

MARTINI WG	A. Roach	
Internet-Draft	Tekelec	
Intended status: Standards Track	April 20, 2010	
Expires: October 22, 2010		

[TOC](#)

## **Registration for Multiple Phone Numbers in the Session Initiation Protocol (SIP)**

**draft-ietf-martini-gin-01**

### **Abstract**

This document defines a mechanism by which a SIP server acting as a traditional Private Branch Exchange (PBX) can register with a SIP Service Provider (SSP) to receive phone calls for terminals designated by phone numbers. In order to function properly, this mechanism relies on the fact that the phone numbers are fully qualified and globally unique.

### **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 October 22, 2010.

### **Copyright Notice**

Copyright (c) 2010 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.](#) Introduction
- [2.](#) Constraints
- [3.](#) Terminology
- [4.](#) Mechanism Overview
- [5.](#) Registering for Multiple Phone Numbers
- [6.](#) SSP Processing of Inbound Phone Number Requests
- [7.](#) Interaction with Other Mechanisms
  - [7.1.](#) Globally Routable User-Agent URIs (GRUU)
    - [7.1.1.](#) Public GRUUs
    - [7.1.2.](#) Temporary GRUUs
  - [7.2.](#) Registration Event Package
    - [7.2.1.](#) PBX Aggregate Registration State
    - [7.2.2.](#) Individual Phone Number Registration State
  - [7.3.](#) Client-Initiated (Outbound) Connections
  - [7.4.](#) Non-Adjacent Contact Registration (Path) and Service Route

Discovery

- [8.](#) Examples
- [8.1.](#) Usage Scenario: Basic Registration
- [8.2.](#) Usage Scenario: Using Path to Control Request URI
- [9.](#) Requirements Analysis
- [10.](#) IANA Considerations
- [10.1.](#) New SIP Option Tag
- [10.2.](#) New SIP URI Parameters
  - [10.2.1.](#) 'bnc' SIP URI paramter
  - [10.2.2.](#) 'sg' SIP URI paramter
- [11.](#) Security Considerations
- [12.](#) Acknowledgements
- [13.](#) References
- [13.1.](#) Normative References
- [13.2.](#) Informative References
- [S](#) Author's Address

---

## 1. Introduction

[TOC](#)

One of SIP's primary functions is providing rendezvous between users. By design, this rendezvous has been provided through a combination of the server look-up procedures defined in RFC 3263 [\[3\]](#) ([Rosenberg, J. and H. Schulzrinne, "Session Initiation Protocol \(SIP\): Locating SIP Servers," June 2002.](#)), and the registrar procedures described in RFC 3261 [\[2\]](#) ([Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.](#)).

The intention of the original protocol design was that any user's AOR would be handled by the authority indicated by the hostport portion of the AOR. The users registered individual reachability information with this authority, which would then route incoming requests accordingly. In actual deployments, some SIP servers have been deployed in architectures that, for various reasons, have requirements to provide dynamic routing information for large blocks of AORs, where all of the AORs in the block were to be handled by the same server. For purposes of efficiency, many of these deployments do not wish to maintain separate registrations for each of the AORs in the block. This leads to the desire for an alternate mechanism for providing dynamic routing information for blocks of AORs.

Although the use of REGISTER to update reachability information for multiple users simultaneously is somewhat beyond the original semantics defined for REGISTER, this approach has seen significant deployment in certain environments. In particular, deployments in which small to medium SIP PBX servers are addressed using E.164 numbers have used this mechanism to avoid the need to maintain DNS entries or static IP addresses for the PBX servers.

In recognition of the momentum that REGISTER-based approaches have seen in deployments, this document defines a REGISTER-based approach that is tailored to E.164-addressed terminals in a SIP PBX environment. It does not address registration of SIP URIs in which the user portion is not an E.164 number.

---

## 2. Constraints

[TOC](#)

The following paragraph is perhaps the most important in understanding the solution defined in this document.

Within the problem space that has been established for this work, several constraints shape our solution. These are being defined in the MARTINI requirements document [\[5\] \(Elwell, J. and H. Kaplan, "Requirements for multiple address of record \(AOR\) reachability information in the Session Initiation Protocol \(SIP\)," March 2010.\)](#). In terms of impact to the solution at hand, the following two constraints have the most profound effect: (1) The PBX cannot be assumed to be assigned a static IP address; and (2) No DNS entry can be relied upon to consistently resolve to the IP address of the PBX.

---

## 3. Terminology

[TOC](#)

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 [\[1\] \(Bradner,](#)

[S., "Key words for use in RFCs to Indicate Requirement Levels," March 1997. \).](#)

Further, the term "SSP" is meant as an acronym for a "SIP Service Provider," while the term "PBX" is used to indicate a SIP Private Branch Exchange.

---

#### 4. Mechanism Overview

[TOC](#)

The overall mechanism is achieved using a REGISTER request with a specially-formatted Contact URI. This document also defines an option tag that can be used to ensure a registrar and any intermediaries understand the mechanism described herein.

The Contact URI itself is tagged with a URI parameter to indicate that it actually represents a multitude of phone-number-associated contacts. We also define some lightweight extensions for Globally Routable UA URIs (GRUU) to allow the use of public and temporary GRUUs assigned by the SSP.

Aside from these extensions, the REGISTER message itself is processed by a registrar in the same way as normal registrations: by updating its location service with additional AOR to Contact bindings.

Note that the list of phone numbers associated with a PBX is a matter of local provisioning at the SSP and at the PBX. The mechanism defined in this document does not provide any means to detect or recover from provisioning mismatches (although the registration event package can be used as a standardized means for auditing such phone numbers; see [Section 7.2.1 \(PBX Aggregate Registration State\)](#)).

---

#### 5. Registering for Multiple Phone Numbers

[TOC](#)

To register for multiple phone numbers, the PBX sends a REGISTER message to the SSP. This REGISTER varies from a typical register in two important ways. First, it must contain an option tag of "gin" in both a "Require" header field and a "Proxy-Require" header field. (The option tag "gin" is an acronym for "generate implicit numbers".) Second, in at least one "Contact" header field, it must include a Contact URI that contains the URI parameter "bnc", and no user portion (hence no "@" symbol). A URI with a "bnc" parameter MUST NOT contain a user portion. Because of the constraints discussed in [Section 2 \(Constraints\)](#), the host portion of the Contact URI will generally contain an IP address, although nothing in this mechanism enforces or relies upon that fact. If the PBX operator chooses to maintain DNS entries that resolve to the IP address of his PBX via RFC 3263 resolution procedures, then this mechanism works just fine with domain names in the Contact header field.

The URI parameter indicates that special interpretation of the Contact URI is necessary: instead of representing a single, concrete Contact URI to be inserted into the location service, it represents a multitude of Contact URIs (one for each associated phone number), semantically resulting in a multitude of AOR-to-Contact rows in the location service.

The registrar, upon receipt of a REGISTER message in the foregoing form, will use the value in the "To" header field to identify the PBX for which registration is being requested. It then authenticates the PBX (using, e.g., SIP Digest authentication, mutual TLS, or some other authentication mechanism). After the PBX is authenticated, the registrar updates its location service with a unique AOR to Contact mapping for each of the phone numbers associated with the PBX. Semantically, each of these mappings will be treated as a unique row in the location service. The actual implementation may, of course, perform internal optimizations to reduce the amount of memory used to store such information.

For each of these unique rows, the AOR will be in the format that the SSP expects to receive from external parties (e.g. "sip: +12145550102@ssp.example.com"), and the corresponding Contact will be formed by adding to the REGISTER's Contact URI a user portion containing the fully-qualified, E.164-formatted phone number (including the preceding "+" symbol) and removing the "bnc" parameter. Aside from the initial "+" symbol, this E.164-formatted number MUST consist exclusively of digits from 0 through 9, and explicitly MUST NOT contain any visual separator symbols (e.g., "-", ".", "(", or ")"). For example, if the "Contact" header field contains the URI <sip: 198.51.100.3:5060;user=phone;bnc>, then the Contact value associated with the aforementioned AOR will be <sip: +12145550102@198.51.100.3:5060;user=phone>.

Aside from the "bnc" parameter, all URI parameters present on the "Contact" URI in the REGISTER message MUST be copied to the Contact value stored in the location service.

Although the SSP treats this registration as a number of discrete rows for the purpose of re-targeting incoming requests, the renewal, expiration, and removal of these rows is bound to the registered "bnc" contact. In particular, this means that REGISTER requests that attempt to de-register a single phone number that has been implicitly registered MUST NOT remove that phone number from the bulk registration. A further implication of this property is that an individual extension that is implicitly registered may also be explicitly registered using a normal, non-bulk registration (subject to SSP policy). If such a registration exists, it is refreshed independently of the bulk registration, and is not removed when the bulk registration is removed.

Although it is not a special case from a protocol perspective, implementors are reminded that this behavior includes any "user" parameter on the URI. For example, if the registering PBX requires incoming messages to include a "user=phone" parameter on their Request-

URI, it will include "user=phone" on the URI in the "Contact" header field of its registration. Because of the requirement on the SSP to copy parameters into the location service, it will appear on any Request-URIs that are re-targeted to the PBX based on that registration. Conversely, if the registering PBX does not include a "user=phone" parameter, any SSP using the procedures defined in this document would not be at liberty to include one on the Request-URI. Any PBX implementing the registration mechanism defined in this document MUST also support the Path mechanism defined by RFC 3327 [\[6\] \(Willis, D. and B. Hoeneisen, "Session Initiation Protocol \(SIP\) Extension Header Field for Registering Non-Adjacent Contacts," December 2002.\)](#), and MUST include a 'path' option-tag in the Supported header field of the REGISTER request (which is a stronger requirement than imposed by the Path mechanism itself). This behavior is necessary because proxies between the PBX and the Registrar may need to insert Path header field values in the REGISTER request for this document's mechanism to function properly, and per RFC 3327 [\[6\] \(Willis, D. and B. Hoeneisen, "Session Initiation Protocol \(SIP\) Extension Header Field for Registering Non-Adjacent Contacts," December 2002.\)](#), they can only do so if the UAC inserted the option-tag in the Supported header field. In accordance with the procedures defined in RFC 3327 [\[6\] \(Willis, D. and B. Hoeneisen, "Session Initiation Protocol \(SIP\) Extension Header Field for Registering Non-Adjacent Contacts," December 2002.\)](#), the PBX is allowed to ignore the Path header fields returned in the REGISTER response.

A Registrar compliant with this document MUST support the Path mechanism defined in RFC 3327 [\[6\] \(Willis, D. and B. Hoeneisen, "Session Initiation Protocol \(SIP\) Extension Header Field for Registering Non-Adjacent Contacts," December 2002.\)](#).

---

## 6. SSP Processing of Inbound Phone Number Requests

[TOC](#)

In general, after processing the AOR to Contact mapping described in the preceding section, the SSP Proxy/Registrar (or equivalent entity) performs traditional Proxy/Registrar behavior, based on the mapping. For any inbound SIP requests whose AOR indicates an E.164 number assigned to one of the SSP's customers, this will generally involve setting the target set to the registered contacts associated with that AOR, and performing request forwarding as described in section 16.6 of RFC 3261 [\[2\] \(Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.\)](#). An SSP using the mechanism defined in this document MUST perform such processing for inbound INVITE requests and SUBSCRIBE requests to the "reg" event package (see [Section 7.2.2 \(Individual Phone Number Registration State\)](#)), and SHOULD

perform such processing for all other method types, including unrecognized SIP methods.

---

## 7. Interaction with Other Mechanisms

[TOC](#)

The following sections describe the means by which this mechanism interacts with relevant REGISTER-related extensions currently defined by the IETF.

Currently, the descriptions are somewhat informal, and omit some details for the sake of brevity. If the MARTINI working group expresses interest in furthering the mechanism described by this document, they will be fleshed out with more detail and formality.

---

### 7.1. Globally Routable User-Agent URIs (GRUU)

[TOC](#)

To enable advanced services to work with terminals behind a SIP PBX, it is important that the GRUU mechanism defined by RFC 5627 [\[10\]](#) ([Rosenberg, J., "Obtaining and Using Globally Routable User Agent URIs \(GRUUs\) in the Session Initiation Protocol \(SIP\)," October 2009.](#)) work correctly with the mechanism defined by this document -- that is, that User Agents services by the PBX can acquire and use GRUUs for their own use.

---

#### 7.1.1. Public GRUUs

[TOC](#)

When a PBX registers a Bulk Number Contact (a Contact with a "bnc" parameter), and also invokes GRUU procedures for that Contact during registration, then the SSP will assign a public GRUU to the PBX in the normal fashion. Because the URI being registered contains a "bnc" parameter, the GRUU will also contain a "bnc" parameter. In particular, this means that the GRUU will not contain a user portion.

When a terminal registers with the PBX using GRUU procedures for a Contact, the PBX adds an "sg" parameter to the GRUU parameter it received from the SSP. This "sg" parameter contains a disambiguation token that the PBX can use to route the request to the proper user agent.

So, for example, when the PBX registers with the following contact header field:

```
Contact: <sip:198.51.100.3;user=phone;bnc>;  
+sip.instance="<urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6>"
```

Then the SSP may choose to respond with a Contact header field that looks like this:

```
<allOneLine>  
Contact: <sip:198.51.100.3;user=phone;bnc>;  
pub-gruu="sip:ssp.example.com;gr=urn:  
uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6";  
+sip.instance="<urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6>"  
;expires=7200  
</allOneLine>
```

When its own terminals register, the PBX can then add whatever device identifier it feels appropriate in an "sg" parameter, and present this value to its own terminals. For example, assume the terminal associated with the phone number "+12145550102" sent the following Contact header field in its register:

```
Contact: <sip:line-1@10.20.1.17>;  
+sip.instance="<urn:uuid:d0e2f290-104b-11df-8a39-0800200c9a66>"
```

The PBX will add an "sg" parameter to the pub-gruu it received from the SSP with a token that uniquely identifies the device (possibly the URN itself; possibly some other identifier); insert a user portion containing the fully-qualified E.164 number associated with the terminal; and return the result to the terminal as its public GRUU. The resulting Contact header field would look something like this:

```
<allOneLine>  
Contact: <sip:line-1@10.20.1.17>;  
pub-gruu="sip:+12145550102@ssp.example.com;gr=urn:  
uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6;sg=00:05:03:5e:70:a6";  
+sip.instance="<urn:uuid:d0e2f290-104b-11df-8a39-0800200c9a66>"  
;expires=3600  
</allOneLine>
```

When an incoming request arrives at the SSP for a GRUU corresponding to a bulk number contact ("bnc"), the SSP performs slightly different processing for the GRUU than a Proxy/Registrar would. When the GRUU is re-targeted to the registered bulk number contact, the SSP MUST copy the "sg" parameter from the GRUU to the new target. The PBX can then use this "sg" parameter to determine which user agent the request should be routed to. For example, the first line of an INVITE request that has been re-targeted to the PBX for the terminal shown above would look like this:



```
<allOneLine>  
INVITE sip:+12145550102@198.51.100.3;gr=urn:uuid:f81d4fae-7dec-  
11d0-a765-00a0c91e6bf6;sg=00:05:03:5e:70:a6 SIP/2.0  
</allOneLine>
```

---

### 7.1.2. Temporary GRUUs

[TOC](#)

PBXes have two options for creating temporary GRUUs for use by its terminals.

---

#### 7.1.2.1. Approach 1 - Self Made GRUUs

[TOC](#)

If a PBX wishes to provide temporary GRUUs for its terminals, it may do so by producing its own "Self-made GRUUs" (as defined in section 4.3 of RFC 5627 [\[10\]](#) ([Rosenberg, J., "Obtaining and Using Globally Routable User Agent URIs \(GRUUs\) in the Session Initiation Protocol \(SIP\)," October 2009.](#))). These GRUUs are produced using the PBX's own IP address (or domain, if it maintains one in DNS). The temporary GRUUs are then propagated to terminals using normal GRUU mechanism.

The ability to produce temporary GRUUs in this fashion is predicated on the conditions described in section 4.3 of RFC 5627. In particular, it requires PBX to be publicly routable, and willing to accept requests destined for its own Self-made GRUUs from sources other than the SSP. If these conditions cannot be satisfied (or the PBX operator chooses not to satisfy them for policy reasons), then the PBX users will not be able to make use of temporary GRUUs.

This mechanism is also predicated on the IP address for the PBX being relatively stable over a long period of time. This is generally a safe assumption to make, as frequent PBX IP address changes will result in intermittent connectivity issues and interruptions to ongoing calls. On a related note: when used with this extension, the SSP will not return a temporary GRUU in the registration response for any contacts that include a "bnc" parameter in their URI.

For example, using the same setup as in the "Public GRUU" section above, a terminal registering with the PBX might obtain a temp gruu by receiving a Contact header field that looks like:

```
<allOneLine>
Contact: <sip:line-1@10.20.1.17>;
pub-gruu="sip:ssp.example.com;gr=urn:uuid:f81d4fae-7dec-11d0-a765-
00a0c91e6bf6;sg=a0471c99573b877b";
+sip.instance="<urn:uuid:d0e2f290-104b-11df-8a39-0800200c9a66>"
;expires=3600
</allOneLine>
```

---

#### 7.1.2.2. Approach 2 - Anonymous Public GRUUs

[TOC](#)

If a PBX does not satisfy the criteria for producing its own "Self-made GRUUs," then it may create temporary GRUUs based on the public GRUUs it received from the SSP at registration time. To create Temporary GRUUs of this form, the PBX will add an opaque "sg" parameter to the public GRUU it received from the SSP, and will omit the user portion.

Note that, because these GRUUs are temporary GRUUs, a unique "sg" parameter will be generated for each successful registration attempt. The PBX tracks the various "sg" values associated with each user agent, and can re-target to the correct instance when the request arrives. For this approach to function, the SSP must be able to resolve a GRUU based solely on the value of its "gr" parameter, as the user portion of the GRUU will not contain an E.164 number. Further, the SSP will not know which actual phone number the request is destined for, only that it corresponds to a phone number belonging to the PBX.

Using the same basic setup as the example for the public GRUU, a terminal might receive a temporary GRUU by getting back a Contact header field that looks like this:

```
<allOneLine>
Contact: <sip:line-1@10.20.1.17>;
temp-gruu="sip:ssp.example.com;gr=urn:uuid:f81d4fae-7dec-11d0-a765-
00a0c91e6bf6;sg=0UYYRV046P";+sip.instance="<urn:uuid:d0e2f290-104b-
11df-8a39-0800200c9a66>";expires=3600
</allOneLine>
```

---

### 7.2. Registration Event Package

[TOC](#)

As this mechanism inherently deals with REGISTER behavior, it is imperative to consider its impact on the Registration Event Package defined by RFC 3680 [\[8\] \(Rosenberg, J., "A Session Initiation Protocol \(SIP\) Event Package for Registrations," March 2004.\)](#). In practice,

there will be two main use cases for subscribing to registration data: learning about the overall registration state for the PBX, and learning about the registration state for a single PBX phone number.

---

#### 7.2.1. PBX Aggregate Registration State

[TOC](#)

If the PBX (or another interested and authorized party) wishes to monitor or audit the registration state for all of the phone numbers currently registered to that PBX, it can subscribe to the SIP registration event package at the PBX's main URI -- that is, the URI used in the "To" header field of the REGISTER message.

The NOTIFY messages for such a subscription will contain a body that contains one record for each phone number associated with the PBX. The AORs will be in the format expected to be received by the SSP (e.g., "sip:+12145550105@ssp.example.com"), and the Contacts will correspond to the mapped Contact created by the registration (e.g., "sip:+12145550105@98.51.100.3").

In particular, the "bnc" parameter is forbidden from appearing in the body of a reg-event notify.

---

#### 7.2.2. Individual Phone Number Registration State

[TOC](#)

As described in [Section 6 \(SSP Processing of Inbound Phone Number Requests\)](#), the SSP will generally retarget all requests addressed to a phone number owned by a PBX to that PBX according to the mapping established at registration time. Although policy at the SSP may override this generally expected behavior, proper behavior of the registration event package requires that all "reg" event SUBSCRIBE requests are processed by the PBX. As a consequence, the requirements on an SSP for processing registration event package SUBSCRIBE requests are not left to policy.

If the SSP receives a SUBSCRIBE request for the registration event package with a Request-URI that indicates a contact registered via the "Bulk Number Contact" mechanism defined in this document, then it MUST proxy that SUBSCRIBE to the PBX in the same way that it would proxy an INVITE bound for that AOR.

Defining the behavior in this way is important, since the reg-event subscriber is interested in finding out about the comprehensive list of devices associated with the phone number. Only the PBX will have authoritative access to this information. For example, if the user has registered multiple terminals with differing capabilities, the SSP will not know about the devices or their capabilities. By contrast, the PBX will.

---

### 7.3. Client-Initiated (Outbound) Connections

[TOC](#)

RFC 5626 [\[9\] \(Jennings, C., Mahy, R., and F. Audet, "Managing Client-Initiated Connections in the Session Initiation Protocol \(SIP\)," October 2009.\)](#) -- needs analysis. Some people think it might "just work."

---

### 7.4. Non-Adjacent Contact Registration (Path) and Service Route Discovery

[TOC](#)

RFC 3327 [\[6\] \(Willis, D. and B. Hoeneisen, "Session Initiation Protocol \(SIP\) Extension Header Field for Registering Non-Adjacent Contacts," December 2002.\)](#) defines a means by which a registrar and its associated proxy can be informed of a route that is to be used between the proxy and the registered user agent. The scope of the route created by a "Path" header field is contact-specific; if an AOR has multiple contacts associated with it, the routes associated with each contact may be different from each other.

At registration time, any proxies between the user agent and the registrar may add themselves to the Path. By doing so, they request that any requests destined to the user agent as a result of the associated registration include them as part of the Route towards the User Agent. Although the Path mechanism does deliver the final Path value to the registering UA, UAs typically ignore the value of the Path.

To provide similar functionality in the opposite direction -- that is, to establish a route for requests sent by a registering UA -- RFC 3608 [\[7\] \(Willis, D. and B. Hoeneisen, "Session Initiation Protocol \(SIP\) Extension Header Field for Service Route Discovery During Registration," October 2003.\)](#) defines a means by which a UA can be informed of a route that is to be used by the UA to route all outbound requests associated with the AOR used in the registration. This information is scoped to the AOR within the UA, and is not specific to the Contact (or Contacts) in the REGISTER request.

The registrar unilaterally generates the values of the service route using whatever local policy it wishes to apply. Although it is common to use the Path and/or Route information in the request in composing the Service-Route, registrar behavior is not constrained in any way that requires it to do so.

In considering the interaction between these mechanisms and the registration of multiple AORs in a single request, implementors of proxies, registrars, and intermediaries must keep in mind the following issues, which stem from the fact that GIN effectively registers multiple AORs and multiple Contacts.

First, all location service records that result from expanding a single "bnc" Contact will necessarily share a single path. Proxies will be unable to make policy decisions on a contact-by-contact basis regarding whether to include themselves in the path. Second, and similarly, all AORs on the PBX that are registered with a common REGISTER message will be forced to share a common Service-Route.

One interesting technique that Path and Service-Route enable is the inclusion of a token or cookie in the user portion of the Service-Route or Path entries. This token or cookie may convey information to proxies about the identity, capabilities, and/or policies associated with the user. Since this information will be shared among several AORs and several Contacts when multiple AOR registration is employed, care should be taken to ensure that doing so is acceptable for all AORs and all Contacts registered in a single REGISTER message.

---

## 8. Examples

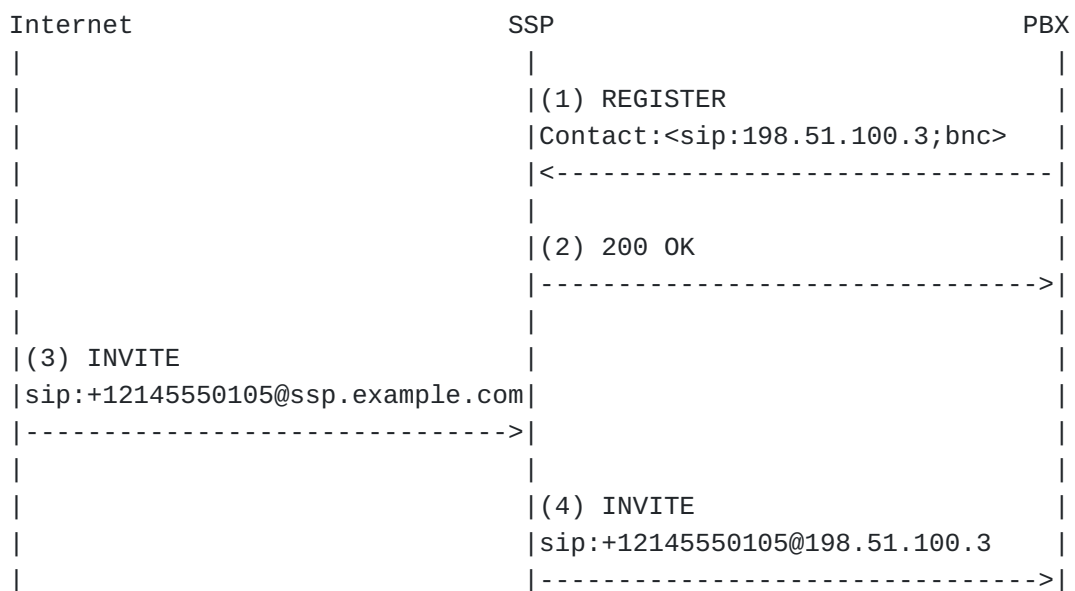
[TOC](#)

---

### 8.1. Usage Scenario: Basic Registration

[TOC](#)

This example shows the message flows for a basic bulk REGISTER transaction, followed by an INVITE addressed to one of the registered terminals. Example messages are shown after the sequence diagram.



(1) The PBX registers with the SSP for a range of phone numbers.

```
REGISTER sip:ssp.example.com SIP/2.0
Via: SIP/2.0/UDP 198.51.100.3:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
To: <sip:pbx@ssp.example.com>
From: <sip:pbx@ssp.example.com>;tag=a23589
Call-ID: 843817637684230@998sdasdh09
CSeq: 1826 REGISTER
Proxy-Require: gin
Require: gin
Supported: path
Contact: <sip:198.51.100.3:5060;user=phone;bnc>
Expires: 7200
Content-Length: 0
```

(3) The SSP receives a request for a phone number assigned to the PBX.

```
INVITE sip:+12145550105@ssp.example.com;user=phone SIP/2.0
Via: SIP/2.0/UDP foo.example;branch=z9hG4bKa0bc7a0131f0ad
Max-Forwards: 69
To: <sip:2145550105@some-other-place.example.net>
From: <sip:gsmith@example.org>;tag=456248
Call-ID: f7aecbfc374d557baf72d6352e1fbcd4
CSeq: 24762 INVITE
Contact: <sip:line-1@192.0.2.178:2081>
Content-Type: application/sdp
Content-Length: ...

<sdp body here>
```

- (4) The SSP retargets the incoming request according to the information received from the PBX at registration time.

```
INVITE sip:+12145550105@198.51.100.3;user=phone SIP/2.0
Via: SIP/2.0/UDP foo.example;branch=z9hG4bKa0bc7a0131f0ad
Via: SIP/2.0/UDP ssp.example.com;branch=z9hG4bKa45cd5c52a6dd50
Max-Forwards: 68
To: <sip:2145550105@some-other-place.example.net>
From: <sip:gsmith@example.org>;tag=456248
Call-ID: 7ca24b9679ffe9aff87036a105e30d9b
CSeq: 24762 INVITE
Contact: <sip:line-1@192.0.2.178:2081>
Content-Type: application/sdp
Content-Length: ...

<sdp body here>
```

---

## 8.2. Usage Scenario: Using Path to Control Request URI

[TOC](#)

This example shows a bulk REGISTER transaction with the SSP making use of the "Path" header field extension [\[6\] \(Willis, D. and B. Hoeneisen, "Session Initiation Protocol \(SIP\) Extension Header Field for Registering Non-Adjacent Contacts," December 2002.\)](#). This allows the SSP to designate a domain on the incoming Request URI that does not necessarily resolve to the PBX from when the SSP applies RFC 3263 procedures to it.

Internet	SSP	PBX
	(1) REGISTER	
	Path:<sip:pbx@198.51.100.3;lr>	
	Contact:<sip:pbx.example;bnc>	
	<----->	
	(2) 200 OK	
	----->	
(3) INVITE		
sip:+12145550105@ssp.example.com		
----->		
	(4) INVITE	
	sip:+12145550105@pbx.example	
	Route:<sip:pbx@198.51.100.3;lr>	
	----->	

- (1) The PBX registers with the SSP for a range of phone numbers. It includes the URI it expects to receive in the Request-URI in its "Contact" header field, and includes information that routes to the PBX in the "Path" header field.

```
REGISTER sip:ssp.example.com SIP/2.0
Via: SIP/2.0/UDP 198.51.100.3:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
To: <sip:pbx@ssp.example.com>
From: <sip:pbx@ssp.example.com>;tag=a23589
Call-ID: 843817637684230@998sdasdh09
CSeq: 1826 REGISTER
Proxy-Require: gin
Require: gin
Supported: path
Path: <sip:pbx@198.51.100.3:5060;lr>
Contact: <sip:pbx.example;user=phone;bnc>
Expires: 7200
Content-Length: 0
```



- (3) The SSP receives a request for a phone number assigned to the PBX.

```
INVITE sip:+12145550105@ssp.example.com;user=phone SIP/2.0
Via: SIP/2.0/UDP foo.example;branch=z9hG4bKa0bc7a0131f0ad
Max-Forwards: 69
To: <sip:2145550105@some-other-place.example.net>
From: <sip:gsmith@example.org>;tag=456248
Call-ID: f7aecbfc374d557baf72d6352e1fbcd4
CSeq: 24762 INVITE
Contact: <sip:line-1@192.0.2.178:2081>
Content-Type: application/sdp
Content-Length: ...
```

<sdp body here>

- (4) The SSP retargets the incoming request according to the information received from the PBX at registration time. Per the normal processing associated with "Path," it will insert the "Path" value indicated by the PBX at registration time in a "Route" header field, and set the request URI to the registered Contact.

```
INVITE sip:+12145550105@pbx.example.com;user=phone SIP/2.0
Via: SIP/2.0/UDP foo.example;branch=z9hG4bKa0bc7a0131f0ad
Via: SIP/2.0/UDP ssp.example.com;branch=z9hG4bKa45cd5c52a6dd50
Route: <sip:pbx@198.51.100.3:5060;lr>
Max-Forwards: 68
To: <sip:2145550105@some-other-place.example.net>
From: <sip:gsmith@example.org>;tag=456248
Call-ID: 7ca24b9679ffe9aff87036a105e30d9b
CSeq: 24762 INVITE
Contact: <sip:line-1@192.0.2.178:2081>
Content-Type: application/sdp
Content-Length: ...
```

<sdp body here>

---

## 9. Requirements Analysis

[TOC](#)

The document "Requirements for multiple address of record (AOR) reachability information in the Session Initiation Protocol (SIP)" [\[5\]](#) [\(Elwell, J. and H. Kaplan, "Requirements for multiple address of record \(AOR\) reachability information in the Session Initiation Protocol](#)

[\(SIP\)," March 2010.](#)) contains a list of requirements and desired properties for a mechanism to register multiple AORs with a single SIP transaction. This section evaluates those requirements against the mechanism described in this document.

REQ1 - The mechanism MUST allow a SIP-PBX to enter into a trunking arrangement with an SSP whereby the two parties have agreed on a set of telephone numbers deemed to have been assigned to the SIP-PBX.

The requirement is satisfied.

REQ2 - The mechanism MUST allow a set of assigned telephone numbers to comprise E.164 numbers, which can be in contiguous ranges, discrete, or in any combination of the two.

The requirement is satisfied; the DIDs associated with a registration is established by bilateral agreement between the SSP and the PBX, and is not part of the mechanism described in this document.

REQ3 - The mechanism MUST allow a SIP-PBX to register reachability information with its SSP, in order to enable the SSP to route to the SIP-PBX inbound requests targeted at assigned telephone numbers.

The requirement is satisfied.

REQ4 - The mechanism MUST NOT prevent UAs attached to a SIP-PBX registering with the SIP-PBX on behalf of AORs based on assigned telephone numbers in order to receive requests targeted at those telephone numbers, without needing to involve the SSP in the registration process.

The requirement is satisfied; in the presumed architecture, PBX terminals register with the PBX, and require no interaction with the SSP.

REQ5 - The mechanism MUST allow a SIP-PBX to handle internally requests originating at its own UAs and targeted at its assigned telephone numbers, without routing those requests to the SSP.

The requirement is satisfied; PBXes may recognize their own DID and their own GRUUs, and perform on-PBX routing without sending the requests to the SSP.

REQ6 - The mechanism MUST allow a SIP-PBX to receive requests to its assigned telephone numbers originating outside the SIP-PBX and arriving via the SSP, so that the PBX can route those requests onwards to its UAs, as it would for internal requests to those telephone numbers.

The requirement is satisfied

REQ7 - The mechanism MUST provide a means whereby a SIP-PBX knows which of its assigned telephone numbers an inbound request from its SSP is targeted at.

The requirement is satisfied. For ordinary calls and calls using Public GRUUs, the DID is indicated in the user portion of the Request-URI. For calls using Temp GRUUs constructed with the mechanism described in [Section 7.1.2.2 \(Approach 2 - Anonymous Public GRUUs\)](#), the "sg" parameter provides a correlation token the PBX can use to identify which terminal the call should be routed to.

REQ8 - The mechanism MUST provide a means of avoiding problems due to one side using the mechanism and the other side not.

The requirement is satisfied through the 'gin' option tag and the 'bnc' Contact parameter.

REQ9 - The mechanism MUST observe SIP backwards compatibility principles.

The requirement is satisfied through the 'gin' option tag.

REQ10 - The mechanism MUST work in the presence of intermediate SIP entities on the SSP side of the SIP-PBX-to-SSP interface (i.e., between the SIP-PBX and the SSP's domain proxy), where those intermediate SIP entities need to be on the path of inbound requests to the PBX.

The requirement is satisfied through the use of the Path mechanism defined in RFC 3327 [\[6\] \(Willis, D. and B. Hoeneisen, "Session Initiation Protocol \(SIP\) Extension Header Field for Registering Non-Adjacent Contacts," December 2002.\)](#)

REQ11 - The mechanism MUST work when a SIP-PBX obtains its IP address dynamically.

The requirement is satisfied by allowing the PBX to use an IP address in the Bulk Number Contact URI contained in a REGISTER Contact header field.

REQ12 - The mechanism MUST work without requiring the SIP-PBX to have a domain name or the ability to publish its domain name in the DNS.

The requirement is satisfied by allowing the PBX to use an IP address in the Bulk Number Contact URI contained in a REGISTER Contact header field.

REQ13 - For a given SIP-PBX and its SSP, there MUST be no impact on other domains, which are expected to be able to use normal RFC 3263 procedures to route requests, including requests needing to be routed via the SSP in order to reach the SIP-PBX.

The requirement is satisfied by allowing the domain name in the Request URI used by external entities to resolve to the SSP's servers via normal RFC 3263 resolution procedures.

REQ14 - The mechanism MUST be able to operate over a transport that provides integrity protection and confidentiality.

The requirement is satisfied; nothing in the proposed mechanism prevent the use of TLS between the SSP and the PBX.

REQ15 - The mechanism MUST support authentication of the SIP-PBX by the SSP and vice versa.

The requirement is satisfied; PBXes may employ either SIP digest authentication or mutually-authenticated TLS for authentication purposes.

REQ16 - The mechanism MUST allow the SIP-PBX to provide its UAs with public or temporary Globally Routable UA URIs (GRUUs) [\[10\] \(Rosenberg, J., "Obtaining and Using Globally Routable User Agent URIs \(GRUUs\) in the Session Initiation Protocol \(SIP\)," October 2009.\)](#).

The requirement is satisfied via the mechanisms detailed in [Section 7.1 \(Globally Routable User-Agent URIs \(GRUU\)\)](#).

REQ17 - The mechanism MUST NOT preclude the ability of the SIP-PBX to route on-PBX requests directly, without hair-pinning the signaling through the SSP.

The requirement is satisfied; PBXes may recognize their own DID and their own GRUUs, and perform on-PBX routing without sending the requests to the SSP. (Note that this requirement duplicates REQ5, and will probably be removed in a future version of the requirements document.)

REQ18 - The mechanism MUST work over any existing transport specified for SIP, including UDP.

The requirement is satisfied to the extent that UDP can be used for REGISTER requests in general. The application of certain extensions and/or network topologies may exceed UDP MTU sizes, but such issues arise both with and without the mechanism described in this document. This document does not exacerbate such issues.

DES1 - The mechanism SHOULD allow an SSP to exploit its mechanisms for providing SIP service to ordinary subscribers in order to provide a SIP trunking service to SIP-PBXes.

The desired property is satisfied; the routing mechanism described in this document is identical to the routing performed for singly-registered AORs.

DES2 - The mechanism SHOULD scale to SIP-PBX's of several thousand assigned telephone numbers.

The desired property is satisfied; nothing in this document precludes DID pools of arbitrary size.

DES3 - The mechanism SHOULD scale to support several thousand SIP-PBX's on a single SSP.

The desired property is satisfied; nothing in this document precludes an arbitrary number of PBXes from attaching to a single SSP.

DES4 - The mechanism SHOULD require relatively modest changes to a substantial population of existing SSP and SIP-PBX implementations, in order to encourage a fast market adoption of the standardized mechanism.

The desired property is difficult to evaluate in the context of any solution. The mechanism proposed in this document uses the REGISTER method, which is the method preferred by many existing PBX deployments. The handling of request routing logic is nearly identical to that of RFC 3261 proxy/registrars, allowing implementors to leverage existing proxy/registrar code.

---

## 10. IANA Considerations

[TOC](#)

This document registers a new SIP option tag to indicate support for the mechanism it defines, plus two new SIP URI parameters.

---

### 10.1. New SIP Option Tag

[TOC](#)

This section defines a new SIP option tag per the guidelines in Section 27.1 of RFC 3261[2] ([Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.](#)).

**Name:**

gin

**Description:** This option tag is used to identify the extension that provides Registration for Multiple Phone Numbers in SIP. When present in a Require or Proxy-Require header field of a REGISTER request, it indicates that support for this extension is required of registrars and proxies, respectively, that are a party to the registration transaction.

**Reference:** RFCXXXX (this document)

---

## 10.2. New SIP URI Parameters

[TOC](#)

This specification defines two new SIP URI parameters, as per the registry created by RFC 3969 [\[4\] \(Camarillo, G., "The Internet Assigned Number Authority \(IANA\) Uniform Resource Identifier \(URI\) Parameter Registry for the Session Initiation Protocol \(SIP\)," December 2004.\)](#).

---

### 10.2.1. 'bnc' SIP URI paramter

[TOC](#)

**Parameter Name:** bnc

**Predefined Values:** No (no values are allowed)

**Reference:** RFCXXXX (this document)

---

### 10.2.2. 'sg' SIP URI paramter

[TOC](#)

**Parameter Name:** sg

**Predefined Values:** No

**Reference:** RFCXXXX (this document)

---

[TOC](#)

## 11. Security Considerations

There are certainly security implications associated with the mechanisms described in this document, mostly dealing with the unprecedented semantic inclusion of multiple AORs in a single REGISTER request. This section will be formulated following an analysis of the security impact of GIN on Path, Service-Route, and Outbound.

---

## 12. Acknowledgements

[TOC](#)

Thanks to John Elwell for his requirements analysis of the mechanism described in this document, and to Dean Willis for his analysis of the interaction between this mechanism and the Path and Service-Route extensions.

---

## 13. References

[TOC](#)

### 13.1. Normative References

[TOC](#)

[1]	<a href="#">Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels,"</a> BCP 14, RFC 2119, March 1997 ( <a href="#">TXT</a> , <a href="#">HTML</a> , <a href="#">XML</a> ).
[2]	Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, " <a href="#">SIP: Session Initiation Protocol</a> ," RFC 3261, June 2002 ( <a href="#">TXT</a> ).
[3]	Rosenberg, J. and H. Schulzrinne, " <a href="#">Session Initiation Protocol (SIP): Locating SIP Servers</a> ," RFC 3263, June 2002 ( <a href="#">TXT</a> ).
[4]	Camarillo, G., " <a href="#">The Internet Assigned Number Authority (IANA) Uniform Resource Identifier (URI) Parameter Registry for the Session Initiation Protocol (SIP)</a> ," BCP 99, RFC 3969, December 2004 ( <a href="#">TXT</a> ).

---

### 13.2. Informative References

[TOC](#)

[5]	Elwell, J. and H. Kaplan, " <a href="#">Requirements for multiple address of record (AOR) reachability information in the Session Initiation Protocol (SIP)</a> ," draft-ietf-martini-reqs-03 (work in progress), March 2010 ( <a href="#">TXT</a> ).
[6]	

	Willis, D. and B. Hoeneisen, " <a href="#">Session Initiation Protocol (SIP) Extension Header Field for Registering Non-Adjacent Contacts</a> ," RFC 3327, December 2002 ( <a href="#">TXT</a> ).
[7]	Willis, D. and B. Hoeneisen, " <a href="#">Session Initiation Protocol (SIP) Extension Header Field for Service Route Discovery During Registration</a> ," RFC 3608, October 2003 ( <a href="#">TXT</a> ).
[8]	Rosenberg, J., " <a href="#">A Session Initiation Protocol (SIP) Event Package for Registrations</a> ," RFC 3680, March 2004 ( <a href="#">TXT</a> ).
[9]	Jennings, C., Mahy, R., and F. Audet, " <a href="#">Managing Client-Initiated Connections in the Session Initiation Protocol (SIP)</a> ," RFC 5626, October 2009 ( <a href="#">TXT</a> ).
[10]	Rosenberg, J., " <a href="#">Obtaining and Using Globally Routable User Agent URIs (GRUUs) in the Session Initiation Protocol (SIP)</a> ," RFC 5627, October 2009 ( <a href="#">TXT</a> ).

---

## Author's Address

[TOC](#)

	Adam Roach
	Tekelec
	17210 Campbell Rd.
	Suite 250
	Dallas, TX 75252
	US
Email:	<a href="mailto:adam@nostrum.com">adam@nostrum.com</a>