

Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: January 17, 2013

A. Johnston
Avaya
J. Rafferty
Dialogic
July 16, 2012

**A Mechanism for Transporting User to User Call Control Information in
SIP
draft-ietf-cuss-sip-uu-07**

Abstract

There is a class of applications which benefit from using SIP to exchange User to User Information (UUI) data during session establishment. This information, known as call control UUI data, is a small piece of data inserted by an application initiating the session, and utilized by an application accepting the session. The rules which apply for a specific application are defined by a UUI package. This UUI data is opaque to SIP and its function is unrelated to any basic SIP function. This document defines a new SIP header field, User-to-User, to transport UUI data, along with an extension mechanism.

Status of this Memo

This Internet-Draft is submitted to IETF 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 January 17, 2013.

Copyright Notice

Copyright (c) 2012 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents

(<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents 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	3
2.	Terminology	3
3.	Requirements Discussion	3
4.	Normative Definition	5
4.1.	Syntax for UII Header Field	6
4.2.	Hex Encoding Definition	7
4.3.	Source Identity of UII data	7
5.	Guidelines for UII Packages	8
5.1.	Extensibility	10
6.	IANA Considerations	10
6.1.	Registration of User-to-User Header Field	10
6.2.	Registration of User-to-User Header Field Parameters	11
6.3.	Registration of UII Packages	11
6.4.	Registration of UII Content Parameters	12
6.5.	Registration of UII Encoding Parameters	12
6.6.	Registration of SIP Option Tag	12
7.	Security Considerations	13
8.	Appendix - Other Possible Mechanisms	13
8.1.	Why INFO is Not Used	13
8.2.	Why Other Protocol Encapsulation UII Mechanisms are Not Used	14
8.3.	MIME body Approach	14
8.4.	URI Parameter	15
9.	Acknowledgements	16
10.	References	16
10.1.	Informative References	16
10.2.	Normative References	17
	Authors' Addresses	17

1. Overview

This document describes the transport of User to User Information (UUI) data using SIP [[RFC3261](#)]. A mechanism is defined for the transport of general application UUI data and for the transport of call control related ITU-T Q.931 User to User Information Element (UUIE) [[Q931](#)] and ITU-T Q.763 User to User Information Parameter [[Q763](#)] data in SIP. UUI data is widely used in the PSTN today for contact centers and call centers. There is also a trend for the related applications to transition from ISDN to SIP. The UUI extension for SIP may also be used for native SIP UAs implementing similar services and to interwork with ISDN services. Note that in most cases, there is an a priori understanding between the UAs in regard to what to do with received UUI data.

This mechanism was designed to meet the use cases, requirements, and call flows for SIP call control UUI detailed in [[RFC6567](#)]. All references to requirement numbers (REQ-N) and figure numbers refer to this document.

The mechanism is a new SIP header field, along with a new SIP option tag. The header field carries the UUI data, along with parameters indicating the encoding of the UUI data, the UUI package, and optionally the content of the UUI data. The package definition contains details about how a particular application can utilize the UUI mechanism. The header field can be included (sometimes called "escaped") into URIs supporting referral and redirection scenarios. In these scenarios, History-Info is used to indicate the inserter of the UUI data. The SIP option tag can be used to indicate support for the header field. Support for the UUI header field indicates that a UA is able to extract the information in the UUI data and pass it up the protocol stack. Individual packages using the UUI mechanism can utilize SIP media feature tags to indicate that a UA supports a particular UUI package. Guidelines for defining UUI packages are provided.

2. Terminology

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 [BCP 14](#), [RFC 2119](#) [[RFC2119](#)].

3. Requirements Discussion

This section describes how the User-to-User header field meets the

requirements in [[RFC6567](#)]. The header field can be included in INVITE requests and responses and BYE requests and responses, meeting REQ-1 and REQ-2.

For redirection and referral use cases and REQ-3, the header field shall be included (escaped) into the Contact or Refer-To URI. Currently, UAs that support attended transfer support the ability to include a Replaces header field into a Refer-To URI, and when acting upon this URI add the Replaces header field to the triggered INVITE. This logic and behavior is identical for the UUI header field. The UA processing the REFER or the 3xx to the INVITE will need to support the UUI mechanism, as UAs in general do not process unknown included header fields.

Since SIP proxy forwarding and retargeting does not affect header fields, the header field meets REQ-4.

The UUI header field will carry the UUI data and not a pointer to the data, so REQ-5 is met.

Since the basic design of the UUI header field is similar to the ISDN UUI service, interworking with PSTN protocols is straightforward and is documented in a separate specification [[I-D.ietf-cuss-sip-uui-isdn](#)], meeting REQ-6.

Requirements REQ-7, REQ-8, and REQ-10 relate to discovery of the mechanism and supported packages, and hence applications. REQ-7 relates to support of the UUI header field, while REQ-8 relates to routing based on support of the UUI header field. REQ-7 is met by defining a new SIP option tag 'uui'. The use of a Require:uui in a request, or Supported:uui in an OPTIONS response could be used to require or discover support of the mechanism. The presence of a Supported:uui or Require:uui header field can be used by proxies to route to an appropriate UA, meeting REQ-8. However, note that only UAs are expected to understand the UUI data - proxies and other intermediaries do not. REQ-10 is met by utilizing SIP feature tags [[RFC3840](#)]. For example, the feature tag 'sip.uui-isdn' could be used to indicate support of the ISDN UUI package, or 'sip.uui-pk1' could be used to indicate support for a particular package, pk1.

Proxies commonly apply policy to the presence of certain SIP header fields in requests by either passing them or removing them from requests. REQ-9 is met by allowing proxies and other intermediaries to remove UUI header fields in a request or response based on policy.

Carrying UUI data elements of at least 129 octets is trivial in the UUI header field, meeting REQ-11. Note that very large UUI data elements should be avoided, as SIP header fields have traditionally

not been large.

To meet REQ-12 for the redirection and referral use cases, History-Info [[I-D.ietf-sipcore-rfc4244bis](#)] can be used. In these retargeting cases, the changed Request-URI will be recorded in the History-Info header field along with the identity of the element that performed the retargeting.

The requirement for integrity protection in REQ-13 could be met by the use of an S/MIME signature over a subset of header fields, as defined in [Section 23.4 of RFC 3261](#) "SIP Header Privacy and Integrity using S/MIME: Tunneling SIP". The requirement of REQ-14 for end-to-end privacy could be met using S/MIME or using encryption at the application layer. Note that the use of S/MIME to secure the UUI data will result in an additional body being added to the request. Hopwise TLS allows the header field to meet REQ-15 for hop-by-hop security.

4. Normative Definition

This document defines a new SIP header field "User-to-User" to transport call control UUI data to meet the requirements in [[RFC6567](#)].

To help tag and identify the UUI data used with this header field, "purpose", "content", and "encoding" header field parameters are defined. The "purpose" header field parameter identifies the package which defines the generation and usage of the UUI data for a particular application. For the case of interworking with the ISDN UUI Service, the ISDN UUI Service interworking package is used. If the "purpose" header field parameter is not present, interworking with the ISDN UUI Service MUST be assumed. The "content" header field parameter identifies the actual content of the UUI data. If not present, the content MUST be assumed to be the default defined for the package. Newly defined UUI packages MUST define or reference at least a default "content" value. The "encoding" header field parameter indicates the method of encoding the information in the UUI data associated with a particular "content" value. This specification only defines "encoding=hex". If the "encoding" header field parameter is not present, the encoding MUST be assumed to be the default defined for the package.

UUI data is considered an opaque series of octets. This mechanism SHOULD NOT be used to convey a URL or URI; the Call-Info header field [[RFC3261](#)] is used for this purpose.

4.1. Syntax for UUI Header Field

The User-to-User header field can be present in INVITE requests and responses only and in BYE requests and responses. Note that only end-to-end responses can be used, e.g. 1xx (excluding 100), 2xx, and 3xx responses.

The following syntax specification uses the augmented Backus-Naur Form (BNF) as described in [RFC 5234](#) and extends [RFC 3261](#) (where token and quoted-string are defined).

```
UUI           = "User-to-User" HCOLON uui-value *(COMMA uui-value)
uui-value     = uui-data *(SEMI uui-param)
uui-data      = token / quoted-string
uui-param     = pkg-param / cont-param / enc-param / generic-param
pkg-param     = "purpose" EQUAL token
cont-param    = "content" EQUAL token
enc-param     = "encoding" EQUAL ("hex" / token)
```

The rules for how many User-to-User header field of each package may be present in a request or a response are defined for each package. The syntax allows any combination of individual User-to-User header fields or User-to-User header fields with multiple comma separated UUI data elements. Multiple User-to-User header fields MAY be present in a request or response. Any size limitations on the UUI data for a particular purpose must be defined by the related UUI package.

UAs SHOULD ignore UUI data from packages or encoding that they do not understand.

If an element supports this specification, it SHOULD include any UUI data included in a redirection URI. Note that redirection can occur multiple times to a request.

Here is an example of an included User-to-User header field from the redirection response F2 of Figure 2:

```
<allOneLine>
Contact: <sip:+12125551212@gateway.example.com?User-to-User=
56a390f3d2b7310023a2%3Bencoding%3Dhex%3Bpurpose%3Dfoo%3B
content%3Dbar>
</allOneLine>
```

The resulting INVITE F4 would contain:

User-to-User: 56a390f3d2b7310023a2;encoding=hex;purpose=foo;content=bar

4.2. Hex Encoding Definition

This specification defines hex encoding of UUI data. The value of "hex" for the "encoding" header field parameter is normatively defined in this section. It is used to encode binary UUI data with a length that terminates at an octet boundary. Each octet of binary data to be represented in the hex encoding MUST be mapped to two hexadecimal digits (represented by ASCII characters 0-9, A-F and a-f), each representing four bits within the octet. The four bits appearing first in the binary UUI data MUST be mapped to the first hexadecimal digit and the four subsequent bits in the binary UUI data MUST be mapped to the second hexadecimal digit. When mapping 4 bits to a hexadecimal digit, the bit appearing first in the binary UUI data shall be most significant. Thus, Hex encoded UUI data must have an even number of hexadecimal digits, and MUST be considered invalid if it has an odd number. Hex encoding is normally done as a token, although quoted-string is permitted, in which case the quotes MUST be ignored.

4.3. Source Identity of UUI data

It is important for the recipient of UUI data to know the identity of the UA that inserted the UUI data. In a request without a History-Info [[I-D.ietf-sipcore-rfc4244bis](#)] header field, the identity of the entity which inserted the UUI data will be assumed to be the source of the SIP message. For a SIP request, typically this is the UA identified by the URI in the From header field or a P-Asserted-Identity [[RFC3325](#)] header field. In a request with a History-Info header field, the recipient needs to parse the Targeted-to-URIs present (hi-targeted-to-uri) to see if any included User-to-User header fields are present. If an included User-to-User header field is present and matches the UUI data in the request, this indicates that redirection has taken place, resulting in the inclusion of UUI data in the request. The inserter of the UUI data will be the UA identified by the Targeted-to-URI of the History-Info element prior to the element with the included UUI data. In a response, the inserter of the UUI data will be the identity of the UA that generated the response. Typically, this is the UA identified in the To header field of the response. Note that any updates to this identity by use of the SIP Connected Identity extension [[RFC4916](#)] or others will update this information.

For an example of History-Info and redirection, consider Figure 2 from [[RFC6567](#)] where the Originating UA is Carol, the Redirector Bob, and the Terminating UA Alice. The INVITE F4 containing UUI data could be:


```
INVITE sips:alice@example.com SIP/2.0
Via: SIP/2.0/TLS lab.example.com:5061
    ;branch=z9hG4bKnashds9
To: Bob <sips:bob@example.com>
From: Carol <sips:carol@example.com>;tag=323sf33k2
Call-ID: dfaosidfoiwe83ifkdf
Max-Forwards: 70
Contact: <sips:carol@lab.example.com>
Supported: histinfo
User-to-User: 342342ef34;encoding=hex
History-Info: <sips:bob@example.com>;index=1
<allOneLine>
History-Info: <sips:alice@example.com?Reason=SIP%3Bcause%3D302
    &User-to-User=342342ef34%3Bencoding%3Dhex>;index=1.1;rc=1
</allOneLine>
```

Without the redirection captured in the History-Info, Alice would conclude the UUI data was inserted by Carol. However, the History-Info containing UUI data (index=1.1) indicates that the inserter was Bob (index=1).

Note that the <allOneLine> tag convention from SIP Torture Test Messages [[RFC4475](#)] is used to show that there are no line breaks in the actual message syntax.

To enable maintaining a record of the inserter identity of UUI data, UAs supporting this mechanism SHOULD support History-Info [[I-D.ietf-sipcore-rfc4244bis](#)] and include Supported: histinfo in all requests and responses.

Border elements such as proxies or B2BUAs which anonymize a SIP URI in a History-Info header field SHOULD leave the corresponding User-to-User parameter, if present, and the corresponding User-to-User header field unchanged. Border elements removing a History-Info header containing a User-to-User parameter SHOULD NOT drop the corresponding User-to-User header. Otherwise, the UA consuming the UUI data may not be able at SIP level to identify the source of the UUI data.

5. Guidelines for UUI Packages

UUI packages defined using this SIP UUI mechanism MUST publish a standards track RFC which describes the usage. Note that this mechanism is not suitable for the transport of arbitrary data between UAs. The following guidelines are provided to help determine if this mechanism is appropriate or some other SIP mechanism should be used. The SIP UUI mechanism is applicable when all of the following

conditions be met:

1. The information is generated and consumed by an application during session setup using SIP, but the application is not necessarily SIP aware.
2. The behavior of SIP entities that support it is not significantly changed (as discussed in [Section 4 of \[RFC5727\]](#)).
3. User Agents (UAs) are the generators and consumers of the UUI data. Proxies and other intermediaries may route based on the presence of a User-to-User header field or a particular package tag but do not otherwise consume or generate the UUI data.
4. There are no overriding privacy issues associated with the information being transported (e.g., geolocation or emergency-related information are examples of inappropriate UUI data).
5. The UUI data is not being utilized for user-to-user Remote Procedure Call (RPC) calls.

UUI packages define the semantics for a particular application usage of UUI data. The content defines the syntax of the UUI data, while the encoding defines the encoding of the UUI data for the content. Each content is defined as a stream of octets, which allows multiple encodings of that content. For example, packages may define:

1. The SIP methods and responses in which the UUI data may be present.
2. The maximum number of UUI data elements that may be inserted into a request or response. (The default is one per encoding.) Note that a UA may still receive a request with more than this maximum number due to redirection. The package must define how to handle this situation.
3. The default values for content and encoding if they are not present. If the same UUI data may be inserted multiple times with different encodings, the packages must state this. A package may support and define multiple contents and their associated encodings, and reuse contents defined by other packages.
4. Any size limitations on the UUI data. Size should be specified in terms of the octet stream output of the content, since the size of the resulting uui-data element will vary depending on the encoding scheme.

A package MUST define a "purpose" header field value to identify the

package in the coding. A package MUST describe the new application which is utilizing the UUI data and provide some use case examples. The default "content" value MUST be defined or referenced in another document for the package. Additional allowed contents MAY also be defined or referenced. Any restrictions on the size of the UUI data must be described. In addition, a package may define a Media Feature tag per [RFC 3840](#) [[RFC3840](#)] to indicate support for this UUI package. For example, the media feature tag sip.uui-pk1 could be defined to indicate support for a UUI package named pk1. The definition of a new SIP option tag solely to identify support for a UUI package is NOT RECOMMENDED unless there are additional SIP behaviors needed to implement this feature.

For an example UUI package definition, see [\[I-D.ietf-cuss-sip-uui-isdn\]](#).

[5.1.](#) Extensibility

New "content" values MUST describe the semantics of the UUI data, valid encodings, and give some example use cases. A previously defined UUI content value can be used in a new package. In this case, the semantics and usage of the content by the new package is defined within the new package. New UUI content types cannot be added to existing packages - instead, a new package would need to be defined. New content values defined are added to the IANA registry with a standards track RFC, which needs to discuss the issues in this section. If no new encoding value is defined for a content, the encoding defaults to "hex" as defined in this document. In this case, the "hex" value will be explicitly stated via the encoding parameter as the encoding for the content.

New "encoding" values associated with a new content MUST reference a specific encoding scheme (such as "hex" which is defined in this specification) or define the new encoding scheme. A previously defined UUI encoding value can be used with a newly defined content. In this case, the usage of the encoding is defined by the content definition. New UUI encodings cannot be added to existing contents - instead, a new content would need to be defined. Newly defined encoding values are added to the IANA registry with a standards track RFC, which needs to discuss the issues in this section.

[6.](#) IANA Considerations

[6.1.](#) Registration of User-to-User Header Field

This document defines a new SIP header field named "User-to-User".

The following row shall be added to the "Header Fields" section of the SIP parameter registry:

Header Name	Compact Form	Reference
User-to-User		[RFCXXXX]

Editor's Note: [RFCXXXX] should be replaced with the designation of this document.

6.2. Registration of User-to-User Header Field Parameters

This document defines the parameters for the header field defined in the preceding section. The header field "User-to-User" can contain the parameters "encoding", "content", and "purpose".

The following rows shall be added to the "Header Field Parameters and Parameter Values" section of the SIP parameter registry:

Header Field	Parameter Name	Predefined Values	Reference
User-to-User	encoding	hex	[RFCXXXX]
User-to-User	content		[RFCXXXX]
User-to-User	purpose		[RFCXXXX]

Editor's Note: [RFCXXXX] should be replaced with the designation of this document.

6.3. Registration of UII Packages

This specification establishes the uui-packages sub-registry under <http://www.iana.org/assignments/sip-parameters>. New uui-packages shall be registered by standards track RFC publication.

The descriptive text for the table of uui-content is:

UII Packages provides information about the usage of the UII data in a User-to-User header field [RFCXXXX].

Package	Description	Reference
---------	-------------	-----------

6.4. Registration of UUI Content Parameters

This specification establishes the uui-content sub-registry under <http://www.iana.org/assignments/sip-parameters>. New uui-content values shall be registered by standards track RFC publication.

The descriptive text for the table of uui-content is:

UUI Content provides information about the content of the UUI data in a User-to-User header field [RFCXXXX].

+-----+-----+-----+		
Content	Description	Reference
+-----+-----+-----+		

6.5. Registration of UUI Encoding Parameters

This specification establishes the uui-encoding sub-registry under <http://www.iana.org/assignments/sip-parameters> and initiates its population with the table below. Additional uui-encoding values shall be registered by a standards track RFC publication.

The descriptive text for the table of uui-encoding is:

UUI Encoding provides information about the encoding of the UUI data in a User-to-User header field [RFCXXXX].

+-----+-----+-----+		
Encoding	Description	Reference
+-----+-----+-----+		
hex	The UUI data is encoded using hexadecimal	[RFCXXXX]
+-----+-----+-----+		

6.6. Registration of SIP Option Tag

This specification registers a new SIP option tag, as per the guidelines in [Section 27.1 of \[RFC3261\]](#).

This document defines the SIP option tag "uui".

The following row has been added to the "Option Tags" section of the SIP Parameter Registry:

+-----+ Name	+-----+ Description	+-----+ Reference
+-----+ uui	+-----+ This option tag is used to indicate that	+-----+ [RFCXXXX]
	a UA supports and understands the	
	User-to-User header field.	
+-----+	+-----+	+-----+

Editor's Note: [RFCXXXX] should be replaced with the designation of this document.

7. Security Considerations

User to user information can potentially carry sensitive information that might require privacy or integrity protection. Standard deployed SIP security mechanisms such as TLS transport, offer these properties on a hop-by-hop basis. To preserve multi-hop or end-to-end confidentiality and integrity of UUI data, approaches using S/MIME or IPSec can be used, as discussed in the draft. However, the lack of deployment of these mechanisms means that applications can not in general rely on them. As such, applications are encouraged to utilize their own security mechanisms.

If the UUI data was included by the UA originator of the SIP request or response, normal SIP mechanisms can be used to determine the identity of the inserter of the UUI data. If the UUI data was included by a UA that was not the originator of the request, History-Info can be used to determine the identity of the inserter of the UUI data. UAs can apply policy based on the origin of the UUI data using this information.

8. Appendix - Other Possible Mechanisms

Two other possible mechanisms for transporting UUI data will be described: MIME body and URI parameter transport.

8.1. Why INFO is Not Used

Since the INFO method [[RFC6086](#)], was developed for ISUP interworking of user-to-user information, it might seem to be the logical choice here. For non-call control user-to-user information, INFO can be utilized for end to end transport. However, for transport of call control user-to-user information, INFO can not be used. As the call flows in [[RFC6567](#)] show, the information is related to an attempt to establish a session and must be passed with the session setup request (INVITE), responses to that INVITE, or session termination requests.

As a result, it is not possible to use INFO in these cases.

8.2. Why Other Protocol Encapsulation UUI Mechanisms are Not Used

Other protocols have the ability to transport UUI data. For example, consider the ITU-T Q.931 User to User Information Element (UU IE) [[Q931](#)] and the ITU-T Q.763 User to User Information Parameter [[Q763](#)]. In addition, NSS (Narrowband Signaling System) [[Q1980](#)] is also able to transport UUI data. Should one of these protocols be in use, and present in both User Agents, then utilizing these other protocols to transport UUI data might be a logical solution. Essentially, this is just adding an additional layer in the protocol stack. In these cases, SIP is not transporting the UUI data; it is encapsulating another protocol, and that protocol is transporting the UUI data. Once a mechanism to transport that other protocol using SIP exists, the UUI data transport function is essentially obtained without any additional effort or work.

However, the authors believe that SIP needs to have its own native UUI data transport mechanism. It is not reasonable for a SIP UA to have to implement another entire protocol (either ISDN or NSS, for example) just to get the very simple UUI data transport service. Of course, this work does not preclude anyone from using other protocols with SIP to transport UUI data.

8.3. MIME body Approach

One method of transport is to use a MIME body. This is in keeping with the SIP-T architecture [[RFC3372](#)] in which MIME bodies are used to transport ISUP information. Since the INVITE will normally have an SDP message body, the resulting INVITE with SDP and UUI data will be multipart MIME. This is not ideal as many SIP UAs do not support multipart MIME INVITES.

A bigger problem is the insertion of a UUI message body by a redirect server or in a REFER. The body would need to be encoded in the Contact URI of the 3xx response or the Refer-To URI of a REFER. Currently, the authors are not aware of any UAs that support this capability today for any body type. As such, the complete set of semantics for this operation would need to be determined and defined. Some issues will need to be resolved, such as, do all the Content-* header fields have to be included as well? And, what if the included Content-Length does not agree with the included body?

Since proxies cannot remove a body from a request or response, it is not clear how this mechanism could meet REQ-9.

The requirement for integrity protection could be met by the use of

an S/MIME signature over the body, as defined in Section 23.3 of [RFC 3261](#) "Securing MIME bodies". Alternatively, this could be achieved using [RFC 4474](#) [[RFC4474](#)]. The requirement for end-to-end privacy could be met using S/MIME encryption or using encryption at the application layer. However, note that neither S/MIME or [RFC 4474](#) enjoys deployment in SIP today.

An example:

```
<allOneLine>
Contact: <sip:+12125551212@gateway.example.com?Content-Type=
application/uui&body=ZeGl9i2icVqaNVailT6F5iJ90m6mvuTS40K05M0vDk0Q4Xs>
</allOneLine>
```

As such, the MIME body approach meets REQ-1, REQ-2, REQ-4, REQ-5, REQ-7, REQ-11, REQ-13, and REQ-14. Meeting REQ-12 seems possible, although the authors do not have a specific mechanism to propose. Meeting REQ-3 is problematic, but not impossible for this mechanism. However, this mechanism does not seem to be able to meet REQ-9.

[8.4.](#) URI Parameter

Another proposed approach is to encode the UUI data as a URI parameter. This UUI parameter could be included in a Request-URI or in the Contact URI or Refer-To URI. It is not clear how it could be transported in a responses which does not have a Request-URI, or in BYE requests or responses.

```
<allOneLine>
Contact: <sip:+12125551212@gateway.example.com;uui=ZeGl9i2icVqaNVailT6
F5iJ90m6mvuTS40K05M0vDk0Q4Xs>
</allOneLine>
```

An INVITE sent to this Contact URI would contain UUI data in the Request-URI of the INVITE. The URI parameter has a drawback in that a URI parameter carried in a Request-URI will not survive retargeting by a proxy as shown in Figure 2 of [[RFC6567](#)]. That is, if the URI is included with an Address of Record instead of a Contact URI, the URI parameter in the Request-URI will not be copied over to the Contact URI, resulting in the loss of the information. Note that if this same URI was present in a Refer-To header field, the same loss of information would occur.

The URI parameter approach would meet REQ-3, REQ-5, REQ-7, REQ-9, and REQ-11. It is possible the approach could meet REQ-12 and REQ-13. The mechanism does not appear to meet REQ-1, REQ-2, REQ-4, and REQ-14.

9. Acknowledgements

Joanne McMillen was a major contributor and co-author of earlier versions of this document. Thanks to Paul Kyzivat for his contribution of hex encoding rules. Thanks to Spencer Dawkins, Keith Drage, Vijay Gurbani, and Laura Liess for their review of the document. The authors wish to thank Roland Jesske, Celine Serrut-Valette, Francois Audet, Denis Alexeitsev, Paul Kyzivat, Cullen Jennings, and Mahalingam Mani for their comments.

10. References

10.1. Informative References

- [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> .
- [Q931] "ITU-T Q.931 User to User Information Element (UU IE)",
<http://www.itu.int/rec/T-REC-Q.931-199805-I/en> .
- [ETSI] "ETSI ETS 300 207-1 Ed.1 (1994), Integrated Services Digital Network (ISDN); Diversion supplementary services".
- [RFC3372] Vemuri, A. and J. Peterson, "Session Initiation Protocol for Telephones (SIP-T): Context and Architectures",
[BCP 63](#), [RFC 3372](#), September 2002.
- [RFC6086] Holmberg, C., Burger, E., and H. Kaplan, "Session Initiation Protocol (SIP) INFO Method and Package Framework", [RFC 6086](#), January 2011.
- [RFC4475] Sparks, R., Hawrylyshen, A., Johnston, A., Rosenberg, J., and H. Schulzrinne, "Session Initiation Protocol (SIP) Torture Test Messages", [RFC 4475](#), May 2006.
- [RFC5727] Peterson, J., Jennings, C., and R. Sparks, "Change Process for the Session Initiation Protocol (SIP) and the Real-time Applications and Infrastructure Area", [BCP 67](#), [RFC 5727](#), March 2010.
- [I-D.ietf-cuss-sip-uui-isdn]
Drage, K. and A. Johnston, "Interworking ISDN Call Control User Information with SIP",
[draft-ietf-cuss-sip-uui-isdn-04](#) (work in progress),
May 2012.

- [Q1980] "ITU-T Q.1980.1 The Narrowband Signalling Syntax (NSS) - Syntax Definition", <http://www.itu.int/itudoc/itu-t/aap/sg11aap/history/q1980.1/q1980.1.html> .

10.2. Normative References

- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), March 1997.
- [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.
- [RFC3324] Watson, M., "Short Term Requirements for Network Asserted Identity", [RFC 3324](#), November 2002.
- [RFC3325] Jennings, C., Peterson, J., and M. Watson, "Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks", [RFC 3325](#), November 2002.
- [RFC6567] Johnston, A. and L. Liess, "Problem Statement and Requirements for Transporting User-to-User Call Control Information in SIP", [RFC 6567](#), April 2012.
- [RFC4474] Peterson, J. and C. Jennings, "Enhancements for Authenticated Identity Management in the Session Initiation Protocol (SIP)", [RFC 4474](#), August 2006.
- [I-D.ietf-sipcore-rfc4244bis]
Barnes, M., Audet, F., Schubert, S., Elburg, H., and C. Holmberg, "An Extension to the Session Initiation Protocol (SIP) for Request History Information", [draft-ietf-sipcore-rfc4244bis-08](#) (work in progress), April 2012.
- [RFC4916] Elwell, J., "Connected Identity in the Session Initiation Protocol (SIP)", [RFC 4916](#), June 2007.
- [RFC3840] Rosenberg, J., Schulzrinne, H., and P. Kyzivat, "Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)", [RFC 3840](#), August 2004.

Authors' Addresses

Alan Johnston
Avaya
St. Louis, MO 63124

Email: alan.b.johnston@gmail.com

James Rafferty
Dialogic

Email: james.rafferty@dialogic.com