A Session Initiation Protocol (SIP) Extension for the Identification of
                                Services
               draft-drage-sipping-service-identification-03

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), its areas, and its working groups. Note that
   other groups may also distribute working documents as Internet-
   Drafts.

   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."

   The list of current Internet-Drafts can be accessed at
   http://www.ietf.org/ietf/1id-abstracts.txt.

   The list of Internet-Draft Shadow Directories can be accessed at
   http://www.ietf.org/shadow.html.

   This Internet-Draft will expire on September 26, 2009.

Copyright Notice

Abstract

   This document describes private extensions to the Session Initiation
   Protocol (SIP) that enable a network of trusted SIP servers to assert
   the service of authenticated users. The use of these extensions is
   only applicable inside an administrative domain with previously
   agreed-upon policies for generation, transport and usage of such
   information. This document does NOT offer a general service
   identification model suitable for use between different trust
   domains, or use in the Internet at large.

   The document also defines a URN to identify both services and UA
   applications. This URN can be used within the SIP header fields
   defined in this document to identify services, and also within the
   framework defined for caller preferences and callee capabilities to
   identify usage of both services and applications between end UAs.

1. Introduction

   This document describes private extensions to the Session Initiation
   Protocol (SIP) that enable a network of trusted SIP servers to assert
   the service possibly subject to the user being entitled to that
   service. The use of these extensions is only applicable inside an
   administrative domain with previously agreed-upon policies for
   generation, transport and usage of such information. This document
   does NOT offer a general service model suitable for use between
   different trust domains, or use in the Internet at large.

   The concept of "service" within SIP has no hard and fast rules.
   draft-ietf-sipping-service-identification
   [I-D.ietf-sipping-service-identification] provides general guidance
   on what constitutes a service within SIP and what does not.

   This document also makes use of the terms "derived service
   identification" and "declarative service identification" as defined
   in draft-ietf-sipping-service-identification
   [I-D.ietf-sipping-service-identification].

   During a session setup proxies may need to understand what service
   the request is related to in order to know what application server to
   contact or other service logic to invoke. The SIP INVITE request
   contains all of the information necessary to determine the service.
   However, the calculation of the service may be computational and
   database intensive. For example, a given trust domain's definition
   of a service might include request authorization. Moreover the
   analysis may require examination of the SDP.

   For example, an INVITE request with video SDP directed to a video-on-
   demand Request-URI could be marked as an IPTV session. An INVITE
   request with push-to-talk over cellular (PoC) routes could be marked
   as a PoC session. An INVITE request with a Require header field
   containing an option tag of "foogame" could be marked as a foogame
   session.

   NOTE: If the information contained within the SIP INVITE request is
   not sufficient to uniquely identify a service, the remedy is to
   extend the SIP signalling to capture the missing element.
   draft-ietf-sipping-service-identification
   [I-D.ietf-sipping-service-identification] provides further
   explanation.

   By providing a mechanism to compute and store the results of the
   domain specific service calculation, i.e. the derived service
   identification, this optimization allows a single trusted proxy to
   perform an analysis of the request and authorize the requestor's

permission to request such a service. The proxy may then include a
service identifier that relieves other trusted proxies and trusted
UAs from performing further duplicate analysis of the request for
their service identification purposes. In addition, this extension
allows user agent clients outside the trust domain to provide a hint
of the requested service.

This extension does not provide for the dialog or transaction to be
rejected if the service is not supported end-to-end. SIP provides
other mechanisms, such as the option-tag and use of the Require and
Proxy-Require header fields, where such functionality is required.
No explicitly signalled service identification exists and the session
proceeds for each nodes definition of the service in use, on the
basis of information contained in SDP and in other SIP header fields.

This mechanism is specifically a mechanism to manage the information
needs of intermediate routeing devices between the calling user and
the user represented by the Request-URI. In support of this
mechanism, a URN is defined to identify the services. This URN has
wider applicability to additionally identify services and terminal
applications. Between end users, caller preferences and callee
capabilities as specified in RFC 3840 [RFC3840] and RFC 3841
[RFC3841] provide an appropriate mechanism for indicating such
service and application identification. These mechanisms have been
extended by draft-rosenberg-sip-app-media-tag
[I-D.rosenberg-sip-app-media-tag] to provide further capabilities in
this area.

The mechanism proposed in this document relies on a new header field
called 'P-Asserted-Service' that contains a URN. This is supported
by a further new header field called 'P-Preferred-Service' that also
contains a URN, and which allows the UA to express a preferences to
the decisions made on service within the trust domain.

P-Asserted-Service: urn:urn-7:3gpp-service.exampletelephony.version1

A proxy server which handles a request can, after authenticating the
originating user in some way (for example: Digest authentication), to
ensure that the user is entitled to that service, insert such a
P-Asserted-Service header field into the request and forward it to
other trusted proxies. A proxy that is about to forward a request to
a proxy server or UA that it does not trust removes all the
P-Asserted-Service header field values.

This document labels services by means of an informal URN. This
provides a hierarchical structure for defining services and
subservices, and provides an address that can be resolvable for
various purposes outside the scope of this document, e.g. to obtain

information about the service so described.

2. Applicability Statement

This document describes private extensions to SIP (see RFC 3261
[RFC3261]) that enable a network of trusted SIP servers to assert the
service of end users or end systems. The use of these extensions is
only applicable inside a 'Trust Domain' as defined in Short term
requirements for Network Asserted Identity (see RFC 3324 [RFC3324]).
Nodes in such a Trust Domain are explicitly trusted by its users and
end-systems to publicly assert the service of each party, and that
they have common and agreed upon definitions of services and
homogeneous service offerings. The means by which the network
determines the service to assert is outside the scope of this
document (though it commonly entails some form of authentication).

The mechanism for defining a trust domain is to provide a certain set
of specifications known as 'Spec(T)', and they specify compliance to
that set of specifications. Spec(T) MUST specify behavior as
documented in RFC 3324 [RFC3324].

This document does NOT offer a general service model suitable for
inter-domain use or use in the Internet at large. Its assumptions
about the trust relationship between the user and the network may not
apply in many applications. For example, these extensions do not
accommodate a model whereby end users can independently assert their
service by use of the extensions defined here. End users assert
their service by including the SIP and SDP parameters that correspond
to the service they require. Furthermore, since the asserted
services are not cryptographically certified, they are subject to
forgery, replay, and falsification in any architecture that does not
meet the requirements of RFC 3324 [RFC3324].

The asserted services also lack an indication of who specifically is
asserting the service, and so it must be assumed that a member of the
Trust Domain is asserting the service. Therefore, the information is
only meaningful when securely received from a node known to be a
member of the Trust Domain.

Despite these limitations, there are sufficiently useful specialized
deployments that meet the assumptions described above, and can accept
the limitations that result, to warrant informational publication of
this mechanism.

3. Conventions

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].

Throughout this document requirements for or references to proxy
servers or proxy behavior apply similarly to other intermediaries
within a Trust Domain (ex: B2BUAs).

The term Trust Domain in this document has the meaning as defined in
RFC 3324 [RFC3324].

4. Syntax of the Header Fields

   The following syntax specification uses the augmented Backus-Naur
   Form (BNF) as described in RFC 4234 [RFC5234].

4.1. The P-Asserted-Service Header

   The P-Asserted-Service header field is used among trusted SIP
   entities (typically intermediaries) to carry the service information
   of the user sending a SIP message.

   The P-Asserted-Service header field carries information that is
   derived service identification. While a declarative service
   identification can assist in deriving the value transferred in this
   header, this should be in the form of streamlining the correct
   derived service identification.

```
        PAssertedService = "P-Asserted-Service"
                           HCOLON PAssertedService-value
        PAssertedService-value = Service-ID *(COMMA Service-ID)
```

   See Section 4.4 for the definition of Service-ID in ABNF.

   Proxies can (and will) add and remove this header field.

   Table 1 extends the headers defined in this document to Table 2 in
   SIP [RFC3261], Section 7.1 of the SIP-specific event notification
   [RFC3265], tables 1 and 2 in the SIP INFO method [RFC2976], tables 1
   and 2 in Reliability of provisional responses in SIP [RFC3262],
   tables 1 and 2 in the SIP UPDATE method [RFC3311], tables 1 and 2 in
   the SIP extension for Instant Messaging [RFC3428], table 1 in the SIP
   REFER method [RFC3515] and tables 2 and 3 in the SIP PUBLISH method
   [RFC3903]:

Header field	where	proxy	ACK	BYE	CAN	INV	OPT	REG	SUB
P-Asserted-Service	R	admr	-	-	-	o	o	-	o

Header field	NOT	PRA	INF	UPD	MSG	REF	PUB
P-Asserted-Service	-	-	-	-	o	o	o

                            Table 1

   There may be multiple P-Asserted-Service header fields. The
   semantics of multiple P-Asserted-Service header fields appearing in
   the same request is not defined.

4.2. The P-Preferred-Service Header

The P-Preferred-Service header field is used by a user agent sending
the SIP request to provide a hint to a trusted proxy of the preferred
service that the user wishes to be used for the P-Asserted-Service
field value that the trusted element will insert.

The P-Preferred-Service header field carries information that is
declarative service identification. Such information should only be
used to assist in deriving a derived service identification at the
recipient entity.

```
    PPreferredService = "P-Preferred-Service"
                            HCOLON PPreferredService-value
    PPreferredService-value = Service-ID *(COMMA Service-ID)
```

See Section 4.4 for the definition of Service-ID in ABNF.

Table 2 extends the headers defined in this document to Table 2 in
SIP [RFC3261], Section 7.1 of the SIP-specific event notification
[RFC3265], tables 1 and 2 in the SIP INFO method [RFC2976], tables 1
and 2 in Reliability of provisional responses in SIP [RFC3262],
tables 1 and 2 in the SIP UPDATE method [RFC3311], tables 1 and 2 in
the SIP extension for Instant Messaging [RFC3428], table 1 in the SIP
REFER method [RFC3515] and tables 2 and 3 in the SIP PUBLISH method
[RFC3903]:

Header field	where	proxy	ACK	BYE	CAN	INV	OPT	REG	SUB
P-Preferred-Service	R	dr	-	-	-	o	o	-	o

Header field	NOT	PRA	INF	UPD	MSG	REF	PUB
P-Preferred-Service	-	-	-	-	o	o	o

Table 2

There may be multiple P-Preferred-Service header fields. The
semantics of multiple P-Preferred-Service header fields appearing in
the same request is not defined.

4.3. Service and Application Definition

Definition of services and their characteristics is outside the scope
of this document. Other standards organizations, vendors and
operators may define their own services and register them.

A hierarchical structure is defined consisting of service identifiers

or application identitifiers, subservice identifiers.

The service and subservice identifiers identify the service as
described in Section 1. The URN may also be used to identify a
service or an application between end users for use within the
context of RFC 3841 [RFC3841] and RFC 3840 [RFC3840].

IANA maintains a registry of service identifier values that have been
assigned. This registry is created by the actions of Section 8.2 of
this document.

Subservice identifiers are not managed by IANA. It is the
responsibility of the organisation that registered the service to
manage the subservices.

4.4. Registration Template

Below, we include the registration template for the URN scheme
according to RFC 3406 [RFC3406]. The URN scheme is defined as an
informal NID.

Namespace ID: urn-7

Registration Information: Registration version: 1; registration
    date: 2009-03-22

Declared registrant of the namespace: TBD

Declaration of syntactic structure: The URN consists of a
    hierarchical service identifier or application identifier, with a
    sequence of labels separated by periods. The left-most label is
    the most significant one and is called 'top-level service
    identifier', while names to the right are called 'sub-services' or
    'sub-applications'. The set of allowable characters is the same
    as that for domain names (see RFC 1123 [RFC1123]) and a subset of
    the labels allowed in RFC 3958 [RFC3958]. Labels are case-
    insensitive and MUST be specified in all lower-case. For any
    given service identifier, labels can be removed right-to-left and
    the resulting URN is still valid, referring a more generic
    service, with the exception of the top-level service identifier
    and possibly the first sub-service or sub-application identifier.
    Labels cannot be removed beyond a defined basic service, for
    example, the label w.x may define a service, but the label w may
    only define an assignment authority for assigning subsequent
    values and not define a service in its own right. In other words,
    if a service identifier 'w.x.y.z' exists, the URNs 'w.x' and
    'w.x.y' are also valid service identifiers, but w may not be a
    valid service identifier if it merely defines who is responsible

for defining x.

```
  Service-ID      = "urn:urn-7:" urn-service-id
  urn-service-id  = top-level *("." sub-service-id)
  top-level       = let-dig [ *26let-dig ]
  sub-service-id  = let-dig [ *let-dig ]
  let-dig         = ALPHA / DIGIT / "-"
```

While the naming convention above uses the term "service" all the
constructs are equally applicable to identifying applications
within the UA.

Note to RFC editor: the value above has been preassigned by IANA.

Relevant ancillary documentation: None

Identifier uniqueness considerations: A service identifier
   identifies a service, and an application identifier an application
   indicated in the service or application registration (see IANA
   Considerations (Section 8)). Uniqueness is guaranteed by the IANA
   registration.

Identifier persistence considerations: The service or application
   identifier for the same service is or application expected to be
   persistent, although there naturally cannot be a guarantee that a
   particular service will continue to be available globally or at
   all times.

Process of identifier assignment: The process of identifier
   assignment is described in the IANA Considerations (Section 8).

Process for identifier resolution: There is no single global
   resolution service for service identifiers or application
   identifiers.

Rules for Lexical Equivalence: 'service' identifiers are compared
   according to case-insensitive string equality.

Conformance with URN Syntax: The BNF in the 'Declaration of
   syntactic structure' above constrains the syntax for this URN
   scheme.

Validation mechanism: Validation determines whether a given string
   is currently a validly-assigned URN (see RFC 3406 [RFC3406]). Due
   to the distributed nature of usage and since not all services are
   available everywhere, validation in this sense is not possible

   Scope: The scope for this URN can be local to a single domain, or
      may be more widely used.

5. **Usage of the P-Preferred-Service and P-Asserted-Service header fields**

5.1. **Usage of the P-Preferred-Service and P-Asserted-Service header fields in Requests**

5.1.1. **Procedures at User Agent Clients (UAC)**

The UAC MAY insert a P-Preferred-Service in a request that creates a dialog, or a request outside of a dialog. This information can assist the proxies in identifying appropriate service capabilities to apply to the call. This information MUST NOT conflict with other SIP or SDP information included in the request. Furthermore, the SIP or SDP information needed to signal functionality of this service MUST be present. Thus if a service requires a video component, then the SDP has to include the media line associated with that video component; it cannot be assumed from the P-Preferred-Service header field value. Similarly if the service requires particular SIP functionality for which a SIP extension and a Require header field value is defined, then the request has to include that SIP signalling as well as the P-Preferred-Service header field value.

A UAC that is within the same trust domain as the proxy it sends a request to, (e.g a media gateway or application server) MAY insert a P-Asserted-Service header field in a request that creates a dialog, or a request outside of a dialog. This information MUST NOT conflict with other SIP or SDP information included in the request. Furthermore, the SIP or SDP information needed to signal functionality of this service MUST be present.

5.1.2. **Procedures at Intermediate Proxies**

A proxy in a Trust Domain can receive a request from a node that it trusts, or a node that it does not trust. When a proxy receives a request from a node it does not trust and it wishes to add a P-Asserted-Service header field, the proxy MUST identify the service appropriate to the capabilities (e.g. SDP) in the request, MAY authenticate the originator of the request (in order to determine whether the user is subscribed for that service), and use the identity which results from this checking and authentication to insert a P-Asserted-Service header field into the request.

When a proxy receives a request containing a P-Preferred-Service header field the Proxy MAY use the contents of that header field to assist in determining the service to be included in a P-Asserted-Service header field, (for instance to prioritize the order of comparison of filter criteria for potential services that the request could match). The proxy MUST NOT use the contents of the

P-Preferred-Service header field to identify the service without
first checking against the capabilities (e.g. SDP) contained in the
request. If the proxy inserts a P-Asserted-Service header field in
the request the proxy MUST remove the P-Preferred-Service header
field before forwarding the request, otherwise the Proxy SHOULD
include the P-Preferred-Service header field when forwarding the
request.

If the proxy receives a request from a node that it trusts, it can
use the information in the P-Asserted-Service header field, if any,
as if it had authenticated the user itself.

If there is no P-Asserted-Service header field present, or it is not
possible to match the request to a specific service as identified by
the service identifier, a proxy MAY add one containing it using its
own analysis of the information contained in the SIP request. If the
proxy received the request from an element that it does not trust and
there is a P-Asserted-Service header present, the proxy MUST replace
that header field contents with a new analysis or remove this header
field.

The analysis performed to identify such service identifiers is
outside the scope of this document. However, it is perfectly valid
as a result of the analysis to not include any service identifier in
the forwarded required, and thus not include a P-Asserted-Service
header.

If a proxy forwards a request to a node outside the proxy's trust
domain, there MUST NOT be a P-Asserted-Service header field in the
forwarded request.

5.1.3. Procedures at User Agent Servers (UAS)

For a UAS outside the trust domain, the P-Asserted-Service header is
removed before it reaches this entity, therefore there are no
procedures for such a device.

However, if a User Agent Server receives a request from a previous
element that it does not trust, it MUST NOT use the P-Asserted-
Service header field in any way.

If a UA is part of the Trust Domain from which it received a request
containing a P-Asserted-Service header field, then it can use the
value freely but it MUST ensure that it does not forward the
information to any element that is not part of the Trust Domain.

5.2. Usage of the P-Preferred-Service and P-Asserted-Service header fields in Responses

There is no usage of these header field in responses.

6. **Examples of Usage**

   In this example, proxy.example.com creates a P-Asserted-Service
   header field from the user identity it discovered from SIP Digest
   authentication, and the list of services appropriate to that user,
   and the services that correspond to the SDP information included in
   the request. Note that F1 and F2 are about identifying the user and
   do not directly form part of the capability provided in this
   document. It forwards this information to a trusted proxy which
   forwards it to a trusted gateway. Note that these examples consist
   of partial SIP messages that illustrate only those headers relevant
   to the authenticated identity problem.

      * F1   useragent.example.com -> proxy.example.com

      INVITE sip:+14085551212@example.com SIP/2.0
      Via: SIP/2.0/TCP useragent.example.com;branch=z9hG4bK-123
      To: <sip:+14085551212@example.com>
      From: "Anonymous" <sip:anonymous@anonymous.invalid>;tag=9802748
      Call-ID: 245780247857024504
      CSeq: 1 INVITE
      Max-Forwards: 70

      v=0
      o=- 2987933615 2987933615 IN IP6 5555::aaa:bbb:ccc:ddd
      s=-
      c=IN IP6 5555::aaa:bbb:ccc:ddd
      t=0 0
      m=audio 3456 RTP/AVPF 97 96
      b=AS:25.4
      a=curr:qos local sendrecv
      a=curr:qos remote none
      a=des:qos mandatory local sendrecv
      a=des:qos mandatory remote sendrecv
      a=sendrecv
      a=rtpmap:97 AMR
      a=fmtp:97 mode-set=0,2,5,7; maxframes

      * F2   proxy.example.com -> useragent.example.com

      SIP/2.0 407 Proxy Authorization
      Via: SIP/2.0/TCP useragent.example.com;branch=z9hG4bK-123
      To: <sip:+14085551212@example.com>;tag=123456
      From: "Anonymous" <sip:anonymous@anonymous.invalid>;tag=9802748
      Call-ID: 245780247857024504
      CSeq: 1 INVITE
      Proxy-Authenticate: .... realm="sip.example.com"

```
* F3   useragent.example.com -> proxy.example.com

INVITE sip:+14085551212@example.com SIP/2.0
Via: SIP/2.0/TCP useragent.example.com;branch=z9hG4bK-124
To: <sip:+14085551212@example.com>
From: "Anonymous" <sip:anonymous@anonymous.invalid>;tag=9802748
Call-ID: 245780247857024504
CSeq: 2 INVITE
Max-Forwards: 70
Proxy-Authorization: realm="sip.example.com" user="fluffy"

v=0
o=- 2987933615 2987933615 IN IP6 5555::aaa:bbb:ccc:ddd
s=-
c=IN IP6 5555::aaa:bbb:ccc:ddd
t=0 0
m=audio 3456 RTP/AVPF 97 96
b=AS:25.4
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=sendrecv
a=rtpmap:97 AMR
a=fmtp:97 mode-set=0,2,5,7; maxframes
```

```
* F4   proxy.example.com -> proxy.pstn.example (trusted)

INVITE sip:+14085551212@proxy. pstn.example SIP/2.0
Via: SIP/2.0/TCP useragent.example.com;branch=z9hG4bK-124
Via: SIP/2.0/TCP proxy.example.com;branch=z9hG4bK-abc
To: <sip:+14085551212@example.com>
From: "Anonymous" <sip:anonymous@anonymous.invalid>;tag=9802748
Call-ID: 245780247857024504
CSeq: 2 INVITE
Max-Forwards: 69
P-Asserted-Service: urn:urn-7:3gpp-service.exampletelephony.version1

v=0
o=- 2987933615 2987933615 IN IP6 5555::aaa:bbb:ccc:ddd
s=-
c=IN IP6 5555::aaa:bbb:ccc:ddd
t=0 0
m=audio 3456 RTP/AVPF 97 96
b=AS:25.4
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=sendrecv
a=rtpmap:97 AMR
a=fmtp:97 mode-set=0,2,5,7; maxframes
```

```
     * F5   proxy.pstn.example -> gw.pstn.example (trusted)

     INVITE sip:+14085551212@gw.pstn.example SIP/2.0
     Via: SIP/2.0/TCP useragent.example.com;branch=z9hG4bK-124
     Via: SIP/2.0/TCP proxy.example.com;branch=z9hG4bK-abc
     Via: SIP/2.0/TCP proxy.pstn.example;branch=z9hG4bK-a1b2
     To: <sip:+14085551212@example.com>
     From: "Anonymous" <sip:anonymous@anonymous.invalid>;tag=9802748
     Call-ID: 245780247857024504
     CSeq: 2 INVITE
     Max-Forwards: 68
     P-Asserted-Service: urn:urn-7:3gpp-service.exampletelephony.version1

     v=0
     o=- 2987933615 2987933615 IN IP6 5555::aaa:bbb:ccc:ddd
     s=-
     c=IN IP6 5555::aaa:bbb:ccc:ddd
     t=0 0
     m=audio 3456 RTP/AVPF 97 96
     b=AS:25.4
     a=curr:qos local sendrecv
     a=curr:qos remote none
     a=des:qos mandatory local sendrecv
     a=des:qos mandatory remote sendrecv
     a=sendrecv
     a=rtpmap:97 AMR
     a=fmtp:97 mode-set=0,2,5,7; maxframes
```

7. Security considerations

The mechanism provided in this document is a partial consideration of the problem of service identification in SIP. For example, these mechanisms provide no means by which end users can securely share service information end-to-end without a trusted service provider. This information is secured by transitive trust, which is only as reliable as the weakest link in the chain of trust.

The trust domain provides a set of servers where the characteristics of the service are agreed for that service identifier value, and that the calling user is entitled to use that service. draft-ietf-sipping-service-identification [I-D.ietf-sipping-service-identification] identifies the impact of allowing such service identifier values to "leak" outside of the trust domain, including implications on fraud, interoperability and stifling of service innovation.

8. IANA considerations

8.1. P-Asserted-Service and P-Preferred-Service header fields

This document specifies two new SIP headers: P-Asserted-Service and
P-Preferred-Service. Their syntax is given in Section 3. These
headers are defined by the following information, which has been
added to the header sub-registry under
http://www.iana.org/assignments/sip-parameters.

```
     Header Name          compact    Reference
     ----------------     -------    ---------
     P-Asserted-Service              [RFCxxxx]
     P-Preferred-Service             [RFCxxxx]
```

Note to the RFC editor: substitute xxxx with the RFC number of this
document.

8.2. Definition of Service-ID values

top-level identifiers are identified by labels managed by IANA,
according to the processes outlined in RFC 5226 [RFC5226] in a new
registry called "Service-ID/Application-ID Labels". Thus, creating a
new service at the top-level requires IANA action. The policy for
adding service labels is 'specification required'. The following two
identifiers are initially defined:

3gpp-service

3gpp-application

Subservice identifiers are not managed by IANA. It is the
responsibility of the organisation that registered the service to
manage the subservices.

Application identifiers are not managed by IANA. It is the
responsibility of the organisation that registered the service to
manage the applicable applications.

Entries in the registration table have the following format:

```
   Service/Application   Reference  Description
   ----------------------------------------------------------------------
   3gpp-service          RFCxxxx    Communication services defined by
                                    3GPP for use by the IM CN subsystem
                                    and its attached UAs. This value
                                    in itself does not define a service
                                    and requires subsequent labels to
                                    define the service.

   3gpp-application      RFCxxxx    Applications defined by 3GPP for
                                    use by UAs attached to the IM CN
                                    subsystem. This value in itself
                                    does not define a service and
                                    requires subsequent labels to define
                                    the service.
```

Note to the RFC editor: substitute xxxx with the RFC number of this
document.

9. **APPENDIX: Changes history**

   Note to RFC Editor: Please remove this entire appendix before
   publication

9.1. **Changes between version -01 and version -02**

   1. Incorporation of terms "derived service identification" and
       "declarative service identification" from
       draft-ietf-sipping-service-identification.

   2. Correction of the URN syntax in examples.

   3. Appropriate introduction to table 1 and table 2 placing these in
       a normative context to those in other tables in other RFCs.

   4. Addition to security considerations section to clarify trust
       domain concept.

   5. References to RFC 3325 changed to RFC 3324 for definition of
       trust domain.

   6. Reference to RFC 2234 updated to RFC 5234 because the later
       revision applies. No consequential technical change.

   7. Reference to RFC 2434 updated to RFC 5226 because the later
       revision applies. No consequential technical change.

   8. References updated to symbolic. Remove of reference identifiers
       from abstract.

   9. Numerous editorial changes and minor clarifications.

9.2. **Changes between version -02 and version -03**

   1. The URN value has been preassigned by IANA. This valye
       substituted into document.

   2. service-id is extended to include "urn:" within the expansion to
       conform to usage.

   3. Procedures inserted where the UAS is inside the trust domain,
       e.g. gateway.

   4. Procedures inserted for the proxy handling of a P-Preferred-
       Service header field.

   5. A number of editorial corrections.

10. References

10.1. Normative References

[RFC1123]   Braden, R., "Requirements for Internet Hosts - Application
            and Support", STD 3, RFC 1123, October 1989.

[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.

[RFC3406]   Daigle, L., van Gulik, D., Iannella, R., and P. Faltstrom,
            "Uniform Resource Names (URN) Namespace Definition
            Mechanisms", BCP 66, RFC 3406, October 2002.

[RFC3958]   Daigle, L. and A. Newton, "Domain-Based Application
            Service Location Using SRV RRs and the Dynamic Delegation
            Discovery Service (DDDS)", RFC 3958, January 2005.

[RFC5226]   Narten, T. and H. Alvestrand, "Guidelines for Writing an
            IANA Considerations Section in RFCs", BCP 26, RFC 5226,
            May 2008.

[RFC5234]   Crocker, D. and P. Overell, "Augmented BNF for Syntax
            Specifications: ABNF", STD 68, RFC 5234, January 2008.

10.2. Informative References

[RFC2976]   Donovan, S., "The SIP INFO Method", RFC 2976,
            October 2000.

[RFC3262]   Rosenberg, J. and H. Schulzrinne, "Reliability of
            Provisional Responses in Session Initiation Protocol
            (SIP)", RFC 3262, June 2002.

[RFC3265]   Roach, A., "Session Initiation Protocol (SIP)-Specific
            Event Notification", RFC 3265, June 2002.

[RFC3311]   Rosenberg, J., "The Session Initiation Protocol (SIP)
            UPDATE Method", RFC 3311, October 2002.

   [RFC3428] Campbell, B., Rosenberg, J., Schulzrinne, H., Huitema, C.,
              and D. Gurle, "Session Initiation Protocol (SIP) Extension
              for Instant Messaging", RFC 3428, December 2002.

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

   [RFC3840] Rosenberg, J., Schulzrinne, H., and P. Kyzivat,
              "Indicating User Agent Capabilities in the Session
              Initiation Protocol (SIP)", RFC 3840, August 2004.

   [RFC3841] Rosenberg, J., Schulzrinne, H., and P. Kyzivat, "Caller
              Preferences for the Session Initiation Protocol (SIP)",
              RFC 3841, August 2004.

   [RFC3903] Niemi, A., "Session Initiation Protocol (SIP) Extension
              for Event State Publication", RFC 3903, October 2004.

   [I-D.ietf-sipping-service-identification]
              Rosenberg, J., "Identification of Communications Services
              in the Session Initiation Protocol (SIP)",
              draft-ietf-sipping-service-identification-03 (work in
              progress), August 2008.

   [I-D.rosenberg-sip-app-media-tag]
              Rosenberg, J., "A Session Initiation Protocol (SIP) Media
              Feature Tag for MIME Application Sub-Types",
              draft-rosenberg-sip-app-media-tag-02 (work in progress),
              November 2007.

Author's Address

    Keith Drage
    Alcatel-Lucent
    Quadrant, Stonehill Green, Westlea
    Swindon, Wilts
    UK

    Email: drage@alcatel-lucent.com