    **Problem Statement and Requirements for Transporting User to User Call
                    Control Information in SIP
                    draft-ietf-cuss-sip-uui-reqs-09**

Abstract

   This document introduces the transport of call control related User
   to User Information (UUI) using the Session Initiation Protocol
   (SIP), and develops several requirements for a new SIP mechanism.
   Some SIP sessions are established by or related to a non-SIP
   application. This application may have information that needs to be
   transported between the SIP User Agents during session establishment.
   In addition to interworking with the ISDN UUI Service, this extension
   will also be used for native SIP endpoints requiring application UUI.

Status of this Memo

Copyright Notice

carefully, as they describe your rights and restrictions with respect
to this document. Code Components extracted from this document must
include Simplified BSD License text as described in Section 4.e of
the Trust Legal Provisions and are provided without warranty as
described in the Simplified BSD License.

Table of Contents

[1](#). **Overview**

   This document describes the transport of User to User Information
   (UUI) during SIP [RFC3261] session setup. This section introduces
   UUI and explains how it relates to SIP.

   We define SIP UUI data as application-specific information that is
   related to a session being established using SIP. It is assumed that
   the application is running in both endpoints in a two party session.
   That is, the application interacts with both the User Agents in a SIP
   session. In order to function properly, the application needs a
   small piece of information, the UUI, to be transported at the time of
   session establishment. This information is essentially opaque data
   to SIP - it is unrelated to SIP routing, authentication, or any other
   SIP function. This application can be considered to be operating at
   a higher layer on the protocol stack. As a result, SIP should not
   interpret, understand, or perform any operations on the UUI. Should
   this not be the case, then the information being transported is not
   considered UUI, and another SIP-specific mechanism will be needed to
   transport the information (such as a new header field). In
   particular, this mechanism creates no requirements on intermediaries
   such as proxies, Back-to-Back User Agents, and Session Border
   Controllers.

   UUI is defined this way for two reasons. Firstly, this supports a
   strict layering of protocols and data. Providing information and
   understanding of the UUI to the transport layer (SIP in this case)
   would not provide any benefits and instead could create cross layer
   coupling. Secondly, it is neither feasible nor desirable for a SIP
   User Agent (UA) to understand the information; instead the goal is
   for the UA to simply pass the information as efficiently as possible
   to the application which does understand the information.

   Note that this document does not discuss the transport of non-call
   control UUI which can be done using the SIP INFO method.

   An important application is the interworking with User to User
   Information (UUI) in ISDN, specifically, the transport of the call
   control related ITU-T Q.931 User to User Information Element (UU IE)
   [Q931] and ITU-T Q.763 User to User Information Parameter [Q763] data
   in SIP. ISDN UUI is widely used in the PSTN today in contact centers
   and call centers. These applications are currently transitioning
   away from using ISDN for session establishment to using SIP. Native
   SIP endpoints will need to implement a similar service and be able to
   interwork with this ISDN service.

   Note that the distinction between call control UUI and non-call
   control UUI is very important. SIP already has a mechanism for

sending arbitrary UUI data between UAs during a session or dialog -
the SIP INFO [RFC6086] method. Call control UUI, in contrast, must
be exchanged at the time of setup and needs to be carried in the
INVITE and a few other methods and responses. Applications that
exchange UUI but do not have a requirement that it be transported and
processed during call setup can simply use SIP INFO and do not need a
new SIP extension.

In this document, four different use case call flows are discussed.
Next, the requirements for call control UUI transport are discussed.

2. Use Cases

This section discusses four use cases for the transport of call
control related user to user information. These use cases will help
motivate the requirements for SIP call control UUI.

2.1. User Agent to User Agent

In this scenario, the originating UA includes UUI in the INVITE sent
through a proxy to the terminating UA. The terminating UA can use
the UUI in any way. If it is an ISDN gateway, it could map the UUI
into the appropriate DSS1 information element or QSIG [QSIG]
information element or ISUP parameter. Alternatively, the using
application might render the information to the user, or use it
during alerting or as a lookup for a screen pop. In this case, the
proxy does not need to understand the UUI mechanism, but normal proxy
rules should result in the UUI being forwarded without modification.
This call flow is shown in Figure 1.

```
        Originating UA             Proxy            Terminating UA
              |                      |                    |
              |    INVITE (UUI) F1   |                    |
              |-------------------->|   INVITE (UUI) F2   |
              |      100 Trying F3  |-------------------->|
              |<-------------------|          200 OK F4  |
              |          200 OK F5 |<-------------------|
              |<-------------------|                    |
              |  ACK F6             |                    |
              |-------------------->|          ACK F7    |
              |                      |-------------------->|
```
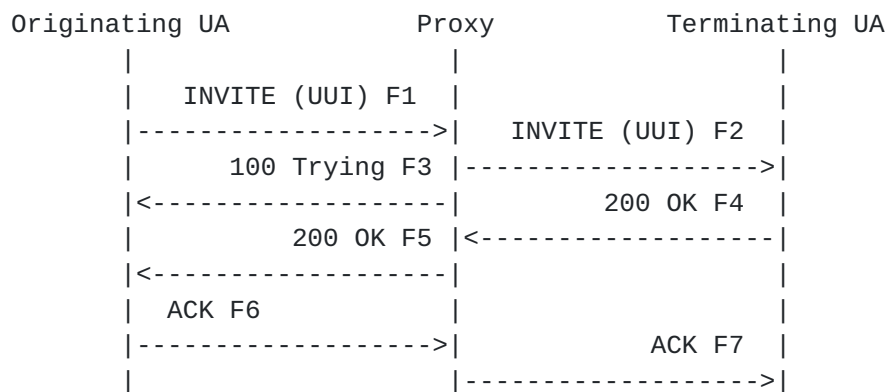
Figure 1. Call flow with UUI exchanged between Originating and
Terminating UAs.

2.2. **Proxy Retargeting**

   In this scenario, the originating UA includes UUI in the INVITE
   request sent through a proxy to the terminating UA. The proxy
   retargets the INVITE request, changing its request-URI to a URI that
   addresses the terminating UA. The UUI data is then received and
   processed by the terminating UA. This call flow is identical to
   Figure 1 except that the proxy retargets the request, i.e., changes
   the Request-URI as directed by some unspecified process. The UUI in
   the INVITE request needs to be passed unchanged through this proxy
   retargeting operation. Note that the contents of the UUI is not used
   by the proxy for routing, as the UUI has only end-to-end significance
   between UAs.

2.3. **Redirection**

   In this scenario, UUI is inserted by an application which utilizes a
   SIP redirect server. The UUI is then included in the INVITE request
   sent by the originating UA to the terminating UA. In this case, the
   originating UA does not necessarily need to support the UUI mechanism
   but does need to support the SIP redirection mechanism used to
   include the UUI data. Two examples of UUI with redirection (transfer
   and diversion) are defined in [ANSII] and [ETSI].

   Note that this case may not precisely map to an equivalent ISDN
   service use case. This is because there is no one-to-one mapping
   between elements in a SIP network and elements in an ISDN network.
   Also, there is not an exact one-to-one mapping between SIP call
   control and ISDN call control. However, this should not prevent the
   usage of SIP call control UUI in these cases. Instead, these slight
   differences between the SIP UUI mechanism and the ISDN service need
   to be carefully noted and discussed in an interworking specification.

   Figure 2 shows this scenario, with the Redirect inserting UUI which
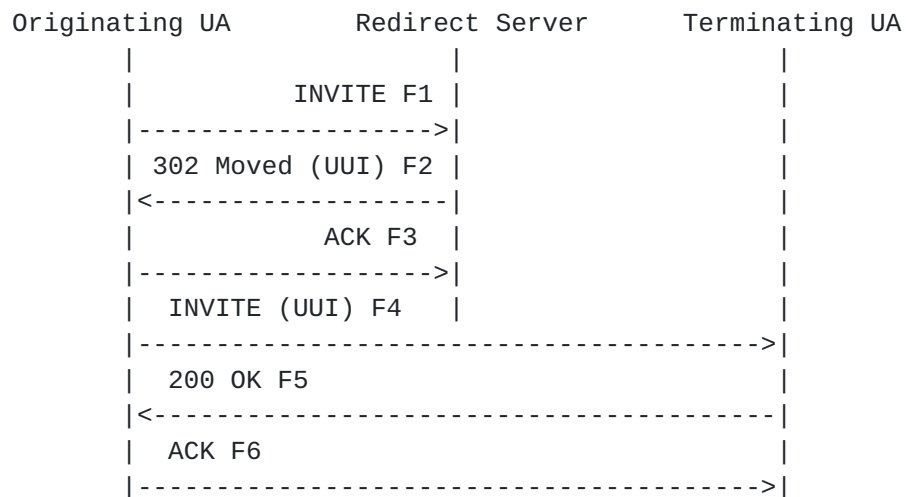   is then included in the INVITE request F4 send to the terminating UA.

```
        Originating UA       Redirect Server      Terminating UA
             |                    |                      |
             |         INVITE F1  |                      |
             |------------------->|                      |
             | 302 Moved (UUI) F2 |                      |
             |<-------------------|                      |
             |          ACK F3    |                      |
             |------------------->|                      |
             |   INVITE (UUI) F4  |                      |
             |---------------------------------------->|
             |   200 OK F5                              |
             |<----------------------------------------|
             |   ACK F6                                 |
             |---------------------------------------->|
```

Figure 2. Call flow with UUI exchanged between Redirect Server and
Terminating UA.

A common example application of this call flow is an Automatic Call
Distributer (ACD) in a PSTN contact center. The originator would be
a PSTN gateway. The ACD would act as a Redirect Server, inserting
UUI based on called number, calling number, time of day, and other
information. The resulting UUI would be passed to the agent's
handset which acts as the terminating UA. The UUI could be used to
lookup information for rendering to the agent at the time of call
answering.

This redirection scenario, and the referral scenario in the next
section, are the most important scenarios for contact center
applications. Incoming calls to a contact center almost always are
redirected or referred to a final destination, sometimes multiple
times, based on collected information and business logic. The
ability to pass along UUI in these call redirection scenarios is
critical.

2.4. Referral

In this scenario, the application uses a UA to initiate a referral,
which causes an INVITE request to be generated between the
originating UA and terminating UA with UUI data inserted by the
Referrer UA. Note that this REFER method [RFC3515] could be part of
a transfer operation or it might be unrelated to an existing call,
such as out-of-dialog REFER request. In some cases, this call flow
is used in place of the redirection call flow: the referrer
immediately answers the call and then sends the REFER request. This
scenario is shown in Figure 3.

```
         Originating UA          Referrer          Terminating UA
              |                     |                     |
              |   REFER (UUI) F1    |                     |
              |<------------------- |                     |
              |   202 Accepted F2   |                     |
              |------------------- >|                     |
              |    INVITE (UUI) F3  |                     |
              |------------------------------------------>|
              | NOTIFY (100 Trying) F4                    |
              |------------------- >|                     |
              |             200 OK F5                     |
              |<------------------- |                     |
              |   200 OK F6                               |
              |<------------------------------------------|
              |   ACK F7                                  |
              |------------------------------------------>|
              | NOTIFY (200 OK) F8 |                      |
              |------------------- >|                     |
              |          200 OK F9  |                     |
              |<------------------- |                     |
```
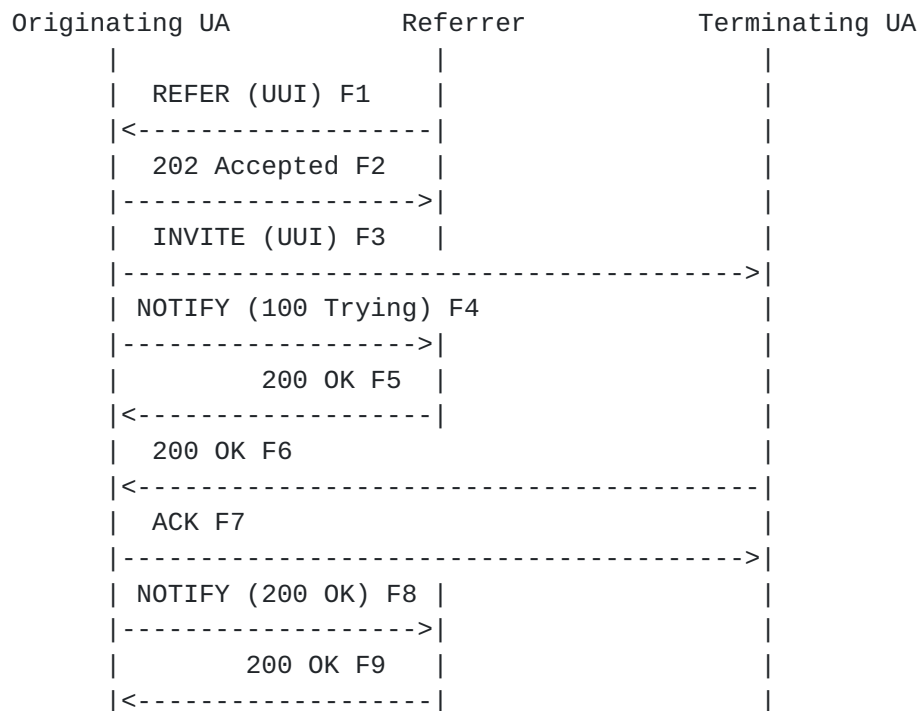
   Figure 3. Call flow with Referral and UUI.

3. Requirements

   This section states the requirements for the transport of call
   control related user to user information (UUI).

   REQ-1: The mechanism will allow UAs to insert and receive UUI data in
   SIP call setup requests and responses.

      SIP messages covered by this include INVITE requests and end-to-
      end responses to the INVITE, i.e. 18x and 200 responses. UUI data
      may also be inserted in 3xx responses to an INVITE. However, if a
      3xx response is recursed on by an intermediary proxy, the
      resulting INVITE will not contain the UUI data from the 3xx
      response. In a scenario where a proxy forks an INVITE to multiple
      UAS who include UUI data in 3xx responses, if a 3xx response is
      the best response sent upstream by the proxy, it will contain the
      UUI data from only one 3xx response.

   REQ-2: The mechanism will allow UAs to insert and receive UUI data in
   SIP dialog terminating requests and responses.

      Q.931 UUI supports inclusion in release and release completion
      messages. SIP messages covered by this include BYE and 200 OK
      responses to a BYE.

REQ-3: The mechanism will allow UUI to be inserted and retrieved in
SIP redirects and referrals.

   SIP messages covered by this include REFER requests and 3xx
   responses to INVITE requests.

REQ-4: The mechanism will allow UUI to be able to survive proxy
retargeting or redirection of the request.

   Retargeting is a common method of call routing in SIP, and must
   not result in the loss of user to user information.

REQ-5: The mechanism should not require processing entities to
dereference a URL in order to retrieve the UUI data.

   Passing a pointer or link to the UUI data will not meet the real-
   time processing considerations and would complicate interworking
   with the PSTN.

REQ-6: The mechanism will support interworking with call control
related DSS1 information elements or QSIG information elements and
ISUP parameters.

REQ-7: The mechanism will allow a UAC to learn that a UAS understands
the UUI mechanism.

REQ-8: The mechanism will allow a UAC to require that a UAS
understands the call control UUI mechanism and have a request routed
based on this information. If the request cannot be routed to a UAS
that understands the UUI mechanism, the request will fail.

   This could be useful in ensuring that a request destined for the
   PSTN is routed to a gateway that supports the UUI mechanism rather
   than an otherwise equivalent PSTN gateway that does not support
   the ISDN mechanism. Note that support of the UUI mechanism does
   not, by itself, imply that a particular application is supported -
   see REQ-10.

REQ-9: The mechanism will allow proxies to remove a particular
application usage of UUI data from a request or response.

   This is a common security function provided by border elements to
   header fields such as Alert-Info or Call-Info URIs. There is no
   requirement for UAs to be able to determine if a particular usage
   of UUI data has been removed from a request or response.

REQ-10: The mechanism will provide the ability for a UA to discover
which application usages of UUI another UA understands or supports.

   The creation of a registry of application usages for the UUI
   mechanism is implied by this requirement. The ISDN Service
   utilizes a field known as the protocol discriminator, which is the
   first octet of the ISDN UUI data, for this purpose.

REQ-11: The UUI is a sequence of octets. The solution will provide a
mechanism of transporting at least 128 octets of user data and a one
octet protocol discriminator, i.e. 129 octets in total.

   There is the potential for non-ISDN services to allow UUI to be
   larger than 128 octets. However, users of the mechanism will need
   be cognizant of the size of SIP messages and the ability of
   parsers to handle extremely large values.

REQ-12: The recipient of UUI will be able to determine the entity
that inserted the UUI. It is acceptable that this is performed
implicitly where it is known that there is only one other end UA
involved in the dialog. Where that does not exist, some other
mechanism will need to be provided. The UUI mechanism does not
introduce stronger authorization requirements for SIP, but instead
the mechanism needs to be able to utilize existing SIP approaches for
request and response identity.

   This requirement comes into play during redirection, retargeting,
   and referral scenarios.

4. Security Considerations

   The security requirements for the UUI mechanism are described in this
   section. It is important to note that UUI security is jointly
   provided at the application layer and at the SIP layer. As such, is
   important for application users of the UUI mechanism to know the
   level of security used and deployed in their particular SIP
   environments, and not to assume that a standardized (but perhaps
   rarely deployed) security mechanism is in place.

   There are three main security models that need to be addressed by the
   UUI mechanism. One model treats the SIP layer as untrusted and
   requires end-to-end integrity protection and/or encryption. This
   model can be achieved by providing these security services at a layer
   above SIP. In this case, the application integrity protects and/or
   encrypts the UUI data before passing it to the SIP layer. This
   method has two advantages: it does not assume or rely on end-to-end
   security mechanisms in SIP which have virtually no deployment, and
   allows the application which understands the contents of the UUI to
   apply a proper level of security. The other approach is for the
   application to pass the UUI without any protection to the SIP layer

and require the SIP layer to provide this security. This approach is
possible in theory, although its practical use would be extremely
limited.

The third model utilizes a trust domain and relies on perimeter
security at the SIP layer. This is the security model of the PSTN
and ISDN where UUI is commonly used today. This approach uses hop-
by-hop security mechanisms and relies on border elements for
filtering and application of policy. This approach is used today in
UUI deployments. Within this approach, there is a requirement that
intermediary elements can detect and remove a UUI element based on
policy, but there is no requirement that an intermediary element be
able to read or interpret the UUI (as the UUI contents only have end-
to-end significance).

The next three requirements capture the UUI security requirements.

REQ-13: The mechanism will allow integrity protection of the UUI.

   This allows the UAS to be able to know that the UUI has not been
   modified or tampered with by intermediaries. Note that there are
   tradeoffs between this requirement and requirement REQ-9 for
   intermediaries to remove UUI. One possible way to satify both of
   these requirements is to utilize hop-by-hop protection. This
   property is not guaranteed by the protocol in the ISDN
   application.

REQ-14: The mechanism will allow end-to-end privacy of the UUI.

   Some UUI may contain private or sensitive information and may
   require different security handling from the rest of the SIP
   message. Note that this property is not available in the ISDN
   application.

REQ-15: The mechanism will allow both end-to-end and hop-by-hop
security models.

   The hop-by-hop model is required by the ISDN UUI service.

[5](#). IANA Considerations

   This document has no IANA requirements.

[6](#). Acknowledgements

   Thanks to Joanne McMillen who was a co-author of earlier versions of

this specification. Thanks to Spencer Dawkins, Keith Drage, Dale
Worley, and Vijay Gurbani for their review of earlier versions of
this document. The authors wish to thank Christer Holmberg,
Frederique Forestie, Francois Audet, Denis Alexeitsev, Paul Kyzivat,
Cullen Jennings, and Mahalingam Mani for their comments on this
topic.

7. Informative References

[RFC3261]   Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston,
            A., Peterson, J., Sparks, R., Handley, M., and E.
            Schooler, "SIP: Session Initiation Protocol", RFC 3261,
            June 2002.

[Q931]   "ITU-T Q.931 User to User Information Element (UU IE)",
            http://www.itu.int/rec/T-REC-Q.931-199805-I/en .

[Q763]   "ITU-T Q.763 Signaling System No. 7 - ISDN user part
            formats and codes",
            http://www.itu.int/rec/T-REC-Q.931-199805-I/en .

[ANSII]   "ANSI T1.643-1995, Telecommunications-Integrated Services
            Digital Network (ISDN)-Explicit Call Transfer
            Supplementary Service".

[ETSI]   "ETSI ETS 300 207-1 Ed.1 (1994), Integrated Services
            Digital Network (ISDN); Diversion supplementary services".

[QSIG]   "ECMA-143 "Private Integrated Services Network (PISN) -
            Circuit Mode Bearer Services - Inter-Exchange Signalling
            Procedures and Protocol" December 2001".

[RFC6086]   Holmberg, C., Burger, E., and H. Kaplan, "Session
            Initiation Protocol (SIP) INFO Method and Package
            Framework", RFC 6086, January 2011.

[RFC3515]   Sparks, R., "The Session Initiation Protocol (SIP) Refer
            Method", RFC 3515, April 2003.

Authors' Addresses

    Alan Johnston
    Avaya
    St. Louis, MO 63124

    Email: alan.b.johnston@gmail.com

    Laura Liess
    Deutsche Telekom AG

    Email: laura.liess.dt@gmail.com