Network Working Group	M. Barnes
Internet-Draft	Nortel
Obsoletes: RFC4244	F. Audet
(if approved)	Skype Labs
Intended status: Standards Track	S. Schubert
Expires: April 29, 2010	NTT
	J. van Elburg
	Detecon International Gmbh
	C. Holmberg
	Ericsson
	October 26, 2009

TOC

An Extension to the Session Initiation Protocol (SIP) for Request History Information
draft-barnes-sipcore-rfc4244bis-03.txt

Status of this Memo

This Internet-Draft will expire on April 29, 2010.

Abstract

This document defines a standard mechanism for capturing the history information associated with a Session Initiation Protocol (SIP) request. This capability enables many enhanced services by providing the information as to how and why a call arrives at a specific application or user. This document defines an optional SIP header, History-Info, for capturing the history information in requests.

Table of Contents

---

1. Introduction

TOC

Many services that SIP is anticipated to support require the ability to determine why and how the call arrived at a specific application. Examples of such services include (but are not limited to) sessions initiated to call centers via "click to talk" SIP Uniform Resource Locators (URLs) on a web page, "call history/logging" style services within intelligent "call management" software for SIP User Agents (UAs), and calls to voicemail servers. Although SIP implicitly provides the redirect/retarget capabilities that enable calls to be routed to chosen applications, there is a need for a standard mechanism within SIP for communicating the retargeting history of such a request. This "request history" information allows the receiving application to determine hints about how and why the call arrived at the application/user.

This document defines a SIP header, History-Info, to provide a standard mechanism for capturing the request history information to enable a wide variety of services for networks and end-users. The History-Info header provides a building block for development of new services.

The requirements for this document are described in Appendix A (Request History Requirements).

---

2. Conventions and 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 [RFC2119] (Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels," March 1997.).

The term "retarget" is used in this document to refer both to the process of a Proxy Server/User Agent Client (UAC) changing a Uniform Resource Identifier (URI) in a request based on the rules for determining request targets as described in Section 16.5 of [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.) and the subsequent forwarding of that request as described in section 16.6 of [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.).

The term "forward" is used consistent with the terminology in [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.). Noting that [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.) uses the term "forwarding" to describe a proxy's handling of requests for domains for which is not responsible, as well as to describe the basic "forwarding" of a request (in section 16.6) once a target has been determined. However, the context of the usage is sufficient to differentiate the slightly different meanings.

The terms "location service" and "redirect" are used consistent with the terminology in [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.).

---

3. Overview of Operations

TOC

SIP implicitly provides retargeting capabilities that enable calls to be routed to specific applications as defined in [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.). The motivation for capturing the request history is that in the process of retargeting a request, old routing information can be

forever lost. This lost information may be important history that allows elements to which the call is retargeted to process the call in a locally defined, application-specific manner. This document defines a mechanism for transporting the request history. Application-specific behavior is outside the scope of this specification.
Current network applications provide the ability for elements involved with the call to exchange additional information relating to how and why the call was routed to a particular destination. The following are examples of such applications:

1. Web "referral" applications, whereby an application residing within a web server determines that a visitor to a website has arrived at the site via an "associate" site that will receive some "referral" commission for generating this traffic

2. Email forwarding whereby the forwarded-to user obtains a "history" of who sent the email to whom and at what time

3. Traditional telephony services such as voicemail, call-center "automatic call distribution", and "follow-me" style services

Several of the aforementioned applications currently define application-specific mechanisms through which it is possible to obtain the necessary history information.
In addition, request history information could be used to enhance basic SIP functionality by providing the following:

*Some diagnostic information for debugging SIP requests. (Note that the diagnostic utility of this mechanism is limited by the fact that its use by entities that retarget is optional.)

*Capturing aliases and Globally Routable User Agent URIs (GRUUs) [RFC5627] (Rosenberg, J., “Obtaining and Using Globally Routable User Agent URIs (GRUUs) in the Session Initiation Protocol (SIP),” October 2009.), which can be overwritten by a home proxy upon receipt of the initial request.

*Facilitating the use of limited use addresses (minted on demand) and sub-addressing.

*Preserving service specific URIs that can be overwritten by a downstream proxy, such as those defined in [RFC3087] (Campbell, B. and R. Sparks, “Control of Service Context using SIP Request-URI,” April 2001.), and control of network announcements and IVR with SIP URI [RFC4240] (Burger, E., Van Dyke, J., and A. Spitzer, “Basic Network Media Services with SIP,” December 2005.).

*A stronger security solution for SIP. A side effect is that each proxy that captures the "request history" information in a secure manner provides an additional means (without requiring signed

keys) for the original requestor to be assured that the request was properly retargeted.

The fundamental functionality provided by the request history information is the ability to inform proxies and UAs involved in processing a request about the history or progress of that request (CAPABILITY-req, see Appendix A (Request History Requirements)). The solution is to capture the Request-URIs as a request is retargeted, in a new header for SIP messages: History-Info (CONTENT-req, see Appendix A (Request History Requirements)). This allows for the capturing of the history of a request that would be lost with the normal SIP processing involved in the subsequent retargeting of the request. This solution proposes no changes in the fundamental determination of request targets or in the request forwarding as defined in Sections 16.5 and 16.6 of the SIP protocol specification [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.).
The History-Info header can appear in any request not associated with an established dialog (e.g., INVITE, REGISTER, MESSAGE, REFER and OPTIONS, PUBLISH and SUBSCRIBE, etc.) (REQUEST-VALIDITY-req, seeAppendix A (Request History Requirements)) and any valid response to these requests (ISSUER-req, seeAppendix A (Request History Requirements)).
This specification defines parameters (see Section 6.1 (Definition)) for carrying the following information in the History-Info header:

*Targeted-to-URI: The targeted-to-URI entry captures the Request-URI for the specific Request as it is forwarded.

*Index: The index reflects the chronological order of the information, indexed to also reflect the forking and nesting of requests.

*Reason: Reason describes why an entry was retargeted.

*Privacy: Privacy is used to request that entries be anonymized.

*Target: The target parameter indicates the mechanism by which the new target is determined, i.e., a "registered contact", or a "mapped URI"

The following is an illustrative example of usage of History-Info. In this example, Alice (sip:alice@atlanta.example.com) calls Bob (sip:bob@biloxi.example.com). Alice's home proxy (sip:atlanta.example.com) forwards the request to Bob's proxy (sip:biloxi.example.com). When the request arrives at sip:biloxi.example.com, it does a location service lookup for bob@biloxi.example.com and changes the target of the request to Bob's

Contact URIs provided as part of normal SIP registration. In this example, Bob is simultaneously contacted on a PC client and on a phone, and Bob answers on the PC client.
One important thing illustrated by this call flow is that without History-Info, Bob would "lose" the target information, including any parameters in the request URI. Bob can now recover that information by looking for the prior entry to the last hi-entry marked as "rc"
The formatting in this scenario is for visual purposes; thus, backslash and CRLF are used between the fields for readability and the headers in the URI are not shown properly formatted for escaping. Refer to Section 6.2 (Examples) for the proper formatting. Additional detailed scenarios are available in the Appendix B (Detailed call flows).

> Note: This example uses loose routing procedures.


Alice
atlanta.example.com
biloxi.example.com
Bob@pc
Bob@phone
INVITE sip:bob@biloxi.example.com;p=x
Supported: histinfo
INVITE sip:bob@biloxi.example.com;p=x
History-Info: <sip:bob@biloxi.example.com;p=x>;index=1
History-Info: <sip:bob@biloxi.example.com;p=x>;index=1.1
INVITE sip:bob@192.0.2.3
History-Info: <sip:bob@biloxi.example.com;p=x>;index=1
History-Info: <sip:bob@biloxi.example.com;p=x>;index=1.1
History-Info: <sip:bob@192.0.2.3>;index=1.1.1;rc
INVITE sip:bob@192.0.2.7
History-Info: <sip:bob@biloxi.example.com;p=x>;index=1
History-Info: <sip:bob@biloxi.example.com;p=x>;index=1.1
History-Info: <sip:bob@192.0.2.7>;index=1.1.2;rc
200
200
History-Info: <sip:bob@biloxi.example.com;p=x>;index=1
History-Info: <sip:bob@biloxi.example.com;p=x>;index=1.1
History-Info: <sip:bob@192.0.2.3>;index=1.1.1;rc
<===Proxy cancels INVITE==>
200
History-Info: <sip:bob@biloxi.example.com;p=x>;index=1
History-Info: <sip:bob@biloxi.example.com;p=x>;index=1.1
History-Info: <sip:bob@192.0.2.3>;index=1.1.1;rc
History-Info: <sip:bob@192.0.2.7?Reason=SIP;cause=487>;\
index=1.1.2;rc
200
History-Info: <sip:bob@biloxi.example.com;p=x>;index=1
History-Info: <sip:bob@biloxi.example.com;p=x>;index=1.1
History-Info: <sip:bob@192.0.2.3;index=1.1.1;rc
History-Info: <sip:bob@192.0.2.7?Reason=SIP;cause=487>;\
index=1.1.2;rc
ACK
ACK
ACK

Figure 1: Basic Call

---

4. General User Agent Behavior

TOC

This section describes the processing specific to UAs for the History-Info header.

4.1. User Agent Client (UAC) Behavior

TOC

The UAC SHOULD include the "histinfo" option tag in the Supported header in any request not associated with an established dialog for which the UAC would like the History-Info header in the response. In addition, the UAC MAY add a History-Info header, using the Request-URI of the request as the hi-target-to-uri, in which case the index MUST be set to a value of 1 in the hi-entry. As a result, intermediaries and the UAS will know at least the original Request-URI, and if the Request-URI was modified by a previous hop. Normally, UACs are not expected to include a History-Info header in an initial request as it is more of a Proxy function; the main reason it is allowed is for B2BUAs who are performing proxy-like functions like routing.
A UAC that does not want an hi-entry added due to privacy considerations MUST include a Privacy header with a priv-value(s) of "header" or "history." A UAC that wants to ensure that privacy not be applied to its identity MUST include a Privacy header with a priv-value of "none."
In the case where a UAC receives a 3xx response with a Contact header and sends a new request in response to it, the UAC MUST include in the outgoing request the previous hi-entry(s) received in the response. The UAC MUST evaluate the last hi-entry in the 3xx response and verify that they are the same (as per the procedures in section Section 4.2.1 (Redirect Server Behavior)); if the hi-entry is not the same as the value in contact, hi-entry MUST be added using the value of Contact. If the hi-entry for the redirection is not included in the 3xx response, then an hi-entry MUST be added to the outgoing request. In this case, the index MUST be created by reading and incrementing the value of the index from the previous hi-entry, thus following the same rules as those prescribed for a proxy in retargeting, described in Section 6.3.3 (Indexing in the History-Info Header). The reason MUST be

added per Section 6.3.2 (Reason in the History-Info Header). The hi-target and hi-aor attributes MUST NOT be added to this hi-entry since there is no way to know the mechanism by which the redirecting entity determined the URI in the Contact header nor whether the previous hi-targeted-to-uri was an AOR.
If no hi-entry for redirection were included at all in the 3xx response, and multiple redirection occurs, the UAC MAY attempt to synthetise the missing hi-entrie(s) before inserting the last one (as per the previous step). At a minimum, the last entry (as per the previous step) MUST be included.
With the exception of the processing of a 3xx response described above, the processing of the History-Info header received in the Response is application specific and outside the scope of this document.

4.2. User Agent Server (UAS) Behavior

TOC

Once the request terminates at the UAS, the processing of the information in the History-Info header by a UAS in a Request depends upon local policy and specific applications at the UAS that might make use of the information. Prior to any application usage of the information, the validity SHOULD be ascertained. For example, the entries MAY be evaluated to determine gaps in indices, which could indicate that an entry has been maliciously removed or removed for privacy reasons. Either way, an application MAY want to be aware of potentially missing information.
If the "histinfo" option tag is received in a request, the UAS MUST include any History-Info received in the request in the subsequent response. If privacy is required, entries MUST be anonymized using [RFC3323] (Peterson, J., “A Privacy Mechanism for the Session Initiation Protocol (SIP),” November 2002.). The UAS MUST follow the rules for a redirect server per Section 4.2.1 (Redirect Server Behavior) in generating a 3xx response.
The processing of History-Info in responses follows the methodology described in Section 16.7 of [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, “SIP: Session Initiation Protocol,” June 2002.), with the processing of History-Info headers adding an additional step, just before Step 9, "Forwarding the Response".

4.2.1. Redirect Server Behavior

TOC

A redirect server MUST include the History-Info headers received in the request in the 3XX response that it sends, and it MUST perform the following steps:

Step 1:

Adding Entries on Behalf of Previous Hops

If an incoming request does not already have a History-Info header field (e.g., the UAC does not include any History-Info header and no proxies in between support History-Info), or if the Request-URI of the incoming request does not match the last hi-entry (e.g., the last hop proxy does not support History-Info), the redirect server MUST insert an hi-entry. The redirect server MUST set the hi-targeted-to-uri to the value of Request URI in the incoming request, unless privacy is required. If privacy is required, the procedures of Section 6.3.1 (Privacy in the History-Info Header) MUST be used. The proxy MUST NOT include a hi-target attribute. The proxy MUST include an hi-index attribute as described in Section 6.3.3 (Indexing in the History-Info Header).

Step 2: Tagging the Last Incoming Entry

The redirect server then examines the last hi-entry of the History-Info header resulting from the previous step. If privacy is required for this entry, the procedures of Section 6.3.1 (Privacy in the History-Info Header) MUST be used for that entry. The Reason header MUST be added to that entry as per the procedures of Section 6.3.2 (Reason in the History-Info Header), and must be set to the proper SIP 3XX response.

Step 3: Generating New Entries for the Response

The redirect servert MUST add a new hi-entry for each of the Contact header URIs, which becomes the new Request-URIs when the recipient forwards the new Request. The index is created as described in Section 6.3.3 (Indexing in the History-Info Header). If privacy is required, the procedures of Section 6.3.1 (Privacy in the History-Info Header) MUST be used. A hi-target parameter MUST be included if the new Request-URI either represents another user or registered contact as per the procedures of Section 6.3.4 (Request Target in the History-Info Header).

Redirection is an iterative process, i.e., a redirect server may redirect "internally " more than one time. A typical example would be a redirect server that redirects a request first to a different user (i.e., it maps to a different AOR), and then redirects again to a registered contact bound to that new AOR. A redirect server that uses such mechamism SHOULD add multiple hi-entry fields to provide a logical description of retargeting process (e.g., bob@example.com to office@example.com to office@192.0.2.5). A Reason MAY be associated with the hi-targeted-to-uri that has been retargeted. See the example

in Appendix B.1 (Sequentially Forking (History-Info in Response)) for an example.

---

5. Proxy Behavior

TOC

The specific processing by proxies for adding the History-Info headers in Requests and Responses is described in this section for the following cases:

- *Forwarding of initial request (see Section 5.1.1 (Initial Request))
- *Resending based on failure response (see Section 5.1.2 (Re-sending based on failure response))
- *Resending based on redirection response (see Section 5.1.3 (Re-sending based on redirection response))

5.1. Adding the History-Info Header to Requests

TOC

This section describes the process of adding the History-Info Header to Requests.
Retargeting is an iterative process, i.e., a proxy may redirect "internally " more than one time. A typical example would be a proxy that redirects a request first to a different user (i.e., it maps to a different AOR), and then forwards to a registered contact bound to that new AOR. A proxy that uses such mechamism SHOULD add multiple hi-entry fields to provide a logical description of the retargeting process.

5.1.1. Initial Request

TOC

Upon receipt of an initial request for a dialog, or a standalone request, a proxy forwarding the request MUST perform the following steps. Note that those steps below do not apply if the request is being re-sent as a result of failure (i.e., timeout, reception of an error response), or redirection caused by receipt of a 3XX message).

> **Step 1:** Adding Entries on Behalf of Previous Hops
>
> If an incoming request does not already have a History-Info header field (e.g., the UAC does not include any History-Info

header and no proxies in between support History-Info), or if the Request-URI of the incoming request does not match the last hi-entry (e.g., the last proxy does not support History-Info), the proxy MUST insert an hi-entry. The proxy MUST set the hi-targeted-to-uri based to the value of Request URI in the incoming request, unless privacy is required. If privacy is required, the procedures of Section 6.3.1 (Privacy in the History-Info Header) MUST be used. The proxy MUST NOT include a hi-target attribute. The proxy MUST include an hi-index attribute as described in Section 6.3.3 (Indexing in the History-Info Header).

Step 2: Generating New Entries for Each Outgoing Requests

The proxy then proceeds to determining the request targets as per 16.5/[RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.) and request forwarding as per 16.6/[RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.). The proxy MUST add a separate hi-entry in each separate outgoing request for each of the current (outgoing) targets in the target set. The proxy MUST set the hi-targeted-to-uri in those separate hi-entry(s) to the value of the Request-URI of the current (outgoing) request, unless privacy is required. If privacy is required, the procedures of Section 6.3.1 (Privacy in the History-Info Header) MUST be used. The proxy MUST include a hi-target attribute for each of the separate entry(s) as described in Section 6.3.4 (Request Target in the History-Info Header). The proxy MUST include an hi-index for each of the separate hi-entry(s) as described in Section 6.3.3 (Indexing in the History-Info Header).

5.1.2. Re-sending based on failure response

TOC

When re-sending a request as a result of retargeting because of failure (i.e., either reception of error responses or a timeout which is considered to be an implicit 487 error response), the proxy MUST perform the following steps:

Step 1: Including the Entries from Error Responses & Timeouts

The proxy MUST build the History-Info header field(s) sent in the outgoing request using the aggregate information associated with the received error responses(s) and timeout(s) for all the branches that are generating failures, including the header

entries in the order indicated by the indexing (see Section 6.3.3 (Indexing in the History-Info Header)). If the received error response did not include any History-Info header fields, the proxy MUST use the same History-Info header fields that were sent in the outgoing request that failed to build the outgoing request.

Step 2: Tagging the Last Entries

The proxy then examines the last hi-entry of the History-Info that was just generated in Step 1 for each one of the branches that generated failures or timeouts and MUST add a Reason header for each one of those entries as per the procedures of Section 6.3.2 (Reason in the History-Info Header).

Step 3: Generating New Entries for Each Outgoing Requests

Same as per Step 3 above for the normal forwarding case.

5.1.3. Re-sending based on redirection response

TOC

When re-sending a request as a result of retargeting because of redirection (i.e., receipt of a 3XX response), the following steps apply:

Step 1: Including Previous Entries

If the received 3XX response does not include any History-Info header fields, the proxy MUST include the History-Info header fields that were sent in the outgoing request that is being redirected. The proxy MUST then perform Steps 2 and 3.

If the 3XX response contains a History-Info Header field, but the last entries does not correspond to the current target (i.e., they do not correspond to the Contact(s) in the 3XX), the proxy MUST include in the outgoing request the same History-Info header fields that were received in the 3XX response. The proxy MUST then perform Steps 2 and 3.

If the 3XX response contains a History-Info Header field and the last entries correspond to the current target (i.e., they correspond to the Contact(s) in the 3XX), the proxy MUST include in the outgoing request the same History-Info header fields that were received in the 3XX response. No other entries need to be added as this is the complete set: the proxy MUST NOT perform Steps 2 and 3.

Step 2:

Tagging the Last Entry

The proxy then examines the last hi-entry of the History-Info that was just generated in Step 1 and MUST add a Reason header this entry as per the procedures of Section 6.3.2 (Reason in the History-Info Header).

Step 3: Generating New Entries for Each Outgoing Requests

Same as per Step 3 above for the normal forwarding case, except that the hi-target parameter MUST NOT be set when the proxy receiving the 3xx does not know the mechanism by which this target was determined. For example, the proxy can not determine the hi-target mechanism when the domain of the Contact is not under the control of the proxy. However, if it under the control of the proxy, then it may be able to determine the mechanism (e.g., Bob can deflect a call to his SIP PC client to his cell phone).

5.2. Sending History-Info in Responses

TOC

A proxy that receives a Request with the "histinfo" option tag in the Supported header, SHOULD forward captured History-Info in subsequent, provisional, and final responses to the Request sent by the ultimate UAS (see Section 4.2 (User Agent Server (UAS) Behavior)).
A proxy MAY anonymize any hi-entry whose domain corresponds to a domain for which it is responsible (as per [RFC3323] (Peterson, J., “A Privacy Mechanism for the Session Initiation Protocol (SIP),” November 2002.)).
For example, anonymity may be required when responses are forwarded to a domain for which it is not responsible.
The processing of History-Info in responses follows the methodology described in Section 16.7 of [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, “SIP: Session Initiation Protocol,” June 2002.), with the processing of History-Info headers adding an additional step, just before Step 9, "Forwarding the Response".

---

6. The History-Info header field

TOC

6.1. Definition

TOC

History-Info is a header field as defined by [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, “SIP: Session Initiation Protocol,” June 2002.). "It may appear in any initial request for a dialog, standalone request or responses associated with these requests. For example, History-Info may appear in INVITE, REGISTER, MESSAGE, REFER, OPTIONS, SUBSCRIBE, and PUBLISH and any valid responses, plus NOTIFY requests that initiate a dialog.
The History-Info header carries the following information, with the mandatory parameters required when the header is included in a request or response:

*Targeted-to-URI (hi-targeted-to-uri): A mandatory parameter for capturing the Request-URI for the specific Request as it is forwarded.

*Index (hi-index): A mandatory parameter for History-Info reflecting the chronological order of the information, indexed to also reflect the forking and nesting of requests. The format for this parameter is a string of digits, separated by dots to indicate the number of forward hops and retargets. This results in a tree representation of the history of the request, with the lowest-level index reflecting a branch of the tree. By adding the new entries in order (i.e., following existing entries per the details in Section 5.1 (Adding the History-Info Header to Requests)), including the index and securing the header, the ordering of the History-Info headers in the request is assured (SEC-req-2, see Appendix A.1 (Security Requirements)). In addition, applications may extract a variety of metrics (total number of retargets, total number of retargets from a specific branch, etc.) based upon the index values.

*Reason: An optional parameter for History-Info, reflected in the History-Info header by including the Reason Header [RFC3326] (Schulzrinne, H., Oran, D., and G. Camarillo, “The Reason Header Field for the Session Initiation Protocol (SIP),” December 2002.) escaped in the hi-targeted-to-uri. A reason is included for the hi-targeted-to-uri that was retargeted as opposed to the hi-targeted-to-uri to which it was retargeted.

*Privacy: An optional parameter for History-Info, reflected in the History-Info header field values by including the Privacy Header [RFC3323] (Peterson, J., “A Privacy Mechanism for the Session Initiation Protocol (SIP),” November 2002.) escaped in the hi-targeted-to-uri or by adding the Privacy header to the Request. The latter case indicates that the History-Info entries for the

domain MUST be anonymized prior to forwarding, whereas the use of the Privacy header escaped in the hi-targeted-to-uri means that a specific hi-entry MUST be anonymized.

*Target (hi-target): An optional parameter for the History-Info indicating the mechanism by which the new target is determined, based on the procedures of 16.5 [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, “SIP: Session Initiation Protocol,” June 2002.). The hi-target is added for a hi-entry when it is first added in a History-Info header field, and only one value is permitted. Upon receipt of a request or response containing the History-Info header, a UA can determine the nature of the target. The following values are defined for this parameter:

-"rc": The entry is a contact that is bound to an AOR in an abstract location service. The AOR-to-contact binding has been placed into the location service by a SIP Registrar that received a SIP REGISTER request.

-"mp": The entry is a URI that represents another user. This occurs in cases where a request is statically or dynamically retargeted to another user. The index entry of the target of the original target is added as a parameter to the "mp" (i.e., it represents the "mapped from" target).

*Extension (hi-extension): A parameter to allow for future optional extensions. As per [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, “SIP: Session Initiation Protocol,” June 2002.), any implementation not understanding an extension should ignore it.

The following summarizes the syntax of the History-Info header, based upon the standard SIP syntax [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, “SIP: Session Initiation Protocol,” June 2002.):

```
History-Info = "History-Info" HCOLON hi-entry *(COMMA hi-entry)

hi-entry = hi-targeted-to-uri *(SEMI hi-param)

hi-targeted-to-uri = name-addr

hi-param = hi-index / hi-target / hi-extension

hi-index = "index" EQUAL 1*DIGIT *("." 1*DIGIT)

hi-target = "rc" / mp-param

mp-param = "mp" EQUAL 1*DIGIT *("." 1*DIGIT)

hi-extension = generic-param
```

The following rules apply:

*There MUST exactly 1 hi-index parameter per hi-entry.

*There MUST be no more than 1 hi-target parameter.

*They MAY be any number of hi-extension parameters.

*The ABNF definitions for "generic-param" and "name-addr" are from [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.).

6.2. Examples

TOC

The following provides some examples of the History-Info header. Note that the backslash and CRLF between the fields in the examples below are for readability purposes only.

```
History-Info: <sip:UserA@ims.example.com>;index=1;foo=bar

History-Info: <sip:UserA@ims.example.com?Reason=SIP%3B\
              cause%3D302>;index=1.1,\
              <sip:UserB@example.com?Privacy=history&Reason=SIP%3B\
              cause%3D486>;index=1.2;mp=1.1,\
              <sip:45432@192.168.0.3>;index=1.3;rc
```

6.3. Procedures

TOC

The following sections defines procedures for different parameters in the History-Info header. These procedures may be applicable to "processing entities" such as Proxies, Redirect Servers or User Agents.

6.3.1. Privacy in the History-Info Header

TOC

The privacy requirements for this document are described in Appendix A.2 (Privacy Requirements).

Since the History-Info header can inadvertently reveal information about the requestor as described in [RFC3323] (Peterson, J., “A Privacy Mechanism for the Session Initiation Protocol (SIP),” November 2002.), the Privacy header can be used to determine whether an intermediary can include the Request-URI in a Request that it receives (PRIV-req-2, see Appendix A.2 (Privacy Requirements)) or that it retargets (PRIV-req-1, see Appendix A.2 (Privacy Requirements)) as an entry in a History-Info header. Thus, the History- Info entry for that identity can be anonymized where the requestor has indicated a priv-value of Session- or Header-level privacy.

Privacy is associated with a specific history information entry, and perhaps any entry that corresponds to that same user, and not the History-Info header itself. This allows for anonymizing some entries, but not others, as required. For example, if Alice sends a request to Bob without any privacy, and Bob redirects to Carol with privacy setup for himself, Carol should receive a request where Alice's history information is present, but Bob's has been anonymized.

In addition, the History-Info header can reveal general routing information which may be viewed by a specific intermediary or network. Thus, a proxy can use local policy to determine whether the History-Info header entries for it's whole domain are private or not when exiting the domain through retargeting (PRIV-req-3). This is accomplished by adding a new priv-value, history, to the Privacy header [RFC3323] (Peterson, J., “A Privacy Mechanism for the Session Initiation Protocol (SIP),” November 2002.) indicating that a specific History-Info header entry can not be forwarded outside the domain.

It is recognized that satisfying the privacy requirements can impact the functionality of this solution by overriding the request to generate the information.

If there is a Privacy header in the request with a priv-value of "session", "header", or "history", an hi-entry SHOULD be added if the request is being retargeted to a URI associated with a domain for which the processing entity is responsible. If there is no Privacy header, but the processing entity's local policies indicate that the hi-entry(s) cannot be forwarded beyond the domain for which this

intermediary is responsible, then a Privacy header with a priv-value of "history" SHOULD be associated with each hi-entry added by the proxy as the request is forwarded within the domain.
If a request is being retargeted to a URI associated with a domain for which the processing identity is not responsible and there is a Privacy header in the request with a priv-value of "session", "header", or "history", the processing entity MUST anonymize hi-entry(s) as per [RFC3323] (Peterson, J., “A Privacy Mechanism for the Session Initiation Protocol (SIP),” November 2002.) prior to forwarding, unless the processing entity knows a priori that it can rely on a downstream processing entity within its domain to apply the requested privacy or local policy allows the forwarding.

6.3.2. Reason in the History-Info Header

TOC

For retargets that are the result of an explicit SIP response, a Reason MUST be associated with the hi-targeted-to-uri. If the SIP response does not include a Reason header (see [RFC3326] (Schulzrinne, H., Oran, D., and G. Camarillo, “The Reason Header Field for the Session Initiation Protocol (SIP),” December 2002.)), the SIP Response Code that triggered the retargeting MUST be included as the Reason associated with the hi-targeted-to-uri that has been retargeted. If the response contains a Reason header for a protocol that is not SIP (e.g., Q.850), it MUST be captured as an additional Reason associated with the hi-targeted-to- uri that has been retargeted, along with the SIP Response Code. If the Reason header is a SIP reason, then it MUST be used as the Reason associated with the hi-targeted-to-uri rather than the SIP response code.
If a request has timed out (instead of being explicitly rejected), it SHOULD be treated as if a 487 "Request Terminated" error response code was received.

6.3.3. Indexing in the History-Info Header

TOC

In order to maintain ordering and accurately reflect the nesting and retargeting of the request, an index MUST be included along with the Targeted-to-URI being captured. Per the syntax in Section 6 (The History-Info header field), the index consists of a dot-delimited series of digits (e.g., 1.1.2). Each dot reflects a hop or level of nesting; thus, the number of hops is determined by the total number of dots. Within each level, the integer reflects the number of peer entities to which the request has been routed. Thus, the indexing results in a logical tree representation for the history of the Request. For each level of indexing, the index MUST start at 1. An

increment of 1 MUST be used for advancing to a new branch. The first entry MUST be set to 1.
The basic rules for adding the index are summarized as follows:

1. Basic Forwarding: In the case of a Request that is being forwarded, the index is determined by adding another sub-level of indexing since the depth/length of the branch is increasing. To accomplish this, the processing entity reads the value from the History-Info header in the received request, if available, and adds another level of indexing by appending the dot delimiter followed by an initial index for the new level of 1. For example, if the index in the last History-Info header field in the received request is 1.1, this proxy would initialize its index to 1.1.1 and forward the request.

2. Retargeting within a processing entity - 1st instance: For the first instance of retargeting within a processing entity, the calculation of the index follows that prescribed for basic forwarding.

3. Retargeting within a processing entity - subsequent instance: For each subsequent retargeting of a request by the same processing entity, another branch is added. With the index for each new branch calculated by incrementing the last/lowest digit at the current level, the index in the next request forwarded by this same processing entity, following the example above, would be 1.1.2.

4. Retargeting based upon a Response: In the case of retargeting due to a specific response (e.g., 302), the index would be calculated per rule 3. That is, the lowest/last digit of the index is incremented (i.e., a new branch is created), with the increment of 1. For example, if the index in the History-Info header of the received request was 1.2, then the index in the History-Info header field for the new hi-targeted- to-URI would be 1.3.

5. Forking requests: If the request forwarding is done in multiple forks (sequentially or in parallel), the index MUST be captured for each forked request per the rules above, with each new Request having a unique index. Each index are sequentially assigned. For example, if the index in the last History-Info header field in the received request is 1.1, this processing entity would initialize its index to 1.1.1 for the first fork, 1.1.2 for the second, and so forth (see Figure 1 (Basic Call) for an example). Note that for each individual fork, only the entry corresponding that that fork is included (e.g., the entry for fork 1.1.1 is not included in the request sent to fork 1.1.2, and vice-versa).

6. When a response is built and it represents the aggregate of multiple forks (e.g., multiple forks that fail), the processing entity builds the subsequent response using the aggregated information associated with each of those forks and including the header entries in the order indicated by the indexing. For example, if a procesing entity received failure responses for forks 1.1.1 and 1.1.2, it would forward both the 1.1.1 and 1.1.2 entries to 1.1. See Appendix B.1 (Sequentially Forking (History-Info in Response)) for an example. Responses are processed as described in Section 16.7 of [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.) with the aggregated History-Info entries processed similar to Step 7 "Aggregate Authentication Header Field Values".

6.3.4. Request Target in the History-Info Header

TOC

The value for the hi-target attribute is based upon the mechanism by which the new target has been determined per the procedures described in 16.5/[RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol," June 2002.). The following describes how the specific values for the hi-target attribute are determined:

*If the Request-URI is a contact that is bound to an AOR in an abstract location service for the domain for which the processing entity is responsible, and the AOR-to-contact binding has been placed into the location service by a SIP Registrar that received a REGISTER request, the hi-target attribute MUST be added to the hi-entry with a value of "rc."

*If the Request-URI is a URI that represents another user than the one indicated by the incoming Request-URI, as this would occur in cases where a request is statically or dynamically retargeted to another user, the hi-target attribute MUST be added to the hi-entry with a value of "mp." The index of the entry corresponding to the original target (i.e., the "mapped-from" target) MUST be added as a parameter to "mp".

TOC

7. Application Considerations

As seen by the example scenarios in the Appendix B (Detailed call flows), History-Info provides a very flexible building block that can be used by intermediaries and UAs for a variety of services. As such, any services making use of History-Info must be designed with the following considerations:

1. History-Info is optional; thus, a service MUST define default behavior for requests and responses not containing History-Info headers.

2. History-Info may be impacted by privacy considerations. Applications requiring History-Info need to be aware that if Header-, Session-, or History-level privacy is requested by a UA (or imposed by an intermediary) that History-Info may not be available in a request or response. This would be addressed by an application in the same manner as the previous consideration by ensuring there is reasonable default behavior should the information not be available.

3. History-Info may be impacted by local policy. Each application making use of the History-Info header SHOULD address the impacts of the local policies on the specific application (e.g., what specification of local policy is optimally required for a specific application and any potential limitations imposed by local policy decisions). Note that this is related to the optionality and privacy considerations identified in 1 and 2 above, but goes beyond that. For example, due to the optionality and privacy considerations, an entity may receive only partial History-Info entries; will this suffice? Note that this would be a limitation for debugging purposes, but might be perfectly satisfactory for some models whereby only the information from a specific intermediary is required.

---

8. Security Considerations

TOC

The security requirements for this document are specified in Appendix A.1 (Security Requirements).
This document defines a header for SIP. The use of the Transport Layer Security (TLS) protocol [RFC5246] (Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2," August 2008.) as a mechanism to ensure the overall confidentiality of the History-Info headers (SEC-req-4) is strongly RECOMMENDED. This results in History-Info having at least the same level of security as other headers in SIP that are inserted by intermediaries. With TLS, History-Info headers are

no less, nor no more, secure than other SIP headers, which generally have even more impact on the subsequent processing of SIP sessions than the History-Info header.
With the level of security provided by TLS (SEC-req-3), the information in the History-Info header can thus be evaluated to determine if information has been removed by evaluating the indices for gaps (SEC-req-1, SEC-req-2). It would be up to the application to define whether it can make use of the information in the case of missing entries.
Note that while using the SIPS scheme (as per [RFC5630] (Audet, F., "The Use of the SIPS URI Scheme in the Session Initiation Protocol (SIP)," October 2009.)) protects History-Info from tampering by arbitrary parties outside the SIP message path, all the intermediaries on the path are trusted implicitly. A malicious intermediary could arbitrarily delete, rewrite, or modify History-Info. This specification does not attempt to prevent or detect attacks by malicious intermediaries.

---

9. IANA Considerations

TOC

This document requires several IANA registrations detailed in the following sections.
This document updates [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.) but uses the same SIP header field name and option tag. The IANA registry needs to update the references to [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.) witht [RFCXXXX].

9.1. Registration of New SIP History-Info Header

TOC

This document defines a SIP header field name: History-Info and an option tag: histinfo. The following changes have been made to http:/// www.iana.org/assignments/sip-parameters The following row has been added to the header field section:.
The following row has been added to the header field section:

Header Name	Compact Form	Reference
History-Info	none	[RFCXXXX]

The following has been added to the Options Tags section:

Name	Description	Reference
histinfo	When used with the Supported header, this option tag indicates the UAC supports the History Information to be captured for requests and returned in subsequent responses. This tag is not used in a Proxy-Require or Require header field since support of History-Info is optional.	[RFCXXXX]

Note to RFC Editor: Please replace RFC XXXX with the RFC number of this specification.

9.2. Registration of "history" for SIP Privacy Header

TOC

This document defines a priv-value for the SIP Privacy header: history The following changes have been made to http://www.iana.org/assignments/sip-priv-values The following has been added to the registration for the SIP Privacy header:

Name	Description	Registrant	Reference
history	Privacy requested for History-Info header(s)	Mary Barnes mary.barnes@nortel.com	[RFCXXXX]

Note to RFC Editor: Please replace RFC XXXX with the RFC number of this specification.

---

10. Contributors

TOC

Cullen Jennings, Mark Watson, and Jon Peterson contributed to the development of the initial requirements for [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.).
Jonathan Rosenberg, Christer Holmberg, Hans Erik van Elburg and Shida Schubert produced the document that provided much of the content for this specification.

---

11. Acknowledgements

TOC

The editor would like to acknowledge the constructive feedback provided by Robert Sparks, Paul Kyzivat, Scott Orton, John Elwell, Nir Chen, Palash Jain, Brian Stucker, Norma Ng, Anthony Brown, Jayshree Bharatia, Jonathan Rosenberg, Eric Burger, Martin Dolly, Roland Jesske, Takuya Sawada, Sebastien Prouvost, and Sebastien Garcin in the development of [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.). The editor would like to acknowledge the significant input from Rohan Mahy on some of the normative aspects of the ABNF for [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.), particularly around the need for and format of the index and around the security aspects.
Thanks to Ian Elz for his feedback on privacy.

---

12. Changes from RFC 4244

TOC

This RFC replaces [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.).
Deployment experience with [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.) over the years has shown a number of issues, warranting an update:

* In order to make [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.) work in "real life", one needs to make "assumptions" on how History-Info is used. For example, many implementations filter out many entries, and only leave specific entries corresponding, for example, to first and last redirection. Since vendors uses different rules, it causes significant interoperability isssues.

* [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.) is overly permissive and evasive about recording entries, causing interoperability issues.

* The examples in the call flows had errors, and confusing because they often assume "loose routing."

* [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.) has lots of repetitive and unclear text

* [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.) gratuitly mandates the use of TLS on every hop. No existing implementation enforces this rule, and instead, the use of TLS or not is a general SIP issue, not an [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.) issue per se.

* [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.) does not include clear procedures on how to deliver current target URI information to the UAS when the Request-URI is replaced with a contact.

* [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.) does not allow for marking History-Info entries for easy processing by User Agents.

This specification is backwards compatible with [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.). The following summarizes the functional changes:

1. Added a tag to indicate the mechanism by which the target for an outgoing request is determined.

2. Rather than recommending that entries be removed in the case of certain values of the privacy header, recommend that the entries are anonymized.

3. Updated processing/handling for 3xx responses to ensure accuracy of the new tags - i.e., the redirecting entity must add the new entry since the proxy does not have access to the information as to how the Contact was determined.

4. Updated the security section to be equivalent to the security recommendations for other SIP headers inserted by intermediaries.

The first 2 changes are intended to facilitate application usage of the History-Info header and eliminate the need to make assumptions based upon the order of the entries and ensure that the most complete set of information is available to the applications.
In addition, editorial changes were done to both condense and clarify the text and examples were simplified and updated to reflect the protocol changes.

12.1. Backwards compatibility

TOC

Proxies conforming to this specification tag the hi-entry parameters with an hi-target parameter. The hi-target parameter did not exist in [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.); therefore, [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.) implementations do not tag the hi-entry parameters. This tagging allows for distinguishing entries that were added by an [RFC4244] (Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," November 2005.) entity, versus one that was added by an entity conforming to this specification.

---

13. Changes since last Version

TOC

NOTE TO THE RFC-Editor: Please remove this section prior to publication as an RFC.
Changes from barnes-sipcore-4244bis-02 to 03:

1. Fixed problem with indices in example in Appendix B.2 (Voicemail).

2. Removed oc and rt from the Hi-target parameter.

3. Removed aor tag

4. Added index parameter to "mp"

5. Added use-cases and call-flows from target-uri into appendix.

Changes from barnes-sipcore-4244bis-01 to 02:

1. Added hi-aor parameter that gets marked on the "incoming" hi-entry.

2. Hi-target parameter defined to be either rc, oc, mp, rt, and now gets included when adding an entry.

3. Added section on backwards compatibility, as well as added the recognition and handling of requests that do not support this specification in the appropriate sections.

4. Updated redirect server/3xx handling to support the new parameters - i.e., the redirecting entity must add the new

entry since the proxy does not have access to the information as to how the Contact was determined.

5. Added section on normative differences between this document and RFC 4244.

6. Restructuring of document to be more in line with current IETF practices.

7. Moved Requirements section into an Appendix.

8. Fixed ABNF to remove unintended ordering requirement on hi-index that was introduced in attempting to illustrate it was a mandatory parameter.

Changes from barnes-sipcore-4244bis-00 to 01 :

1. Clarified "retarget" definition.

2. Removed privacy discussion from optionality section - just refer to privacy section.

3. Removed extraneous text from target-parameter (leftover from sip-4244bis). Changed the terminology from the "reason" to the "mechanism" to avoid ambiguity with parameter.

4. Various changes to clarify some of the text around privacy.

5. Reverted proxy response handling text to previous form - just changing the privacy aspects to anonymize, rather than remove.

6. Other editorial changes to condense and simplify.

7. Moved Privacy examples to Appendix.

8. Added forking to Basic call example.

Changes from barnes-sip-4244bis-00 to barnes-sipcore-4244bis-00:

1. Added tags for each type of retargeting including proxy hops, etc. - i.e., a tag is defined for each specific mechanism by which the new Request-URI is determined. Note, this is extremely helpful in terms of backwards compatibility.

2. Fixed all the examples. Made sure loose routing was used in all of them.

3. Removed example where a proxy using strict routing is using History-Info for avoiding trying same route twice.

4. Remove redundant Redirect Server example.

5. Index are now mandated to start at "1" instead of recommended.

6. Clarified 3xx behavior as the entity sending the 3XX response MUST add the hi-target attribute to the previous hi-entry to ensure that it is appropriately tagged (i.e., it's the only one that knows how the contact in the 3xx was determined.)

7. Removed lots of ambiguity by making many "MAYs" into "SHOULDs" and "some "SHOULDs" into "MUSTs".

8. Privacy is now recommended to be done by anonymizing entries as per RFC 3323 instead of by removing or omitting hi-entry(s).

9. Requirement for TLS is now same level as per RFC 3261.

10. Clarified behavior for "Privacy" (i.e., that Privacy is for Hi-entries, not headers).

11. Removed "OPTIONALITY" as specific requirements, since it's rather superflous.

12. Other editorial changes to remove redundant text/sections.

Changes from RFC4244 to barnes-sip-4244bis-00:

1. Clarified that HI captures both retargeting as well as cases of just forwarding a request.

2. Added descriptions of the usage of the terms "retarget", "forward" and "redirect" to the terminology section.

3. Added additional examples for the functionality provided by HI for core SIP.

4. Added hi-target parameter values to HI header to ABNF and protocol description, as well as defining proxy, UAC and UAS behavior for the parameter.

5. Simplified example call flow in section 4.5. Moved previous call flow to appendix.

6. Fixed ABNF per RFC4244 errata "dot" -> "." and added new parameter.

TOC

14. References

14.1. Normative References

TOC

[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 (TXT).
[RFC3326]	Schulzrinne, H., Oran, D., and G. Camarillo, "The Reason Header Field for the Session Initiation Protocol (SIP)," RFC 3326, December 2002 (TXT).
[RFC3323]	Peterson, J., "A Privacy Mechanism for the Session Initiation Protocol (SIP)," RFC 3323, November 2002 (TXT).
[RFC2119]	Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels," BCP 14, RFC 2119, March 1997 (TXT, HTML, XML).
[RFC5246]	Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2," RFC 5246, August 2008 (TXT).
[RFC4244]	Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information," RFC 4244, November 2005 (TXT).

14.2. Informative References

TOC

[RFC5627]	Rosenberg, J., "Obtaining and Using Globally Routable User Agent URIs (GRUUs) in the Session Initiation Protocol (SIP)," RFC 5627, October 2009 (TXT).
[RFC5630]	Audet, F., "The Use of the SIPS URI Scheme in the Session Initiation Protocol (SIP)," RFC 5630, October 2009 (TXT).
[RFC3087]	Campbell, B. and R. Sparks, "Control of Service Context using SIP Request-URI," RFC 3087, April 2001 (TXT).
[RFC4240]	Burger, E., Van Dyke, J., and A. Spitzer, "Basic Network Media Services with SIP," RFC 4240, December 2005 (TXT).
[RFC5039]	Rosenberg, J. and C. Jennings, "The Session Initiation Protocol (SIP) and Spam," RFC 5039, January 2008 (TXT).
[RFC4458]	Jennings, C., Audet, F., and J. Elwell, "Session Initiation Protocol (SIP) URIs for Applications such as Voicemail and Interactive Voice Response (IVR)," RFC 4458, April 2006 (TXT).
[RFC3761]	Faltstrom, P. and M. Mealling, "The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM)," RFC 3761, April 2004 (TXT).
[RFC4769]	Livingood, J. and R. Shockey, "IANA Registration for an Enumservice Containing Public Switched Telephone Network (PSTN) Signaling Information," RFC 4769, November 2006 (TXT).
[I-D.ietf-enum-cnam]	Shockey, R., "IANA Registration for an Enumservice Calling Name Delivery (CNAM) Information and IANA Registration for URI type 'pstndata'," draft-ietf-enum-cnam-08 (work in progress), September 2008 (TXT).

Appendix A. Request History Requirements

TOC

The following list constitutes a set of requirements for a "Request History" capability.

1. CAPABILITY-req: The "Request History" capability provides a capability to inform proxies and UAs involved in processing a request about the history/progress of that request. Although this is inherently provided when the retarget is in response to a SIP redirect, it is deemed useful for non-redirect retargeting scenarios, as well.

2. GENERATION-req: "Request History" information is generated when the request is retargeted.

   A. In some scenarios, it might be possible for more than one instance of retargeting to occur within the same Proxy. A proxy should also generate Request History information for the 'internal retargeting'.

   B. An entity (UA or proxy) retargeting in response to a redirect or REFER should include any Request History information from the redirect/REFER in the new request.

3. ISSUER-req: "Request History" information can be generated by a UA or proxy. It can be passed in both requests and responses.

4. CONTENT-req: The "Request History" information for each occurrence of retargeting shall include the following:

   A. The new URI or address to which the request is in the process of being retargeted,

   B. The URI or address from which the request was retargeted, and wether the retarget URI was an AOR

   C. The mechanism by which the new URI or address was determined,

   D. The reason for the Request-URI or address modification,

   E. Chronological ordering of the Request History information.

5. REQUEST-VALIDITY-req: Request History is applicable to requests not sent within an established dialog (e.g., INVITE, REGISTER, MESSAGE, and OPTIONS).

6. BACKWARDS-req: Request History information may be passed from the generating entity backwards towards the UAC. This is needed to enable services that inform the calling party about the dialog establishment attempts.

7. FORWARDS-req: Request History information may also be included by the generating entity in the request, if it is forwarded onwards.

TOC

A.1. Security Requirements

The Request History information is being inserted by a network element retargeting a Request, resulting in a slightly different problem than the basic SIP header problem, thus requiring specific consideration. It is recognized that these security requirements can be generalized to a basic requirement of being able to secure information that is inserted by proxies.
The potential security problems include the following:

1. A rogue application could insert a bogus Request History entry either by adding an additional entry as a result of retargeting or entering invalid information.

2. A rogue application could re-arrange the Request History information to change the nature of the end application or to mislead the receiver of the information.

3. A rogue application could delete some or all of the Request History information.

Thus, a security solution for "Request History" must meet the following requirements:

1. SEC-req-1: The entity receiving the Request History must be able to determine whether any of the previously added Request History content has been altered.

2. SEC-req-2: The ordering of the Request History information must be preserved at each instance of retargeting.

3. SEC-req-3: The entity receiving the information conveyed by the Request History must be able to authenticate the entity providing the request.

4. SEC-req-4: To ensure the confidentiality of the Request History information, only entities that process the request should have visibility to the information.

It should be noted that these security requirements apply to any entity making use of the Request History information.

TOC

A.2. Privacy Requirements

Since the Request-URI that is captured could inadvertently reveal information about the originator, there are general privacy requirements that MUST be met:

1. PRIV-req-1: The entity retargeting the Request must ensure that it maintains the network-provided privacy (as described in [RFC3323] (Peterson, J., “A Privacy Mechanism for the Session Initiation Protocol (SIP),” November 2002.)) associated with the Request as it is retargeted.

2. PRIV-req-2: The entity receiving the Request History must maintain the privacy associated with the information. In addition, local policy at a proxy may identify privacy requirements associated with the Request-URI being captured in the Request History information.

3. PRIV-req-3: Request History information subject to privacy shall not be included in ougoing messages unless it is protected as described in [RFC3323] (Peterson, J., “A Privacy Mechanism for the Session Initiation Protocol (SIP),” November 2002.).

Appendix B. Detailed call flows

TOC

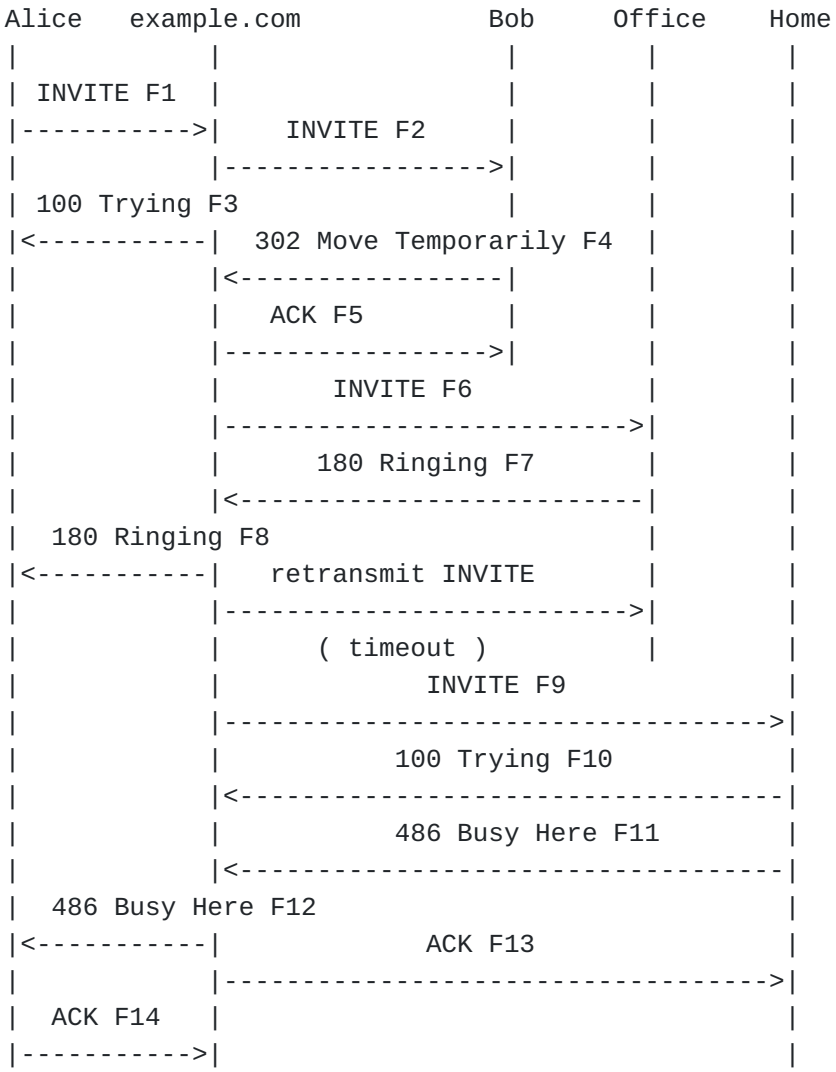
The scenarios in this section provide sample use cases for the History-Info header for informational purposes only. They are not intended to be normative.

B.1. Sequentially Forking (History-Info in Response)

TOC

This scenario highlights an example where the History-Info in the response is useful to an application or user that originated the request.
Alice sends a call to Bob via sip:example.com. The proxy sip:example.com sequentially tries Bob on a SIP UA that has bound a contact with the sip:bob@example.com AOR, and then several alternate addresses (Office and Home) unsuccessfully before sending a response to Alice. In this example, note that Office and Home are not the same AOR as sip:bob@example.com, but rather different AORs that have been configured as alternate addresses for Bob in the proxy. In other words, Office and Bob are not bound through SIP Registration with Bob's AOR.

This type of arrangement is common for example when a "routing" rule to a PSTN number is manually configured in a Proxy.
This scenario is provided to show that by providing the History-Info to Alice, the end-user or an application at Alice could make a decision on how best to attempt finding Bob. Without this mechanism, Alice might well attempt Office (and thus Home) and then re-attempt Home on a third manual attempt at reaching Bob. With this mechanism, either the end-user or application could know that Bob is not answering in the Office, and his busy on his home phone. If there were an alternative address for Bob known to this end-user or application, that hasn't been attempted, then either the application or the end- user could attempt that. The intent here is to highlight an example of the flexibility of this mechanism that enables applications well beyond SIP as it is certainly well beyond the scope of this document to prescribe detailed applications.



Message Details

F1 INVITE alice -> example.com

```
INVITE sip:alice@example.com SIP/2.0
Via: SIP/2.0/TCP 192.0.2.3:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>
Supported: histinfo
Call-Id: 12345600@example.com
CSeq: 1 INVITE
Contact: Alice <sip:alice@192.0.2.3>
Content-Type: application/sdp
Content-Length: <appropriate value>
<!-- SDP Not Shown -->
```

F2 INVITE example.com -> Bob

```
INVITE sip:bob@192.0.2.4 SIP/2.0
Via: SIP/2.0/TCP proxy.example.com:5060
Via: SIP/2.0/TCP 192.0.2.3:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>
Supported: histinfo
Call-Id: 12345600@example.com
CSeq: 1 INVITE
Record-Route: <sip:proxy.example.com;lr>
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.4>;index=1.1;rc
Contact: Alice <sip:alice@192.0.2.3>
Content-Type: application/sdp
Content-Length: <appropriate value>
<!-- SDP Not Shown -->
```

F3 100 Trying example.com -> alice

```
SIP/2.0 100 Trying
Via: SIP/2.0/TCP 192.0.2.3:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>
Call-Id: 12345600@example.com
CSeq: 1 INVITE
Content-Length: 0
```

F4 302 Moved Temporarily Bob -> example.com

```
SIP/2.0 302 Moved Temporarily
Via: SIP/2.0/TCP proxy.example.com:5060
Via: SIP/2.0/TCP 192.0.2.3:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>;tag=3
Call-Id: 12345600@example.com
CSeq: 1 INVITE
Record-Route: <sip:proxy.example.com;lr>
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.4?Reason=SIP;cause=302>;\
              index=1.1;rc
History-Info: <sip:office@example.com>;index=1.2
Contact: <sip:office@example.com>
Content-Length: 0
```

F5 ACK 192.0.2.4 -> Bob

```
ACK sip:home@example.com SIP/2.0
Via: SIP/2.0/TCP proxy.example.com:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>
Call-Id: 12345600@example.com
CSeq: 1 ACK
Content-Length: 0
```

F6 INVITE example.com -> office

```
INVITE sip:office@192.0.2.3.com SIP/2.0
Via: SIP/2.0/TCP proxy.example.com:5060;branch=2
Via: SIP/2.0/TCP 192.0.2.3:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>
Supported: histinfo
Call-Id: 12345600@example.com
Record-Route: <sip:proxy.example.com;lr>
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.4?Reason=SIP;cause=302>;\
              index=1.1;rc
History-Info: <sip:office@example.com>;index=1.2;mp=1
History-Info: <sip:office@192.0.2.5>;index=1.2.1
CSeq: 1 INVITE
Contact: Alice <sip:alice@192.0.2.3>
Content-Type: application/sdp
Content-Length: <appropriate value>
<!-- SDP Not Shown -->
```

F7 180 Ringing office -> example.com

```
SIP/2.0 180 Ringing
Via: SIP/2.0/TCP proxy.example.com:5060;branch=2
Via: SIP/2.0/TCP 192.0.2.3:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>;tag=5
Supported: histinfo
Call-ID: 12345600@example.com
Record-Route: <sip:proxy.example.com;lr>
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.4?Reason=SIP;cause=302>;\
              index=1.1;rc
History-Info: <sip:office@example.com>;index=1.2;mp=1
History-Info: <sip:office@192.0.2.5>;index=1.2.1
CSeq: 1 INVITE
Content-Length: 0
```

F8 180 Ringing example.com -> alice

```
SIP/2.0 180 Ringing
Via: SIP/2.0/TCP example.com:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>
Supported: histinfo
Call-Id: 12345600@example.com
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.4?Reason=SIP;cause=302>;\
              index=1.1;rc
History-Info: <sip:office@example.com>;index=1.2;mp=1
History-Info: <sip:office@192.0.2.5>;index=1.2.1
CSeq: 1 INVITE
Content-Length: 0
```

F9 INVITE example.com -> home

```
INVITE sip:home@192.0.2.6 SIP/2.0
Via: SIP/2.0/TCP proxy.example.com:5060;branch=3
Via: SIP/2.0/TCP 192.0.2.3:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>
Supported: histinfo
Call-Id: 12345600@example.com
Record-Route: <sip:proxy.example.com;lr>
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.4?Reason=SIP;cause=302>;\
              index=1.1;rc
History-Info: <sip:office@example.com>;index=1.2;mp=1
History-Info: <sip:office@192.0.2.5?Reason=SIP;cause=480>;\
              index=1.2.1>;index=1.2.1
History-Info: <sip:home@example.com>;index=1.3;mp=1.2
History-Info: <sip:home@192.0.2.6>;index=1.3.1
CSeq: 1 INVITE
Contact: Alice <sip:alice@192.0.2.3>
Content-Type: application/sdp
Content-Length: <appropriate value>
<!-- SDP Not Shown -->
```

F10 100 Trying home -> example.com

```
SIP/2.0 100 Trying
Via: SIP/2.0/TCP proxy.example.com:5060;branch=3
Via: SIP/2.0/TCP 192.0.2.3:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>
Call-Id: 12345600@example.com
CSeq: 1 INVITE
Content-Length: 0
```

F11 486 Busy Here home -> example.com

```
SIP/2.0  486 Busy Here
Via: SIP/2.0/TCP proxy.example.com:5060;branch=3
Via: SIP/2.0/TCP 192.0.2.3:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>
Call-Id: 12345600@example.com
Record-Route: <sip:proxy.example.com;lr>
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.4?Reason=SIP;cause=302>;\
              index=1.1;rc
History-Info: <sip:office@example.com>;index=1.2;mp=1
History-Info: <sip:office@192.0.2.5?Reason=SIP;cause=480>;\
              index=1.2.1>;index=1.2.1
History-Info: <sip:home@example.com>;index=1.3;mp=1.2
History-Info: <sip:home@192.0.2.6>;index=1.3.1
CSeq: 1 INVITE
Content-Length: 0
```

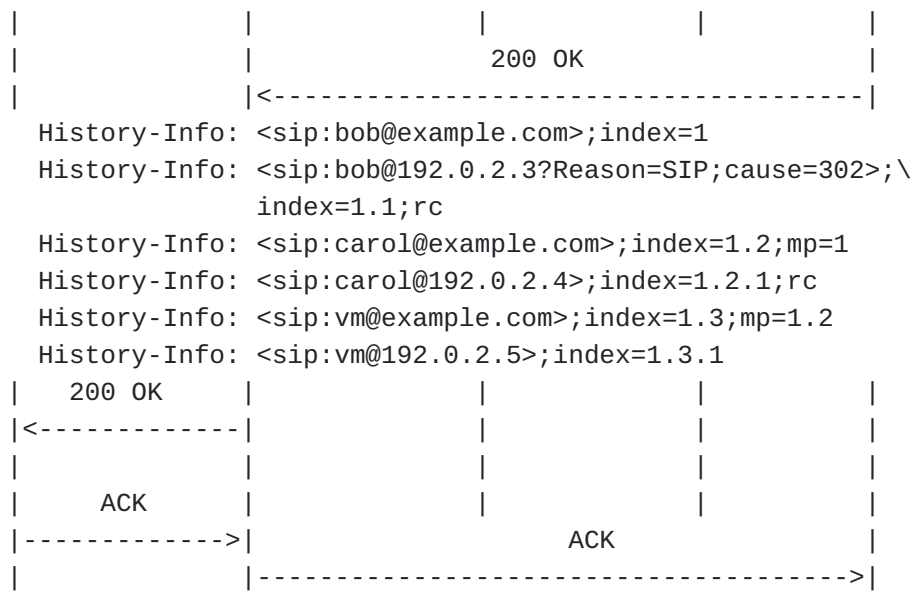
```
F12 486 Busy Here example.com -> alice

SIP/2.0  486 Busy Here
Via: SIP/2.0/TCP 192.0.2.3:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>
Call-Id: 12345600@example.com
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.4?Reason=SIP;cause=302>;\
              index=1.1;rc
History-Info: <sip:office@example.com>;index=1.2;mp=1
History-Info: <sip:office@192.0.2.5?Reason=SIP;cause=480>;\
              index=1.2.1>;index=1.2.1
History-Info: <sip:home@example.com>;index=1.3;mp=1.2
History-Info: <sip:home@192.0.2.6>;index=1.3.1
CSeq: 1 INVITE
Content-Length: 0



F13 ACK example.com -> home

ACK sip:home@example.com SIP/2.0
Via: SIP/2.0/TCP proxy.example.com:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>
Call-Id: 12345600@example.com
CSeq: 1 ACK
Content-Length: 0



F14 ACK alice -> example.com

ACK sip:bob@example.com SIP/2.0
Via: SIP/2.0/TCP 192.0.2.3:5060
From: Alice <sip:alice@example.com>
To: Bob <sip:bob@example.com>
Call-Id: 12345600@example.com
Route: <sip:proxy.example.com;lr>
CSeq: 1 ACK
Content-Length: 0
```

TOC

B.2. Voicemail

This scenario highlights an example where the History-Info in the request is primarily of use by an edge service (e.g., voicemail server). It should be noted that this is not intended to be a complete specification for this specific edge service as it is quite likely that additional information is needed by the edge service. History-Info is just one building block that this service can use.
Alice called Bob, which had been forwarded to Carol, which forwarded to VM (voicemail server). Based upon the retargeted URIs and Reasons (and other information) in the INVITE, the VM server makes a policy decision about what mailbox to use, which greeting to play, etc.


Alice
example.com
Bob
Carol
VM
INVITE sip:bob@example.com
INVITE sip:bob@192.0.2.3
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.3>;index=1.1;rc
100 Trying
302 Moved Temporarily
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.3?Reason=SIP;cause=302>;\
index=1.1;rc
History-Info: <sip:carol@example.com>;index=1.2
INVITE sip:Carol@192.0.2.4
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.3?Reason=SIP;cause=302>;\
index=1.1;rc
History-Info: <sip:carol@example.com>;index=1.2;mp=1
History-Info: <sip:carol@192.0.2.4>;index=1.2.1;rc
180 Ringing
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.3?Reason=SIP;cause=302>;\
index=1.1;rc
History-Info: <sip:carol@example.com>;index=1.2;mp=1
History-Info: <sip:carol@192.0.2.4>;index=1.2.1;rc
180 Ringing
. . .
(timeout)
INVITE sip:vm@192.0.2.5
History-Info: <sip:bob@example.com>;index=1
History-Info: <sip:bob@192.0.2.3?Reason=SIP;cause=302>;\
index=1.1;rc
History-Info: <sip:carol@example.com>;index=1.2;mp=1
History-Info: <sip:carol@192.0.2.4>;index=1.2.1;rc
History-Info: <sip:vm@example.com>;index=1.3;mp=1.2
History-Info: <sip:vm@192.0.2.5>;index=1.3.1



B.3. Automatic Call Distribution

TOC

This scenario highlights an example of an Automatic Call Distribution service, where the agents are divided into groups based upon the type of customers they handle. In this example, the Gold customers are given higher priority than Silver customers, so a Gold call would get serviced even if all the agents servicing the Gold group were busy, by retargeting the request to the Silver Group for delivery to an agent. Upon receipt of the call at the agent assigned to handle the incoming call, based upon the History-Info header in the message, the application at the agent can provide an indication that this is a Gold call, from how many groups it might have overflowed before reaching the agent, etc. and thus can be handled appropriately by the agent.
For scenarios whereby calls might overflow from the Silver to the Gold, clearly the alternate group identification, internal routing, or actual agent that handles the call should not be sent to UA1. Thus, for this scenario, one would expect that the Proxy would not support the sending of the History-Info in the response, even if requested by Alice.
As with the other examples, this is not prescriptive of how one would do this type of service but an example of a subset of processing that might be associated with such a service. In addition, this example is not addressing any aspects of Agent availability, which might also be done via a SIP interface.


Alice
example.com
Gold
Silver
Agent
INVITE sip:Gold@example.com
Supported: histinfo
INVITE sip:Gold@example.com
History-Info: <sip:Gold@example.com>;index=1
History-Info: <sip:Gold@gold.example.com>;index=1.1
302 Moved Temporarily
History-Info: <sip:Gold@example.com>;index=1
History-Info: <sip:Gold@gold.example.com?Reason=SIP;cause=302>;\
index=1.1
Contact: <sip:Silver@example.com>
INVITE sip:Silver@example.com
History-Info: <sip:Gold@example.com>;index=1
History-Info: <sip:Gold@gold.example.com?Reason=SIP;cause=302>;\
index=1.1
History-Info: <sip:Silver@example.com>;index=2;mp=1
History-Info: <sip:Silver@silver.example.com>;index=2.1
INVITE sip:Silver@192.0.2.7
History-Info: <sip:Gold@example.com>;index=1
History-Info: <sip:Gold@gold.example.com?Reason=SIP;cause=302>;\
index=1.1
History-Info: <sip:Silver@example.com>;index=2;mp=1
History-Info: <sip:Silver@silver.example.com>;index=2.1
History-Info: <sip:Silver@192.0.2.7>;index=2.1.1;rc
200 OK
History-Info: <sip:Gold@example.com>;index=1
History-Info: <sip:Gold@gold.example.com?Reason=SIP;cause=302>;\
index=1.1
History-Info: <sip:Silver@example.com>;index=2;mp=1
History-Info: <sip:Silver@silver.example.com>;index=2.1
History-Info: <sip:Silver@192.0.2.7>;index=2.1.1;rc
200 OK
History-Info: <sip:Gold@example.com>;index=1

```
  History-Info: <sip:Gold@gold.example.com?Reason=SIP;cause=302>;\
                index=1.1
  History-Info: <sip:Silver@example.com>;index=2;mp=1
  History-Info: <sip:Silver@silver.example.com>;index=2.1
  History-Info: <sip:Silver@192.0.2.7>;index=2.1.1;rc
|              |              |             |            |
   200 OK      |              |             |            |
|<-------------|              |             |            |
|              |              |             |            |
|    ACK       |              |             |            |
|------------->|                  ACK                    |
|              |-----------------------------------------|>|
```

B.4. History-Info with Privacy Header

TOC

This example provides a basic call scenario such as the one in Figure 1 (Basic Call) but without forking, with sip:biloxi.example.com adding the Privacy header indicating that the History-Info header information is anonymized outside the biloxi.example.com domain. This scenario highlights the potential functionality lost with the use of "history" privacy in the Privacy header for the entire request and the need for careful consideration on the use of privacy for History-Info.

Figure 2: Example with Privacy Header

TOC

B.5. Privacy Header for a Specific History-Info Entry

This example also provides a basic call scenario such as the one in Figure 1 (Basic Call) but without forking, however, due to local policy at sip:biloxi.example.com, only the final hi-entry in the History-Info, which is Bob's local URI, contains a priv-value of "history", thus providing Alice with some information about the history of the request, but anonymizing Bob's local URI.

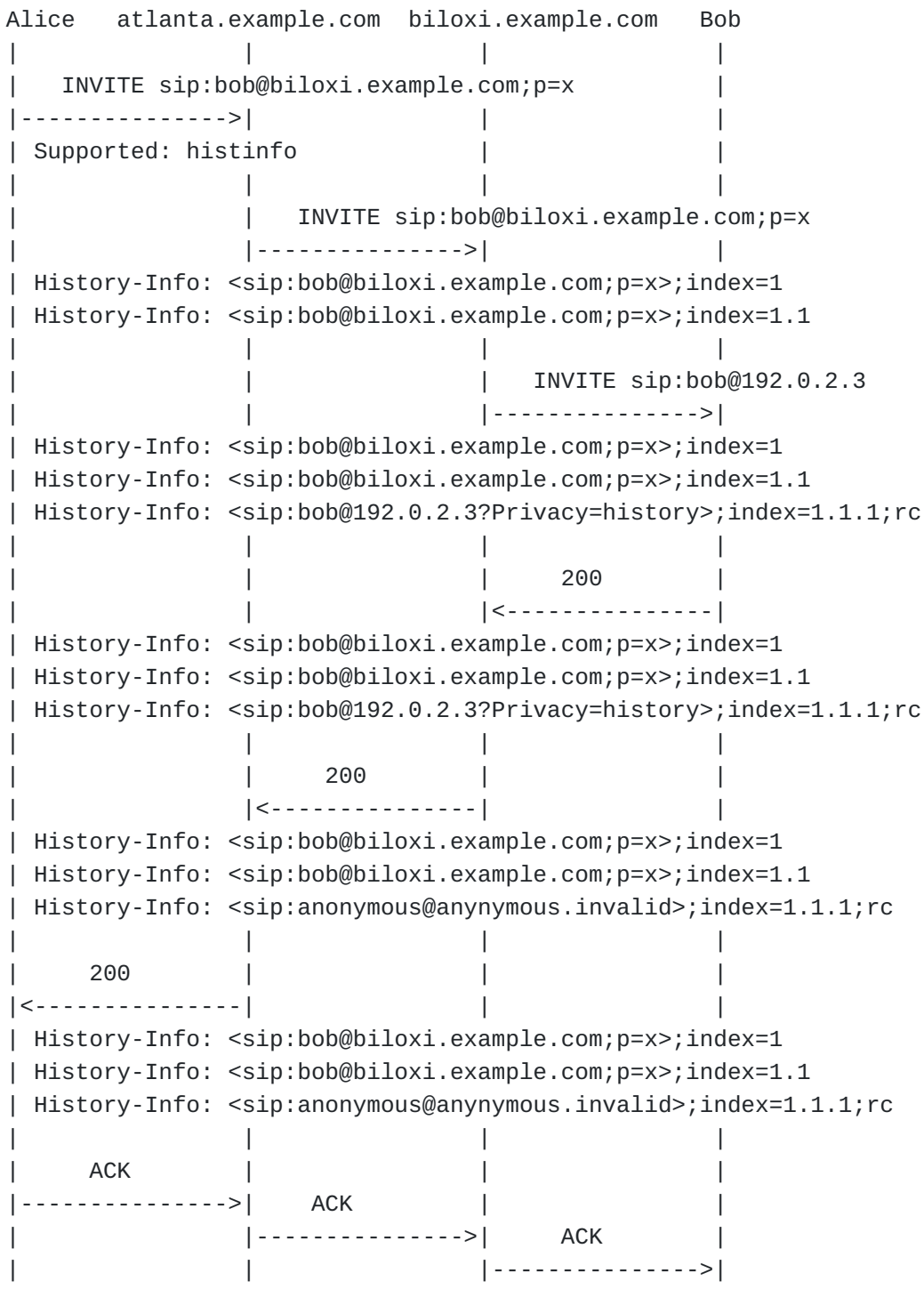


Figure 3: Example with Privacy Header for Specific URI

TOC

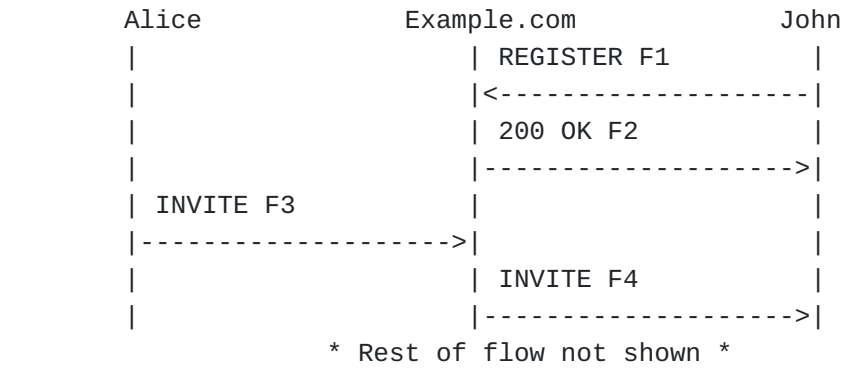
B.6. Determining the Alias used.

SIP user agents are associated with an address-of-record (AOR). It is possible for a single UA to actually have multiple AOR associated with it. One common usage for this is aliases. For example, a user might have an AOR of sip:john@example.com but also have the AORs sip:john.smith@example.com and sip:jsmith@example.com. Rather than registering against each of these AORs individually, the user would register against just one of them, and the home proxy would automatically accept incoming calls for any of the aliases, treating them identically and ultimately forwarding them towards the UA. This is common practice in the Internet Multimedia Subsystem (IMS), where it is called implicit registrations and each alias is called a public identity.

It is a common requirement for a UAS, on receipt of a call, to know which of its aliases was used to reach it. This knowledge can be used to choose ringtones to play, determine call treatment, and so on. For example, a user might give out one alias to friends and family only, resulting in a special ring that alerts the user to the importance of the call.

Following call-flow and example messages show how History-Info can be used to find out the alias used to reach the callee.

UAS can see which alias was used in the call by looking at the hi-entry prior to the last hi-entry with the "rc" tag.



F1 REGISTER John -> Example.com

```
REGISTER sip:example.com SIP/2.0
Via: SIP/2.0/UDP 192.0.2.1;branch=z9hG4bKnashds7
Max-Forwards: 70
From: John <sip:john@example.com>;tag=a73kszlfl
To: John <sip:john@example.com>
Call-ID: 1j9FpLxk3uxtm8tn@192.0.2.1
CSeq: 1 REGISTER
Contact: <sip:john@192.0.2.1>
Content-Length: 0
```

F2 200 OK Example.com -> John

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.0.2.1;branch=z9hG4bKnashds7
From: John <sip:john@example.com>;tag=a73kszlfl
To: John <sip:john@example.com>
Call-ID: 1j9FpLxk3uxtm8tn@192.0.2.1
CSeq: 1 REGISTER
Contact: <sip:john@192.0.2.1>;expires=3600
Content-Length: 0
```

F3 INVITE Alice -> Example.com

```
INVITE sip:john.smith@example.com SIP/2.0
Via: SIP/2.0/TCP 192.0.2.3:5060;branch=232sxxeserg
From: Alice <sip:alice@example.com>
To: John <sip:john.smith@example.com>
Supported: histinfo
Call-Id: 12345600@example.com
CSeq: 1 INVITE
History-Info: <sip:john.smith@example.com>;index=1;
Contact: Alice <sip:alice@192.0.2.3>
Content-Type: application/sdp
Content-Length: <appropriate value>
```

```
[SDP Not Shown]

F4 INVITE Example.com -> Bob

INVITE sip:john@192.0.2.1 SIP/2.0
Via: SIP/2.0/TCP proxy.example.com:5060;branch=as2334se
Via: SIP/2.0/TCP 192.0.2.3:5060;branch=232sxxeserg
From: Alice <sip:alice@example.com>
To: John <sip:john.smith@example.com>
Supported: histinfo
Call-Id: 12345600@example.com
CSeq: 1 INVITE
Record-Route: <sip:proxy.example.com;lr>
History-Info: <sip:john.smith@example.com>;index=1;
History-Info: <sip:john@192.0.2.1>;index=1.1;rc
Contact: Alice <sip:alice@192.0.2.3>
Content-Type: application/sdp
Content-Length: <appropriate value>

[SDP Not Shown]
```

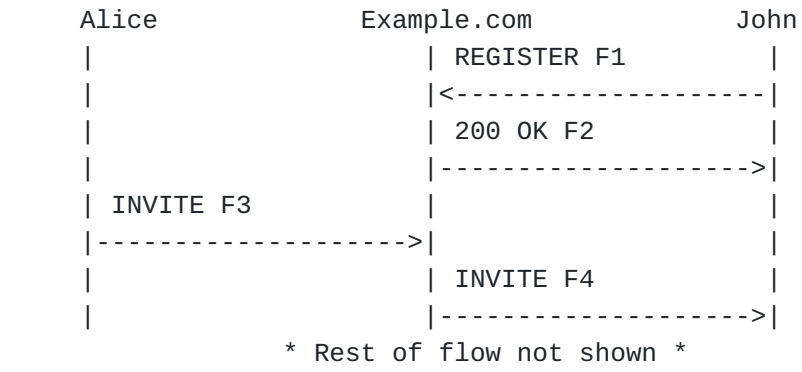
Figure 4: Alias Example

B.7. GRUU

TOC

A variation on the problem in Appendix B.6 (Determining the Alias used.) occurs with Globally Routable User Agent URI (GRUU) [RFC5627] (Rosenberg, J., "Obtaining and Using Globally Routable User Agent URIs (GRUUs) in the Session Initiation Protocol (SIP)," October 2009.). A GRUU is a URI assigned to a UA instance which has many of the same properties as the AOR, but causes requests to be routed only to that specific instance. It is desirable for a UA to know whether it was reached because a correspondent sent a request to its GRUU or to its AOR. This can be used to drive differing authorization policies on whether the request should be accepted or rejected, for example. However, like the AOR itself, the GRUU is lost in translation at the home proxy. Thus, the UAS cannot know whether it was contacted via the GRUU or its AOR.
Following call-flow and example messages show how History-Info can be used to find out the GRUU used to reach the callee.
GRUU is merely an AoR with a URI parameter that distinguishes the target instance, and as any URI parameters are preserved in history-info as Request-URI is trasnlated, UA can see if the request was

addressed to a specific instance (gruu) by evaluating the presence of "gr" parameter in the hi-entry prior to the last hi-entry with the "rc" tag.



F1 REGISTER John -> Example.com

```
REGISTER sip:example.com SIP/2.0
Via: SIP/2.0/UDP 192.0.2.1;branch=z9hG4bKnashds7
Max-Forwards: 70
From: John <sip:John@example.com>;tag=a73kszlfl
Supported: gruu
To: John <sip:john@example.com>
Call-ID: 1j9FpLxk3uxtm8tn@192.0.2.1
CSeq: 1 REGISTER
Contact: <sip:john@192.0.2.1>
 ;+sip.instance="<urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6>"
Content-Length: 0
```

F2 200 OK Example.com -> John

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.0.2.1;branch=z9hG4bKnashds7
From: John <sip:john@example.com>;tag=a73kszlfl
To: John <sip:john@example.com> ;tag=b88sn
Call-ID: 1j9FpLxk3uxtm8tn@192.0.2.1
CSeq: 1 REGISTER
Contact: <sip:john@192.0.2.1>
 ;pub-gruu="sip:john@example.com
 ;gr=urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6"
 ;temp-gruu=
  "sip:tgruu.7hs==jd7vnzga5w7fajsc7-ajd6fabz0f8g5@example.com;gr"
 ;+sip.instance="<urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6>"
 ;expires=3600
Content-Length: 0
```

Assuming Alice has a knowledge of a gruu either through
prior communication or through other means such as presence
places a call to John's gruu.

F3 INVITE Alice -> Example.com

```
INVITE sip:john@example.com
```

```
 ;gr=urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6 SIP/2.0
Via: SIP/2.0/TCP  192.0.2.3:5060;branch=232sxxeserg
From: Alice <sip:alice@example.com>;tag=kkaz-
To: <sip:john@example.com
 ;gr=urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6>
Supported: gruu, histinfo
Call-Id: 12345600@example.com
CSeq: 1 INVITE
History-Info: <sip:john@example.com
 ;gr=urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6>;index=1
Contact: Alice <sip:alice@192.0.2.3>
Content-Length: <appropriate value>

F4 INVITE Example.com -> John

INVITE sip:john@192.0.2.1 SIP/2.0
Via: SIP/2.0/TCP proxy.example.com:5060;branch=as2334se
Via: SIP/2.0/TCP 192.0.2.3:5060;branch=232sxxeserg
From: Alice <sip:alice@example.com>;tag=kkaz-
To: John <sip:john@example.com>
Supported: gruu, histinfo
Call-Id: 12345600@example.com
CSeq: 1 INVITE
Record-Route: <sip:proxy.example.com;lr>
History-Info: <sip:john@example.com
 ;gr=urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6>;index=1
History-Info: <sip:john@192.0.2.1>;index=1.1;rc
Contact: Alice <sip:alice@192.0.2.3>
Content-Type: application/sdp
Content-Length: <appropriate value>
```

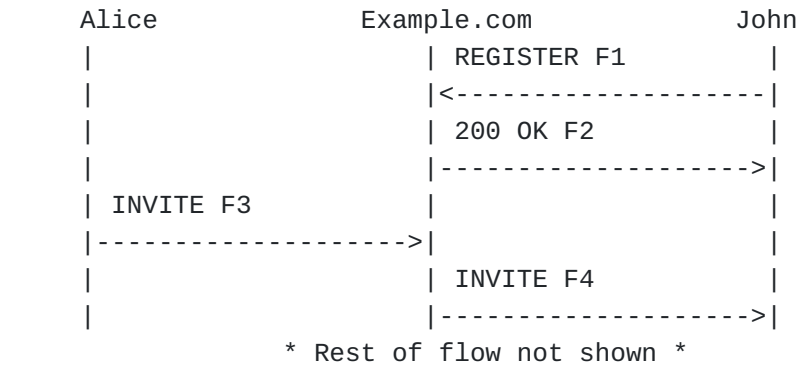
Figure 5: GRUU Example

B.8. Limited Use Address

TOC

A limited use address is a SIP URI that is minted on-demand, and passed out to a small number (usually one) remote correspondent. Incoming calls targeted to that limited use address are accepted as long as the UA still desires communications from the remote target. Should they no longer wish to be bothered by that remote correspondent, the URI is invalidated so that future requests targeted to it are rejected. Limited use addresses are used in battling voice spam [RFC5039] (Rosenberg, J. and C. Jennings, “The Session Initiation Protocol (SIP)

and Spam," January 2008.). The easiest way to provide them would be for a UA to be able to take its AOR, and "mint" a limited use address by appending additional parameters to the URI. It could then give out the URI to a particular correspondent, and remember that URI locally. When an incoming call arrives, the UAS would examine the parameter in the URI and determine whether or not the call should be accepted. Alternatively, the UA could push authorization rules into the network, so that it need not even see incoming requests that are to be rejected. This approach, especially when executed on the UA, requires that parameters attached to the AOR, but not used by the home proxy in processing the request, will survive the translation at the home proxy and be presented to the UA. This will not be the case with the logic in RFC 3261, since the Request-URI is replaced by the registered contact, and any such parameters are lost.
Using the history-info John's UA can easily see if the call was addressed to its AoR, GRUU or a temp-gruu and treat the call accordingly by looking at the hi-entry prior to the last hi-entry with the "rc" tag.



F1 REGISTER John -> Example.com

```
REGISTER sip:example.com SIP/2.0
Via: SIP/2.0/UDP 192.0.2.1;branch=z9hG4bKnashds7
Max-Forwards: 70
From: John <sip:John@example.com>;tag=a73kszlfl
Supported: gruu
To: John <sip:john@example.com>
Call-ID: 1j9FpLxk3uxtm8tn@192.0.2.1
CSeq: 1 REGISTER
Contact: <sip:john@192.0.2.1>
 ;+sip.instance="<urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6>"
Content-Length: 0
```

F2 200 OK Example.com -> John

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.0.2.1;branch=z9hG4bKnashds7
From: John <sip:john@example.com>;tag=a73kszlfl
To: John <sip:john@example.com> ;tag=b88sn
Call-ID: 1j9FpLxk3uxtm8tn@192.0.2.1
CSeq: 1 REGISTER
Contact: <sip:john@192.0.2.1>
 ;pub-gruu="sip:john@example.com
 ;gr=urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6"
 ;temp-gruu=
  "sip:tgruu.7hs==jd7vnzga5w7fajsc7-ajd6fabz0f8g5@example.com;gr"
 ;+sip.instance="<urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6>"
 ;expires=3600
Content-Length: 0
```

Assuming Alice has a knowledge of a temp-gruu, she places a call to the temp-gruu.

F3 INVITE Alice -> Example.com

```
INVITE sip:tgruu.7hs==jd7vnzga5w7fajsc7-ajd6fabz0f8g5@example.com
```

```
 ;gr SIP/2.0
Via: SIP/2.0/TCP 192.0.2.3:5060;branch=232sxxeserg
From: Alice <sip:alice@example.com>;tag=kkaz-
To: <sip:sip:tgruu.7hs==jd7vnzga5w7fajsc7-ajd6fabz0f8g5@example.com
 ;gr>
Supported: gruu, histinfo
Call-Id: 12345600@example.com
CSeq: 1 INVITE
History-Info:
 <sip:tgruu.7hs==jd7vnzga5w7fajsc7-ajd6fabz0f8g5@example.com;gr>
 ;index=1
Contact: Alice <sip:alice@192.0.2.3>
Content-Length: <appropriate value>

F4 INVITE Example.com -> John

INVITE sip:john@192.0.2.1 SIP/2.0
Via: SIP/2.0/TCP proxy.example.com:5060;branch=as2334se
Via: SIP/2.0/TCP 192.0.2.3:5060;branch=232sxxeserg
From: Alice <sip:alice@example.com>;tag=kkaz-
To: John <sip:john@example.com>
Supported: gruu, histinfo
Call-Id: 12345600@example.com
CSeq: 1 INVITE
Record-Route: <sip:proxy.example.com;lr>
History-Info:
 <sip:tgruu.7hs==jd7vnzga5w7fajsc7-ajd6fabz0f8g5@example.com;gr>
 ;index=1
History-Info: <sip:john@192.0.2.1>;index=1.1;rc
Contact: Alice <sip:alice@192.0.2.3>
Content-Type: application/sdp
Content-Length: <appropriate value>
```

Figure 6: Limited Use Address Example

B.9. Sub-Address

TOC

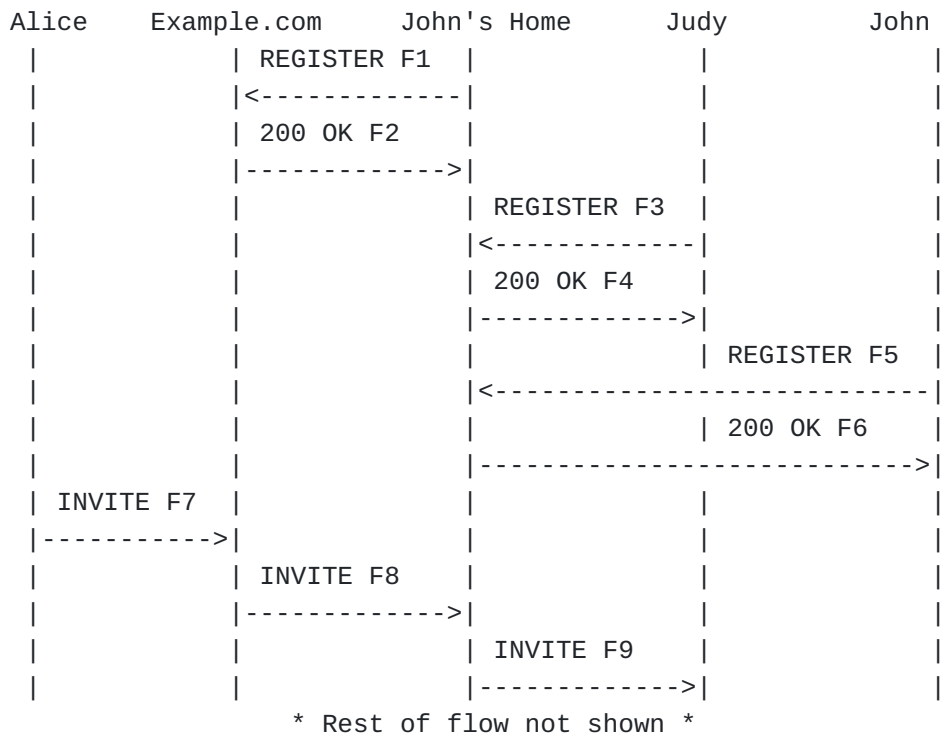
Sub-Addressing is very similar to limited use addresses. Sub-addresses are addresses within a subdomain that are multiplexed into a single address within a parent domain. The concept is best illustrated by example. Consider a VoIP service provided to consumers. A consumer obtains a single address from its provider, say sip:family@example.com. However, Joe is the patriarch of a family with four members, and would

like to be able to have a separate identifier for each member of his family. One way to do that, without requiring Joe to purchase new addresses for each member from the provider, is for Joe to mint additional URI by adding a parameter to the AOR. For example, his wife Judy with have the URI sip:family@example.com;member=judy, and Joe himself would have the URI sip:family@example.com;member=joe. The SIP server provider would receive requests to these URI, and ignoring the unknown parameters (as required by [RFC3261] (Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, “SIP: Session Initiation Protocol,” June 2002.)) route the request to the registered contact, which corresponds to a SIP server in Joes home. That server, in turn, can examine the URI parameters and determine which phone in the home to route the call to.

This feature is not specific to VoIP, and has existing in Integrated Services Digital Networking (ISDN) for some time. It is particularly useful for small enterprises, in addition to families. It is also similar in spirit (though not mechanism) to the ubiquitous home routers used by consumers, which allow multiple computers in the home to "hide" behind the single IP address provided by the service provider, by using the TCP and UDP port as a sub-address.

The sub-addressing feature is not currently feasible in SIP because of the fact that any SIP URI parameter used to convey the sub-address would be lost at the home proxy, due to the fact that the Request-URI is rewritten there.

Call-flow and example messages below show the how History-Info can be used to deliver the sub-address. UAS or Proxy can determine the sub-address by looking at the hi-entry prior to the last hi-entry with the "rc" tag.



F1 REGISTER John's Home Server -> Example.com

```
REGISTER sip:example.com SIP/2.0
Via: SIP/2.0/UDP 192.0.2.1;branch=z9hG4bKnashds7
Max-Forwards: 70
From: John <sip:johnhome@example.com>;tag=a73kszlfl
To: John <sip:johnhome@example.com>
Call-ID: 1j9FpLxk3uxtm8tn@192.0.2.1
CSeq: 1 REGISTER
Contact: <sip:johnhome@192.0.2.1>
Content-Length: 0
```

F2 200 OK Example.com -> John's Home Server

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.0.2.1;branch=z9hG4bKnashds7
From: John <sip:johnhome@example.com>;tag=a73kszlfl
To: John <sip:johnhome@example.com> ;tag=b88sn
Call-ID: 1j9FpLxk3uxtm8tn@192.0.2.1
CSeq: 1 REGISTER
Contact: <sip:johnhome@192.0.2.1>;expires=3600
Content-Length: 0
```

We assume that John's server acts as a proxy allowing
each of the device in the house to register.

F3 REGISTER Judy's phone -> John's Home Server

```
REGISTER sip:192.168.1.1 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.2;branch=z9hG4bKnasdds
Max-Forwards: 70
From: Judy <sip:judy@192.168.1.1>;tag=a73kszlfl
To: Judy <sip:judy@192.168.1.1>
Call-ID: 12345pLxk3uxtm8tn@192.168.1.2
CSeq: 1 REGISTER
Contact: <sip:judy@192.168.1.2>
Content-Length: 0

F4 200 OK John's Home Server -> Judy's phone

SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.168.1.2;branch=z9hG4bKnashds7
From: Judy <sip:judy@192.168.1.1>;tag=a73kszlfl
To: Judy <sip:judy@192.168.1.1>tag=b88sn
Call-ID: 12345pLxk3uxtm8tn@192.168.1.2
CSeq: 1 REGISTER
Contact: <sip:judy@192.168.1.2>;expires=3600
Content-Length: 0

F5 REGISTER John's phone -> John's Home Server

REGISTER sip:192.168.1.1 SIP/2.0
Via: SIP/2.0/UDP 192.168.1.3;branch=z9hG4bKnasdds
Max-Forwards: 70
From: Judy <sip:john@192.168.1.1>;tag=a73kszlfl
To: Judy <sip:john@192.168.1.1>
Call-ID: 12346pLxk3uxtm8tn@192.168.1.3
CSeq: 1 REGISTER
Contact: <sip:john@192.168.1.3>
Content-Length: 0

F6 200 OK John's Home Server -> John's phone

SIP/2.0 200 OK
Via: SIP/2.0/UDP 192.168.1.3;branch=z9hG4bKnashds7
From: John <sip:john@192.168.1.1>;tag=a73kszlfl
To: John <sip:john@192.168.1.1> ;tag=b88sn
Call-ID: 12346pLxk3uxtm8tn@192.168.1.3
CSeq: 1 REGISTER
Contact: <sip:john@192.168.1.3>;expires=3600
Content-Length: 0

F7 INVITE Alice -> Example.com

INVITE sip:johnhome@example.com;member=judy SIP/2.0
Via: SIP/2.0/TCP 192.0.2.3:5060;branch=232sxxeserg
From: Alice <sip:alice@example.com>
```

```
To: Judy <sip:johnhome@example.com;member=judy>
Supported: histinfo
Call-Id: 12345600@example.com
CSeq: 1 INVITE
History-Info: <sip:johnhome@example.com;member=judy>;index=1;
Contact: Alice <sip:alice@192.0.2.3>
Content-Type: application/sdp
Content-Length: <appropriate value>

[SDP Not Shown]
```

F8 INVITE Example.com -> John's Home

```
INVITE sip:johnhome@192.0.2.1 SIP/2.0
Via: SIP/2.0/TCP proxy.example.com:5060;branch=as2334se
Via: SIP/2.0/TCP 192.0.2.3:5060;branch=232sxxeserg
From: Alice <sip:alice@example.com>
To: Judy <sip:johnhome@example.com;member=judy>
Supported: histinfo
Call-Id: 12345600@example.com
CSeq: 1 INVITE
Record-Route: <sip:proxy.example.com;lr>
History-Info: <sip:johnhome@example.com;member=judy>;index=1;
History-Info: <sip:john@192.0.2.1>;index=1.1;rc
Contact: Alice <sip:alice@192.0.2.3>
Content-Type: application/sdp
Content-Length: <appropriate value>

[SDP Not Shown]
```

John's Home server can see that the call was addressed to Judy by evaluating the entry prior to the last entry with the "rc" tag and forwards the call accordingly.

F9 INVITE John's Home -> Judy

```
INVITE sip:judy@192.168.1.2 SIP/2.0
Via: SIP/2.0/TCP 192.168.1.1:5060;branch=abc2334se
Via: SIP/2.0/TCP proxy.example.com:5060;branch=as2334se
Via: SIP/2.0/TCP 192.0.2.3:5060;branch=232sxxeserg
From: Alice <sip:alice@example.com>
To: Judy <sip:johnhome@example.com;member=judy>
Supported: histinfo
Call-Id: 12345600@example.com
CSeq: 1 INVITE
Record-Route: <sip:proxy.example.com;lr>
History-Info: <sip:johnhome@example.com;member=judy>;index=1;
History-Info: <sip:john@192.0.2.1>;index=1.1;rc
History-Info: <sip:judy@192.168.1.1>;index=1.1.1;mp=1.1
```

```
History-Info: <sip:judy@192.168.1.2>;index=1.1.1.1;rc
Contact: Alice <sip:alice@192.0.2.3>
Content-Type: application/sdp
Content-Length: <appropriate value>

[SDP Not Shown]
```

Figure 7: Sub-Address Example

B.10. Service Invocation

TOC

Several SIP specifications have been developed which make use of complex URIs to address services within the network rather than subscribers. The URIs are complex because they contain numerous parameters that control the behavior of the service. Examples of this include the specification which first introduced the concept, [RFC3087] (Campbell, B. and R. Sparks, "Control of Service Context using SIP Request-URI," April 2001.), control of network announcements and IVR with SIP URI [RFC4240] (Burger, E., Van Dyke, J., and A. Spitzer, "Basic Network Media Services with SIP," December 2005.), and control of voicemail access with SIP URI [RFC4458] (Jennings, C., Audet, F., and J. Elwell, "Session Initiation Protocol (SIP) URIs for Applications such as Voicemail and Interactive Voice Response (IVR)," April 2006.).
A common problem with all of these mechanisms is that once a proxy has decided to rewrite the Request-URI to point to the service, it cannot be sure that the Request-URI will not be destroyed by a downstream proxy which decides to forward the request in some way, and does so by rewriting the Request-URI.
Section on voicemail (Voicemail) shows how History-Info can be used to invocate a service.

B.11. Toll Free Number

TOC

Toll free numbers, also known as 800 or 8xx numbers in the United States, are telephone numbers that are free for users to call.
In the telephone network, toll free numbers are just aliases to actual numbers which are used for routing of the call. In order to process the call in the PSTN, a switch will perform a query (using a protocol called TCAP), which will return either a phone number or the identity of a carrier which can handle the call.

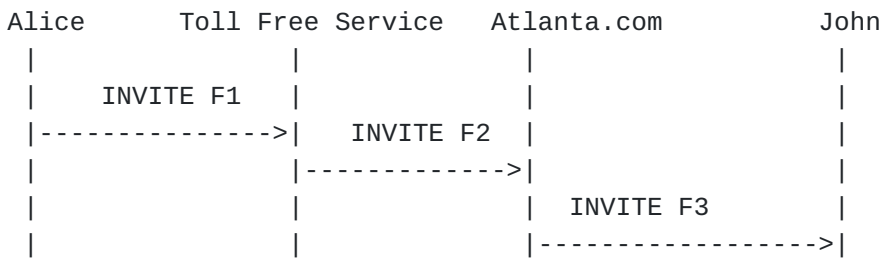
There has been recent work on allowing such PSTN translation services to be accessed by SIP proxy servers through IP querying mechanisms. ENUM, for example [RFC3761] (Faltstrom, P. and M. Mealling, “The E.164 to Uniform Resource Identifiers (URI) Dynamic Delegation Discovery System (DDDS) Application (ENUM),” April 2004.) has already been proposed as a mechanism for performing Local Number Portability (LNP) queries [RFC4769] (Livingood, J. and R. Shockey, “IANA Registration for an Enumservice Containing Public Switched Telephone Network (PSTN) Signaling Information,” November 2006.), and recently been proposed for performing calling name queries [I-D.ietf-enum-cnam] (Shockey, R., “IANA Registration for an Enumservice Calling Name Delivery (CNAM) Information and IANA Registration for URI type 'pstndata',” September 2008.). Using it for 8xx number translations is a logical next-step.

Once such a translation has been performed, the call needs to be routed towards the target of the request. Normally, this would happen by selecting a PSTN gateway which is a good route towards the translated number. However, one can imagine all-IP systems where the 8xx numbers are SIP endpoints on an IP network, in which case the translation of the 8xx number would actually be a SIP URI and not a phone number. Assuming for the moment it is a PSTN connected entity, the call would be routed towards a PSTN gateway. Proper treatment of the call in the PSTN (and in particular, correct reconciliation of billing records) requires that the call be marked with both the original 8xx number AND the target number for the call. However, in our example here, since the translation was performed by a SIP proxy upstream from the gateway, the original 8xx number would have been lost, and the call will not interwork properly with the PSTN.

Furthermore, even if the translation of the 8xx number was a SIP URI, the enterprise or user who utilize the 8xx service would like to know whether the call came in via 8xx number in order to treat the call differently (for example to play a special announcement..) but if the original R-URI is lost through translation, there is no way to tell if the call came in via 8xx number.

Similar problems arise with other "special" numbers and services used in the PSTN, such as operator services, pay numbers (9xx numbers in the U.S), and short service codes such as 311.

To find the service number, the UAS can look at the hi-entry prior to the first hi-entry with "mp" tag. Technically call can be forwarded to these "special" numbers from non "special" numbers, but with the way these services authorize trasnlation, it is not common.



F1: INVITE 192.0.2.1 -> proxy.example.com

```
INVITE sip:+18005551002@example.com;user=phone  SIP/2.0
Via: SIP/2.0/TCP 192.0.2.1:5060;branch=z9hG4bK-74bf9
From: Alice <sip:+15551001@example.com;user=phone>;tag=9fxced76sl
To: sip:+18005551002@example.com;user=phone
Call-ID: c3x842276298220188511
CSeq: 1 INVITE
Max-Forwards: 70
Supported: histinfo
History-Info: <sip:+18005551002@example.com;user=phone >;index=1
Contact: <sip:alice@192.0.2.1>
Content-Type: application/sdp
Content-Length: <appropriate value>

[SDP Not Shown]
```

F2: INVITE proxy.example.com -> atlanta.com

```
INVITE sip:+15555551002@atlanta.com SIP/2.0
Via: SIP/2.0/TCP 192.0.2.4:5060;branch=z9hG4bK-ik80k7g-1
Via: SIP/2.0/TCP 192.0.2.1:5060;branch=z9hG4bK-74bf9
From: Alice <sip:+15551001@example.com;user=phone>;tag=9fxced76sl
To: sip:+18005551002@example.com;user=phone
Call-ID: c3x842276298220188511
CSeq: 1 INVITE
Max-Forwards: 70
Supported: histinfo
History-Info: <sip:+18005551002@example.com;user=phone >;index=1,
              <sip:+15555551002@atlanta.com>;index=1.1;mp=1
Contact: <sip:alice@192.0.2.1>
Content-Type: application/sdp
Content-Length: <appropriate value>

[SDP Not Shown]
```

F3: INVITE atlanta.com -> Joe

```
INVITE sip:joe@192.168.1.2 SIP/2.0
```

```
Via: SIP/2.0/TCP 192.168.1.1:5060;branch=z9hG4bK-pxk7g-3
Via: SIP/2.0/TCP 192.0.2.4:5060;branch=z9hG4bK-ik80k7g-1
Via: SIP/2.0/TCP 192.0.2.1:5060;branch=z9hG4bK-74bf9
From: Alice <sip:+15551001@example.com;user=phone>;tag=9fxced76sl
To: sip:+18005551002@example.com;user=phone
Call-ID: c3x842276298220188511
CSeq: 1 INVITE
Max-Forwards: 70
Supported: histinfo
History-Info: <sip:+18005551002@example.com;user=phone >;index=1,
              <sip:+15555551002@atlanta.com>;index=1.1;mp=1,
              <sip:joe@atlanta.com>;index=1.1.1;mp=1.1,
              <sip:joe@192.168.1.2>;index=1.1.2;rc
Contact: <sip:alice@192.0.2.1>
Content-Type: application/sdp
Content-Length: <appropriate value>

[SDP Not Shown]
```

Figure 8: Service Number Example

Authors' Addresses

TOC

	Mary Barnes
	Nortel
	Richardson, TX
Email:	mary.barnes@nortel.com
	Francois Audet
	Skype Labs
Email:	francois.audet@skypelabs.com
	Shida Schubert
	NTT
Email:	shida@ntt.com
	Hans Erik van Elburg
	Detecon International Gmbh
	Oberkasseler str. 2
	Bonn, 53227
	Germany

Email:	ietf.hanserik@gmail.com
	Christer Holmberg
	Ericsson
	Hirsalantie 11, Jorvas
	Finland
Email:	christer.holmberg@ericsson.com