropriate mechanism (or mechanismsN it is not always appropriate to have a single solution for a given problem, though this is of course desirable from an interoperability viewpoint) is left as an exercise for later. That does not mean that the authors do not have preferences, and/or specific knowledge of certain mechanisms, and would go in greater depth in describing one mechanism while being superficial in describing another.

The message ladder diagram will be added once we decide on the transport of the CLUE messages

2. Assumptions

The Basic Clue data model is specified in the framework document. The framework defines three messages that carry the Clue data:

Provider Capabilities Announcement

Consumer Capability Message

Consumer Configure Request

CLUE messages may need to be sent at the initialization of a call, and possibly in irregular intervals which are spaced apart in the order of seconds, minutes or even longer.

There is no hard real-time transmission requirement for CLUE messages; latencies in the seconds range are acceptable.

The Clue message handshake is different from the offer/answer exchange [rfc3264], primarily because the CLUE exchange is uni-directional, requiring a similar exchange for each side of the media flow, while one offer/answer exchange defines both sides of the media flow.

There is no hard requirement for synchronization of CLUE messages, though there may be a need for sequencing, (TBD).

CLUE messages may need to describe the characteristics of all endpoints in a conference (TBD), and that conference can potentially include dozens of endpoints.

There will be an SDP offer/answer exchange as part of the solution. The offer/answer will be used to establish the media channels and negotiate SDP parameters as well as to allow interoperability with systems that do not support the CLUE protocol. The CLUE data will try to not duplicate SDP attributes.

3. Transport for CLUE messages

CLUE messages need to be conveyed from one CLUE capable system to another. This conveyance is called OtransportO of CLUE messages. It should be clear that the message transport can be based on a transport layer (layer 4 in ISO/OSI) protocol or other layers, such as the application layer.

In contrast to the Ocontent representationO, the transport of CLUE messages is somewhat more tightly bound to the environment. In some scenarios it may be possible to reuse most of the mechanisms defined in an option for transport between SIP and H.323, while in others this is not possible.

The selection of the transport may have some affect on the content representation.(Need to write more about the specifications that need to be written, ie. SIP-INFO package, RTP payload format, etc.)

3.1. Option 1 : Piggy-pack on SIP

SIP includes a number of methods that can carry (directly or through content indirection) CLUE messages. Many of these messages can be exchanged during the lifetime of a session without having adverse side effects such as complete codec initialization (what happens today in many products when using re-invite). Piggy-packing CLUE messages on SIP has the advantage that any built-in transport and reliability mechanisms of SIP can be re-used. It also has the feature (advantage?) that CLUE signaling is being conveyed in the signaling plane rather than in the media plane (making things such as decomposition potentially easier and certainly more intuitive).

One option that was mentioned before was to define a new INFO package [RFC 6086]. When looking at using SIP signaling there are other options like subscribe/ notify or Message method, see RFC 6086 section 8.4.1. Note that subscribe creates a separate dialog usage and is normally sent outside of existing dialog. We can also use the UPDATE method [RFC3311]. There were concerns about using re-invite claiming that it takes too long since that commonly used codec boxes teardown every existing media session during re-invites. RFC 3311 says that although UPDATE can be used on confirmed dialogs, it is RECOMMENDED that a re-INVITE be used instead. This is because an UPDATE needs to be answered immediately, ruling out the possibility of user approval. Such approval will frequently be needed, and is possible with a re-INVITE. The thinking so far was that if we want to encode the CLUE data not as SDP we may be better to use INFO.

3.2. Option 2: CLUE control channel on the media plane over UDP

During the initial SIP handshake, a CLUE channel is established (if both systems are CLUE capable). Over this channel, secure UDP packets are exchanged in a reliable fashion, for example, by a CLUE defined protocol that can have a reliable handshake based on similar mechanisms in BFCP over UDP. Standard ICE can be used to deal with firewalls and NATs. Issues here are congestion control and the architectural issue that signaling information is conveyed in the media plane (which may or may not be anything beyond an aesthetic problem).

3.3. Option 3: CLUE control channel on the media plane over TCP

This option is similar to the use of CLUE on the media plane, but uses TCP as the transport protocol. TCP takes care of reliability issues as well as congestion control. However, the NAT/firewall traversal may be a major issue, as ICE-TCP has not seen any deployment in the video conferencing industry. In addition, keep-alive messages may present a problem for sessions with thousands of

attendees, which is possible under some deployment scenarios.

3.4. Option 4: CLUE control channel over UDP and RTP

This option is similar to option 2, except that the mechanisms of RTP can be used to make the transmission sufficiently reliable (through re-transmission or FEC extensions of RTP).

3.5. Option 5: FTP

CLUE messages could conceivably be placed into files, which could be polled at regular intervals (or through a simple message) using ftp. A bit archaic and has security and NAT/filewall issues. We mention this option for completemess only, and because it was used in at least one legacy telepresence system. In the authorsO opinions, itOs probably not a viable choice.

3.6. Option 6: HTTP

The authors have not studied this option, and suggest to remove it in the -01 option of this document unless people find it viable (and come up with text). HTTP transport over port 80 is increasingly used to get through NAT/firewall blockages, and this mechanism may be required if technologies such as RTCWEB begin to be used in videoconferencing and telepresence. Questions include, dach box has a web server? Would there need to be a central web server for CLUE control at service provider?

4. Content Representation

The data model in the framework-01 draft does not have a specific representation of the data. Many different representation OlanguagesO, for example XML, possibly SDP, ASN.1, and others can be used, and we need to decide on one. The decision may be based on the selected transport, but not necessarily.

One key observation that has to be made at this point (described in greater detail above) is that the framework-01 draftOs message exchange system appears to make it impossible to directly add the CLUE exchange to the offer/answer mechanism SIP videoconferencing endpoints use today. It is, therefore, not a hard requirement to use SDP for the representation of the CLUE messages. We have a freedom of choice here, which is why this section exists.

Another observation is that the IETF is not the only body who standardizes telepresence systems; the ITU-T is also working in this field. While it probably shouldnOt be a hard requirement for an IETF

document to allow for seamless interoperability with the H.323 standards suite, it appears to be desirable if it can be easily done.

It is very well possible that even moderately complex CLUE messages may exceed MTU sizes commonly found in todayOs Internet. There has been discussion in CLUE of sessions with thousands of participants. Even if a CLUE message can be compressed into a few bytes for each endpoint, such sessions can easily violate the commonly found Ethernet 1492-byte MTU Accordingly, message transport protocols will have to be prepared to split CLUE messages into fragments, which has implications on the design complexity of those protocols. This problem is especially an issue for verbose representations, such as XML.

4.1. Option 1 : SDP

SDP and its various extensions are used in SIP based systems for the offer/answer exchange, and, therefore, those systems include SDP parsers that could probably be extended to support CLUE messages. SDP is also a fairly compact, but still (though barely) human readable content representation language. Against SDP speaks mostly that SDP was never designed to describe anything as complex as the CLUE data.

4.2. Option 2 : XML

XML is very flexible, and the representation of choice for many IETF technologies not bound to a certain legacy. It certainly allows for all flexibility needed to represent all CLUE messages currently considered. It also is naturally extensible in a way SDP is not. On the downside, XML is fairly verbose, which has implications on the transport.

4.3. Option 3 : ASN.1

ASN.1 is similarly flexible and extensible as XML, and (in its binary representation) fairly compact. While it is commonly used in H.323, and while the video conferencing industry certainly has access to the tools necessary to deploy ASN.1 (a major obstacle in other industries), it is not widely used by SIP implementations.

4.4. Option 4 : Clue Defined Format

It is, of course, possible that the CLUE WG defines its own format, possibly compact, possibly binary and possibly extensible representation language or format for CLUE messages.

4.5. Examples

An example or examples should be added here when possible

5. Clue Discovery

This section summarizes ways to discover whether systems involved are CLUE-capable. For simplicity, point-to-point scenarios are assumed. Multipoint scenarios can potentially make discovery considerably more complex.

Discovery appears to be necessarily bound to the capability exchange of the involved systems.

5.1. Option 1 : CLUE discovery as a side effect of opening a CLUE control channel

If, for the transport of CLUE messages, a media plane control channel were used (option 2,3,4 of the transport options), then the discovery of CLUE capability would be a side effect of the opening of this control channel during the initial offer/answer exchange.

5.2. Option 2 : SIP Message Transport

If we use the INFO message then by using the Recv-Info header field the support for the CLUE package can be signalled.

6. IANA Considerations

This document makes no request of IANA.

Note to RFC Editor: this section may be removed on publication as an RFC.

7. Security Considerations

Any method for bypassing NAT/Firewall protections of course brings security issues, which need to be dealt with.

8. Acknowledgements

The list of authors needs to grow.

9. Informative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

Authors' Addresses

Dr. Stephan Wenger
Vidyo
433 Hackensack Ave
Hackensack, NJ 07601
USA

Email: stewe@stewe.org

Marshall Eubanks
AmericaFree.TV
P.O. Box 141
Clifton, Virginia 20124
USA

Phone: +1-703-501-4376
Email: marshall.eubanks@gmail.com

Roni Even
Huawei

Email: ron.even.tlv@gmail.com

Gonzalo Camarillo
Ericsson

Email: Gonzalo.Camarillo@ericsson.com