SIP WG
Internet-Draft
Obsoletes: RFC4244
(if approved)
Intended status: Standards Track
Expires: September 5, 2009

M. Barnes
F. Audet
Nortel

March 4, 2009

An Extension to the Session Initiation Protocol (SIP) for Request History Information
draft-barnes-sip-rfc4244bis-00.txt

Status of this Memo

This Internet-Draft is submitted to IETF in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF), its areas, and its working groups. Note that other groups may also distribute working documents as Internet-Drafts.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

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

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

This Internet-Draft will expire on September 5, 2009.

Copyright Notice

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 a new optional SIP header, History-Info, for capturing the history information in requests.

Table of Contents

1. Introduction

1.1. Overview

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 currently no standard mechanism within SIP for communicating the 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.

Section 1.3 provides additional background motivation for the Request History capability.

Section 2 identifies the requirements for a solution, with Section 3 providing an overall description of the solution.

Section 4 provides the details of the additions to the SIP protocol. Example uses of the new header are included in Section 4.5, with additional scenarios included in the Appendix A

Section 5 summarizes the application considerations identified in the previous sections.

Section 6 summarizes the security implications.

1.2. Conventions and Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

The term "retarget" is used in this document to refer to the process of a Proxy Server/User Agent Client (UAC) changing a Uniform Resource Identifier (URI) in a request based on a lookup in a location service

and thus changing the target of the request.

The term "forward" is used consistent with the terminology in [RFC3261]. However, it should be noted that 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, whether it's a "retargeted", "redirected" or "forwarded" request. Thus, the usage of "forward" in this document, other than in reference to the usage in section 16.6 of [RFC3261], refers to the request being forwarded to a next hop proxy.

The terms "location service" and "redirect" are used consistent with the terminology in [RFC3261].

1.3. Background: Why define a Generic Request History Header?

SIP implicitly provides retargeting, redirection and forwarding capabilities that enable calls to be routed to specific applications as defined in [RFC3261]. The term 'retarget' is used in this document to refer to the process of a Proxy Server/User Agent Client (UAC) changing a Uniform Resource Identifier (URI) in a request based on a lookup in a location service and thus changing the target of the request. The target(s) for a user can be created through registration or other means, which are outside the scope of this document and [RFC3261]. The rules for determining request targets as described in Section 16.5 of [RFC3261].

The motivation for capturing the request history is that in the process of retargeting and forwarding 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. It does not define any application-specific behavior for a Proxy or UA upon receipt of the information. Indeed, such behavior should be a local decision for the recipient application.

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, the details of which are for further study:

- 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) [I-D.ietf-sip-gruu], 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], control of network announcements and IVR with SIP URI [RFC4240], and control of voicemail access with SIP URI [RFC4458]
- 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.

2. Request History Requirements

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. OPTIONALITY-req: The "Request History" information is optional.
   A. In many cases, it is anticipated that whether the history is added to the Request would be a local policy decision enforced by the specific application; thus, no specific protocol element is needed.

   B. Due to the capability being "optional" from the SIP protocol perspective, the impact to an application of not having the "Request History" must be described. Applicability guidelines to be addressed by applications using this capability must be provided as part of the solution to these requirements.

3. GENERATION-req: "Request History" information is generated when the request is retargeted or forwarded (to a next hop proxy).
   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.
4. ISSUER-req: "Request History" information can be generated by a UA or proxy. It can be passed in both requests and responses.
5. CONTENT-req: The "Request History" information for each occurrence of retargeting or forwarding shall include the following:
   A. The new URI or address to which the request is in the process of being retargeted or forwarded,
   B. The URI or address from which the request was retargeted or forwarded,
   C. An indication as to whether the request was retargeted versus forwarded,
   D. The reason for the Request-URI or address modification,
   E. Chronological ordering of the Request History information.
6. REQUEST-VALIDITY-req: Request History is applicable to requests not sent within an established dialog (e.g., INVITE, REGISTER, MESSAGE, and OPTIONS).
7. 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.
8. FORWARDS-req: Request History information may also be included by the generating entity in the request, if it is forwarded onwards.

2.1. 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, either by retargeting and capturing the information, or as an application making use of the information received in either a Request or Response.

2.2. 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]) associated with the Request as it is retargeted or forwarded.
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 requirements shall not be included in outgoing messages unless it is protected as described in [RFC3323].

3. Request History Information Description

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). The solution is to capture the Request-URIs as a request is forwarded in a new header for SIP messages: History-Info (CONTENT-req). This allows for the capturing of the history of a request that would be lost with the normal SIP processing involved in the subsequent forwarding 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].

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) and any valid response to these requests (ISSUER-req).

The History-Info header is added to a Request when a new request is created by a UAC or forwarded by a Proxy, or when the target of a request is changed. The term "retarget" refers to this changing of the target of a request and the subsequent forwarding of that request. It should be noted that retargeting only occurs when the Request-URI indicates a domain for which the processing entity is responsible. In terms of the SIP protocol, the processing associated with retargeting is described in Sections 16.5 and 16.6 of [RFC3261]. As described in Section 16.5 of [RFC3261], it is possible for the target of a request to be changed by the same proxy multiple times (referred to as 'internal retargeting' in Section 2), as the proxy MAY add targets to the target set after beginning Request Forwarding. Section 16.6 of [RFC3261] describes Request Forwarding. It is during this process of Request Forwarding that the History Information is captured as an optional, additional header field. Thus, the addition of the History-Info header does not impact fundamental SIP Request Forwarding. An entity (UA or proxy) changing the target of a request in response to a redirect or REFER SHOULD also propagate any History-Info header from the initial Request in the new request (GENERATION-req, FORWARDS-req).

3.1. Optionality of History-Info

The History-Info header is optional in that neither UAs nor Proxies are required to support it. A new Supported header, "histinfo", is included in the Request to indicate whether the History-Info header is returned in Responses (BACKWARDS-req). In addition to the "histinfo" Supported header, local policy determines whether or not the header is added to any request, or for a specific Request-URI,

being retargeted or forwarded. It is possible that this could restrict the applicability of services that make use of the Request History Information to be limited to retargeting within domain(s) controlled by the same local policy, or between domain(s) which negotiate policies with other domains to ensure support of the given policy, or services for which complete History Information isn't required to provide the service (OPTIONALITY-req). All applications making use of the History-Info header MUST clearly define the impact of the information not being available and specify the processing of such a request.

3.2. Securing History-Info

This document defines a new header for SIP. The use of the Transport Layer Security (TLS) protocol [RFC5246] as a mandatory 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. If TLS is not available for the connection over which the request is being forwarded, then the request MUST NOT include the History-Info header or the request MUST be redirected to the client, including the History-Info header, so that the request can be retargeted by the client.

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 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.

3.3. Ensuring the Privacy of History-Info

Since the History-Info header can inadvertently reveal information about the requestor as described in [RFC3323], the Privacy header SHOULD be used to determine whether an intermediary can include the History-Info header in a Request that it receives and forwards (PRIV-req-2) or that it retargets (PRIV-req-1). Thus, the History- Info header SHOULD NOT be included in Requests where the requestor has indicated a priv-value of Session- or Header-level privacy.

In addition, the History-Info header can reveal general routing information, which may be viewed by a specific intermediary or network, to be subject to privacy restrictions. Thus, local policy MAY also be used to determine whether to include the History-Info header at all, whether to capture a specific Request-URI in the header, or whether it be included only in the Request as it is retargeted within a specific domain (PRIV-req-3). In the latter case, this is accomplished by adding a new priv-value, history, to the Privacy header [RFC3323] indicating whether any or a specific History-Info header(s) SHOULD be forwarded.

It is recognized that satisfying the privacy requirements can impact the functionality of this solution by overriding the request to generate the information. As with the optionality and security requirements, applications making use of History-Info SHOULD address any impact this may have or MUST explain why it does not impact the application.

4. Request History Information Protocol Details

This section contains the details and usage of the proposed new SIP protocol elements. It also discusses the security aspects of the solution.

4.1. Protocol Structure of History-Info

History-Info is a header field as defined by [RFC3261]. It is an optional header field and MAY appear in any request or response not associated with a dialog or which starts a dialog. 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.

This document adds the following entry to Table 2 of [RFC3261]. The additions to this table are also provided for extension methods at the time of publication of this document. This is provided as a courtesy to the reader and is not normative in any way.

Header field	where	proxy	ACK	BYE	CAN	INV	OPT	REG	MSG
History-Info		amdr	-	-	-	o	o	o	o

SUB	NOT	REF	INF	UPD	PRA	PUB
History-Info		amdr	o	o	o	-	-	-	o

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 4.3.3.1), including the index and securing the header, the ordering of the History-Info headers in the request is assured (SEC-req-2). 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.
- Retarget (hi-target): An optional parameter for History-Info reflecting that the hi-targeted-to-uri was retargeted to a contact URI based on a lookup in a location service. A retarget parameter is not added for a hi-entry when it is first added in a History-Info header, but rather is added when the retargeting actually occurs - i.e., the parameter indicates that the specific hi-targeted-to-uri was retargeted and thus the previous information in the request-URI is "lost". Note that retargeting only occurs when the hi-targeted-to-uri indicates a domain for which the processing entity is responsible. Thus, it would be the same processing entity that initially added the hi-targeted-to-URI to the header that would be adding the retarget parameter. Upon receipt of a request or response containing the History-Info header, a UA can determine the "lost" target for a specific request by traversing the HI entries in reverse order to find the first one tagged with the retarget parameter. [Editor's note: the term "retarget" is tentative and will be changed to reflect the consensus so that it is meaningful to the applications such as those described in [I-D.rosenberg-sip-target-uri-delivery] and is unambigous with regards to SIP terminology in [RFC3261].]
- Reason: An optional parameter for History-Info, reflected in the History-Info header by including the Reason Header [RFC3326] escaped in the hi-targeted-to-uri. A reason is not included for a hi-targeted-to-uri when it is first added in a History-Info header, but rather is added when the retargeting actually occurs in the same situations in which the retarget parameter is added.

- Privacy: An optional parameter for History-Info, reflected in the History-Info header field values by including the Privacy Header [RFC3323] with a priv-value of "history" escaped in the hi-targeted-to-uri or by adding the Privacy header with a priv-value of "history" to the Request. The use of the Privacy Header with a priv-value of "history" indicates whether a specific or all History-Info headers should not be forwarded.
- Extension (hi-extension): An optional parameter to allow for future optional extensions. As per [RFC3261], 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]:

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

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

hi-targeted-to-uri = name-addr

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

hi-param = hi-target / hi-extension

hi-target = "retarget"

hi-extension = generic-param
```

4.2. Protocol examples

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?Reason=SIP%3B\
              cause%3D302>;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;retarget,\
              <sip:45432@vm.example.com>;index=1.3
```

4.3. Protocol Usage

This section describes the processing specific to UAs and Proxies for the History-Info header, the "histinfo" option tag, and the priv-value of "history". As discussed in Section 1.3, the fundamental objective is to capture the target Request-URIs as a request is forwarded. This allows for the capturing of the history of a request that would be lost due to subsequent (re)targeting and forwarding. To accomplish this for the entire history of a request, either the UAC must capture the Request-URI in a History-Info header in the initial request or a proxy must add a History-Info header with both a hi-entry for the Request-URI in the initial request and a hi-entry for the target Request-URI as the request is forwarded. The basic processing is for each entity forwarding a request to add a hi-entry for the target Request-URI, updating the index and adding the Reason and Retarget parameters as appropriate for any retargeted Request-URIs.

4.3.1. User Agent Client (UAC) Behavior

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 improve the diagnostic utility of its request by adding a History-Info header, using the Request-URI of the request as the hi-target-to-uri and initializing the index to the RECOMMENDED 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. The UAC SHOULD NOT include the hi-target parameter in the hi-entry in this case.

In the case where the request is routed to a redirect server and the UAC receives a 3xx response with a Contact header, the UAC MAY maintain the previous hi-entry(s) in the request. In this case, the Reason header and "retarget" parameter SHOULD be associated with the hi-targeted-to-uri in the previous (last) hi-entry, as described in Section 4.3.3.1.2. A new hi-entry MAY then be added for the URI from the Contact header (which becomes the new Request-URI). In this case, the index is 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 4.3.3.1.3. In this case, the UAC MAY add hi-target to the request. An example of this scenario can be found in Appendix A.5.

A UAC that does not want the History-Info header added due to privacy considerations SHOULD include a Privacy header with a priv-value(s) of "session", "header", or "history" in the request.

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. However, the validity of the information SHOULD be ensured prior to any application usage. 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.

4.3.2. User Agent Server (UAS) Behavior

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 a UAS wishes to determine the original targeted URI, the values in the History-Info header field are traversed in reverse order. Note that, the value of the "index" attribute is not relevant; the traversal is in order of the header fields values themselves until the first header field value containing the "retarget" parameter is found. If there is no hi-entry with the "retarget" parameter, the intended target URI for the request cannot be reliably determined. If it does exist, the URI is examined. If the domain of the URI matches the domain of the UA, based on the UA's configured awareness of its own domain, that URI is the target URI for the request. If the domains do not match, the target URI cannot be reliably determined. This domain check is present to handle cases where a request is forwarded through two separate domains, and the domain of the actual UAS didn't support this specification, but the previous domain did.

If the "histinfo" option tag is received in a request, the UAS SHOULD include any History-Info received in the request in the subsequent response.

4.3.3. Proxy Behavior

The inclusion of the History-Info header in a Request does not alter the fundamental processing of proxies for determining request targets as defined in Section 16.5 of [RFC3261]. Whether a proxy adds the History-Info header or a new hi-entry as it forwards a Request depends upon the following considerations:

1. Whether the Request contains the "histinfo" option tag in the Supported header.
2. Whether the proxy supports the History-Info header.
3. Whether the Request contains a Privacy header with a priv-value of "session", "header", or "history".
4. Whether any History-Info header added for a proxy/domain should go outside that domain. An example being the use of the History-Info header within the specific domain in which it is retargeted, however, policies (for privacy, user and network security, etc.) would prohibit the exposure of that information outside that domain. To accommodate such a scenario, a proxy MAY insert the Privacy header with a priv-value of "history" when the request is being forwarded within the same domain. An example of such an application is provided in Appendix A.4.
5. Whether a hi-entry is added for a specific Request-URI due to local privacy policy considerations. A proxy MAY add the Privacy header with a priv-value of "history" associated with the specific hi-targeted-to-uri.

An example policy would be a proxy that only adds the History-Info header if the "histinfo" option tag is in the Supported header. Other proxies may have a policy that they always add the header, but never forward it outside a particular domain, accomplishing this by adding a Privacy header with a priv-value of "history" to each hi-entry to allow the information to be collected for internal retargeting only.

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).

Consistent with basic SIP processing of optional headers, proxies SHOULD maintain the History-Info header(s), received in messages being forwarded, independent of whether local policy supports History-Info.

The specific processing by proxies for adding the History-Info headers in Requests and Responses, to accommodate the considerations outlined above, is described in detail in the following sections.

4.3.3.1. Adding the History-Info Header to Requests

Upon evaluation of the considerations under which the History-Info header is to be included in requests (e.g., no Privacy header overriding inclusion, local policy supports, etc.), detailed in Section 4.3.3, a proxy SHOULD add a hi-entry as it forwards a

Request. Section 16.6 of [RFC3261] defines the steps to be followed as the proxy forwards a Request. Step 5 prescribes the addition of optional headers. Although this would seem the appropriate step for adding the History-Info header, the interaction with Step 6, "Postprocess routing information", and the impact of a strict route in the Route header could result in the Request-URI being changed; thus, adding the History-Info header between Steps 8 (adding Via header) and 9 (adding Content-Length) is RECOMMENDED. Note that in the case of loose routing, the Request-URI does not change during the forwarding of a Request; thus, the capturing of History-Info for such a request results in duplicate Request-URIs with different indices. The hi-entry MUST be added following any hi-entry received in the request being forwarded. Additionally, if a request is received that doesn't include a History-Info header, the proxy MAY add a History-Info header with a hi-entry preceding the one being added for the current request being forwarded. The index for this hi-entry is RECOMMENDED to start at 1. The following subsections define the details of creating the information associated with each hi-entry.

4.3.3.1.1. Privacy in the History-Info Header

If there is a Privacy header in the request with a priv-value of "session", "header", or "history", a hi-entry MAY be added, if the request is being forwarded to a Request-URI associated with a domain for which the processing entity is responsible (and provided local policy supports the History-Info header, etc.). If a request is being forwarded to a Request-URI associated with a domain for which the proxy is not responsible and there is a Privacy header in the request with a priv-value of "session", "header", or "history", the proxy SHOULD remove any hi-entry(s) prior to forwarding, depending upon local policy and whether the proxy might know a priori that it can rely on a downstream privacy service to apply the requested privacy.

For the scenario where there is no Privacy header in the request and the request is being forwarded to a Request-URI associated with the domain(s) for which this entity is responsible, there are several additional considerations:

- If there is no local policy associated with privacy, then a hi-entry MAY be added to the Request.
- If the proxy's local policies, per consideration 4 in section Section 4.3.3, indicate that the History-Info header should not 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 that proxy in this scenario.

- If the proxy's policy, per consideration 5 in Section 4.3.3, indicates that History-Info for a specific Request-URI should not 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 the specific hi-entry, for that specific hi-targeted-to-uri, added by that proxy in this scenario.

If a request is being forwarded to a Request-URI associated with a domain for which the proxy is not responsible and local policy requires privacy associated with any, or with specific, hi-entries it has added, any hi-entry with a priv-value of "history" SHOULD be removed prior to forwarding.

4.3.3.1.2. Reason in the History-Info Header

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, 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 non-SIP Reason header (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.

For retargets as a result of timeouts or internal events, a Reason MAY be associated with the hi-targeted-to-uri that has been retargeted.

The addition of the Reason should occur prior to the forwarding of the request (which may add a new hi-entry with a new hi-targeted-to-uri) as it is associated with the hi-targeted-to-uri that has been retargeted, since it reflects the reason why the Request to that specific URI was not successful.

4.3.3.1.3. Indexing in the History-Info Header

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 4.1, 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. It is recommended

that for each level of indexing, the index start at 1. It is recommended that an increment of 1 is used for advancing to a new branch.

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 level of indexing since the depth/length of the branch is increasing. To accomplish this, the proxy 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 RECOMMENDED to be 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 Proxy - 1st instance: For the first instance of retargeting within a Proxy, the calculation of the index follows that prescribed for basic forwarding.
3. Retargeting within a Proxy - subsequent instance: For each subsequent retargeting of a request by the same proxy, 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 proxy, 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 RECOMMENDED to be 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. Retargeting the request in parallel (forking): If the request forwarding is done in parallel, the index MUST be captured for each forked request per the rules above, with each new Request having a unique index. The only difference in the messaging for this scenario and the messaging produced per basic proxy retargeting in rules 2 and 3 is these forwarded requests do not have History-Info entries associated with their peers. The proxy builds the subsequent response (or request) using the aggregated information associated with each of those requests and including the header entries in the order indicated by the indexing. Responses are processed as described in Section 16.7 of [RFC3261] with the aggregated History-Info entries processed similar to Step 7 "Aggregate Authentication Header Field Values". Appendix A.2 provides an example of a parallel request scenario, highlighting this indexing mechanism.

4.3.3.1.4. Request Target in the History-Info Header

A "retarget" attribute SHOULD be added in the case that a proxy is changing the Request-URI based on a location service lookup or as a result of receiving a 3xx response with a Contact header containing URIs to which the request should be redirected. The addition of the "retarget" parameter should occur prior to the forwarding of the request (which may add a new hi-entry with a new hi-targeted-to- uri) as it is associated with the hi-targeted-to-uri that has been retargeted. If the incoming request already contains a History-Info header field, and the hi-targeted-to-uri in the last hi-entry is identical to the Request-URI of the received request, the proxy MUST add a "retarget" attribute to that hi-entry. In the case that the request did not contain a History-Info header, or if the last hi-entry is not identical to the Request-URI of the received request, the proxy MUST add another History-Info header field value as described in Section 4.3.3.1. and MUST add a "retarget" attribute to this hi-entry. The index is set as defined in Section 4.3.3.1.3.

Once the proxy has translated the Request-URI into a contact URI based on a location service lookup, it MUST add an additional hi-entry containing the Contact URI for each request to be forwarded as described in Section 4.3.3.1. The "retarget" attribute MUST NOT be added to this hi-entry.

If the proxy is redirecting and not forwarding the request, it MAY include the History-Info headers received in the request in the response. In this case, the proxy MUST add the "retarget" attribute to the last hi-entry received in the request to the last hi-entry in the response. The proxy SHOULD NOT add an hi-entry for the contact URI; the proxy that receives the 3xx response does that.

4.3.3.2. Processing History-Info in Responses

A proxy that receives a Request with the "histinfo" option tag in the Supported header, and depending upon a local policy supporting the capture of History-Info, SHOULD return captured History-Info in subsequent, provisional, and final responses to the Request, subject to the following considerations for privacy:

- If the response is being forwarded to a Request-URI associated with a domain for which the proxy is not responsible and there was a Privacy header, in the request received by the proxy, with a priv-value of "session", "header", or "history", the proxy MUST remove the History-Info header (i.e., all hi-entries) prior to forwarding.
- If a request is being forwarded to a Request-URI associated with a domain for which the proxy is not responsible and local policy requires privacy associated with any or all hi-entry(s) it has

added, any hi-entry with a priv-value of "history" MUST be removed prior to forwarding.

- o If a proxy receives a response from another intermediary associated with a domain for which it is responsible, including hi-entry(s) with privacy headers, and that response is to be forwarded to a domain for which it is not responsible, then those hi-entry(s) MUST be removed.

The processing of History-Info in responses follows the methodology described in Section 16.7 of [RFC3261], with the processing of History-Info headers adding an additional step, just before Step 9, "Forwarding the Response".

4.3.4. Redirect Server Behavior

A redirect server SHOULD NOT add any new History-Info, as that would be done by the entity receiving the 3xx response. However, a redirect server MAY include History-Info in responses by adding any History-Info headers received in a request to a subsequent response and if so, it MUST add the "retarget" parameter to the last hi-entry.

4.4. Security for History-Info

As discussed in Section 3, the security requirements are met by recommending the use of TLS (a basic SIP requirement per [RFC3261]) for hop-by-hop security. If TLS is not available on the connection over which a request containing a History-Info header is being forwarded, then either of the following two options MUST be implemented:

- o The History-Info header MUST be removed prior to forwarding the request, or
- o The request MUST be redirected, including the History-Info header in the response, to allow the UAC to securely issue the request, including the History-Info header.

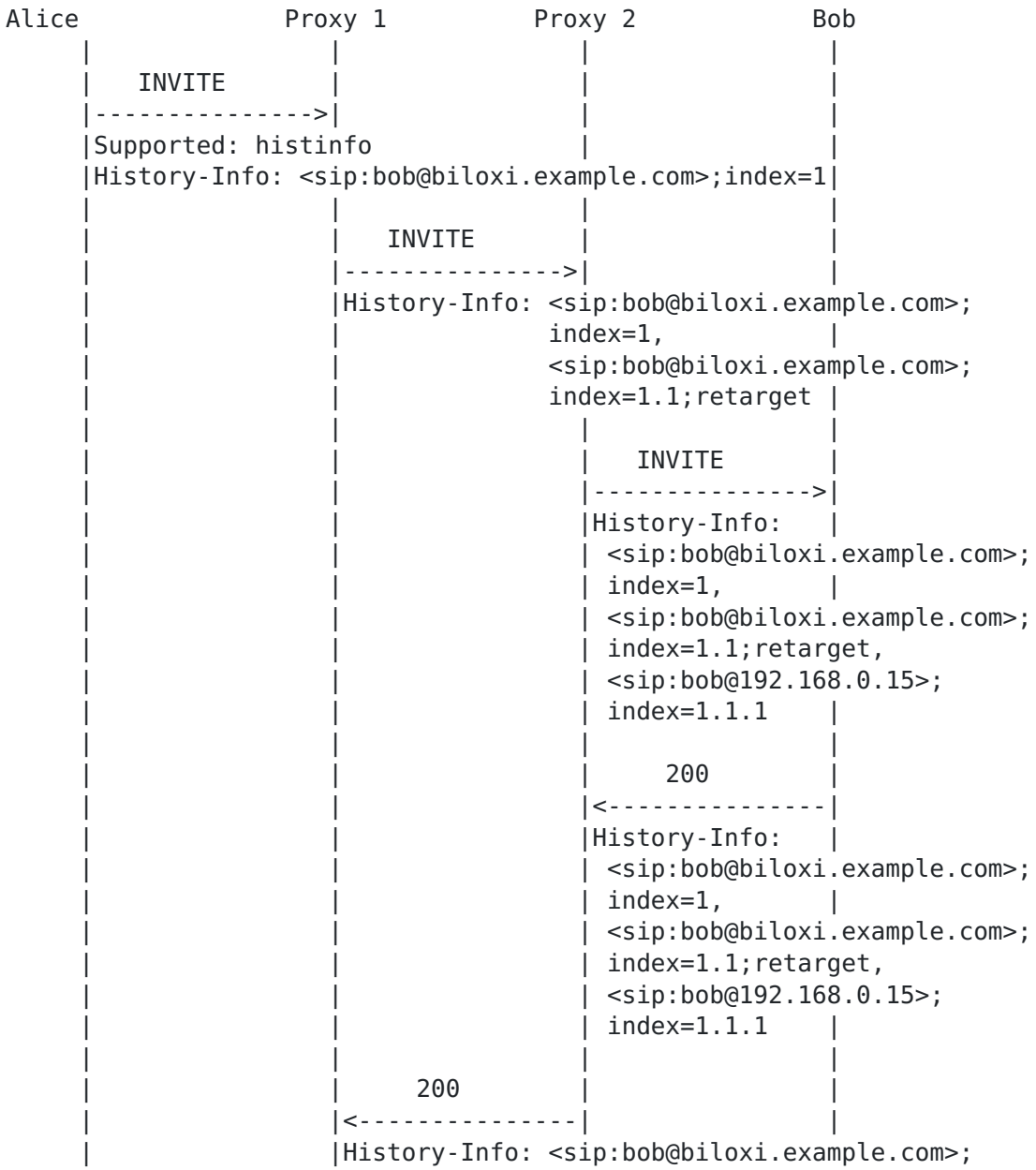
4.5. Example Call Flows with History-Info Header

This section contains some basic call examples using the History-Info header, including the use of privacy and the "retarget" attribute.

The formatting in these scenarios 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 4.2 for the proper formatting. Additional detailed scenarios are available in the Appendix A.

4.5.1. Basic Call with History-Info

In this example, Alice (@atlanta.example.com) calls Bob (@biloxi.example.com). Alice's home proxy (Proxy 1) forwards the request to Bob's proxy (Proxy2). When the request arrives at Proxy2 in domain biloxi.example.com, Proxy2 does a location service lookup for bob@biloxi.example.com and changes the target of the request to Bob's local URI.



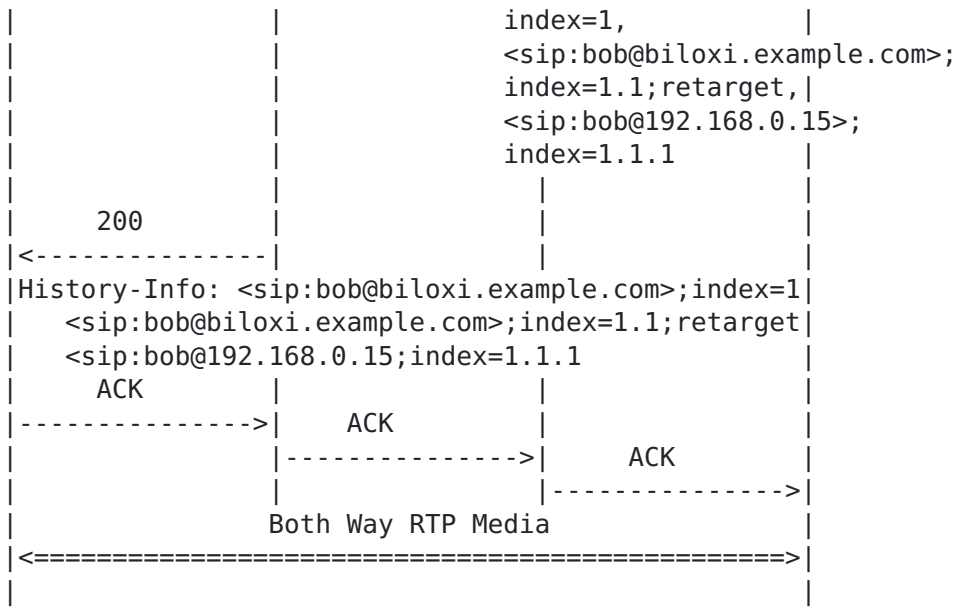


Figure 1: Basic Call

4.5.2. History-Info with Privacy Header

The next example provides the basic call scenario Section 4.5.1 using one of the privacy mechanisms, with Proxy2 adding the Privacy header indicating that the History-Info header is not to be propagated outside Proxy2's 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.

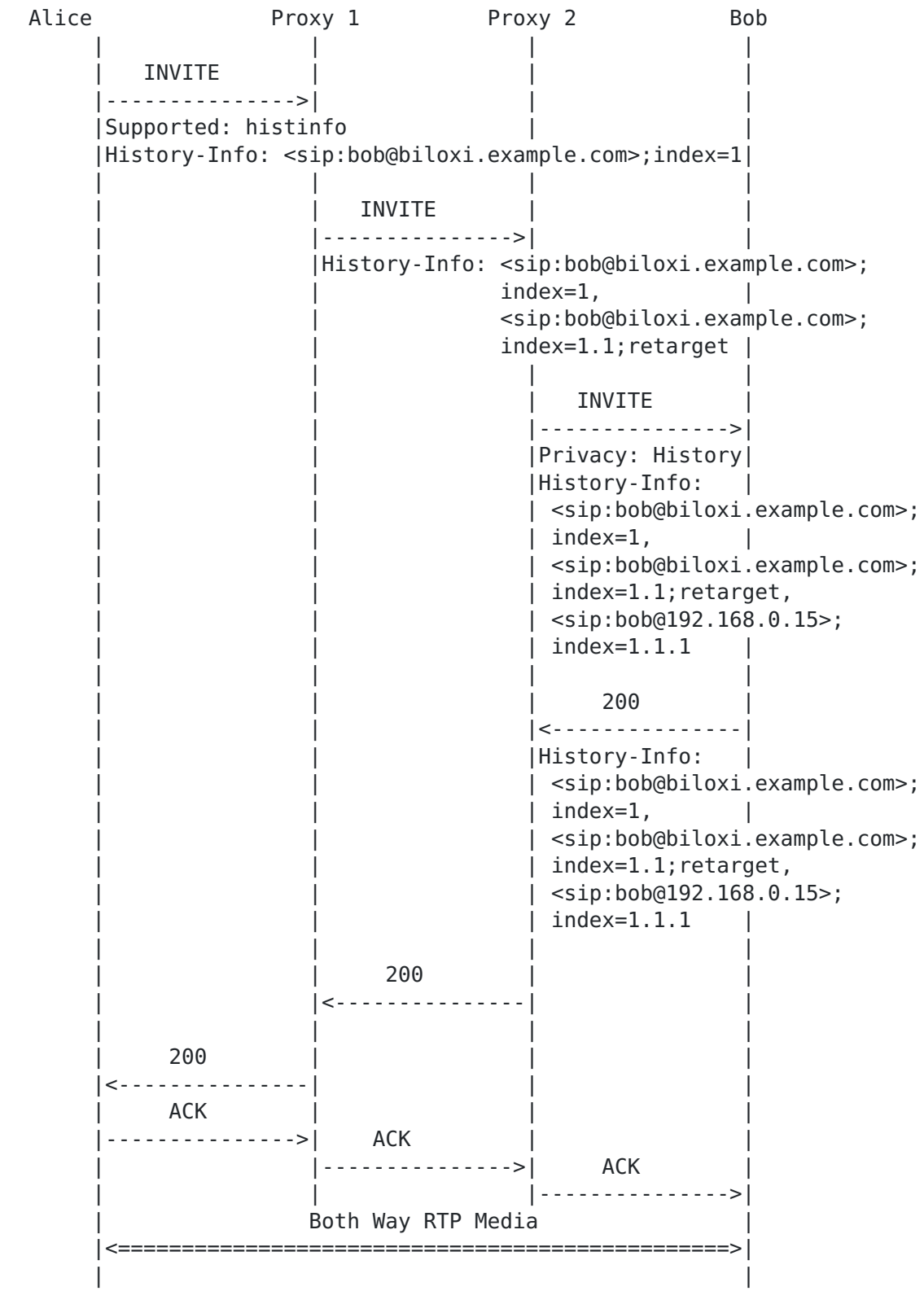

Alice
Proxy 1
Proxy 2
Bob
INVITE
Supported: histinfo
History-Info: <sip:bob@biloxi.example.com>;index=1
INVITE
History-Info: <sip:bob@biloxi.example.com>;
index=1,
<sip:bob@biloxi.example.com>;
index=1.1;retarget
INVITE
Privacy: History
History-Info:
<sip:bob@biloxi.example.com>;
index=1,
<sip:bob@biloxi.example.com>;
index=1.1;retarget,
<sip:bob@192.168.0.15>;
index=1.1.1
200
History-Info:
<sip:bob@biloxi.example.com>;
index=1,
<sip:bob@biloxi.example.com>;
index=1.1;retarget,
<sip:bob@192.168.0.15>;
index=1.1.1
200
200
ACK
ACK
ACK
Both Way RTP Media

Figure 2: Example with Privacy Header for Entire Request at Proxy2

4.5.3. Privacy Header for a Specific History-Info Entry

This example also provides the basic call scenario Section 4.5.1 using one of the privacy mechanisms, however, due to local policy at Proxy2, 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 maintaining privacy for Bob's local URI.

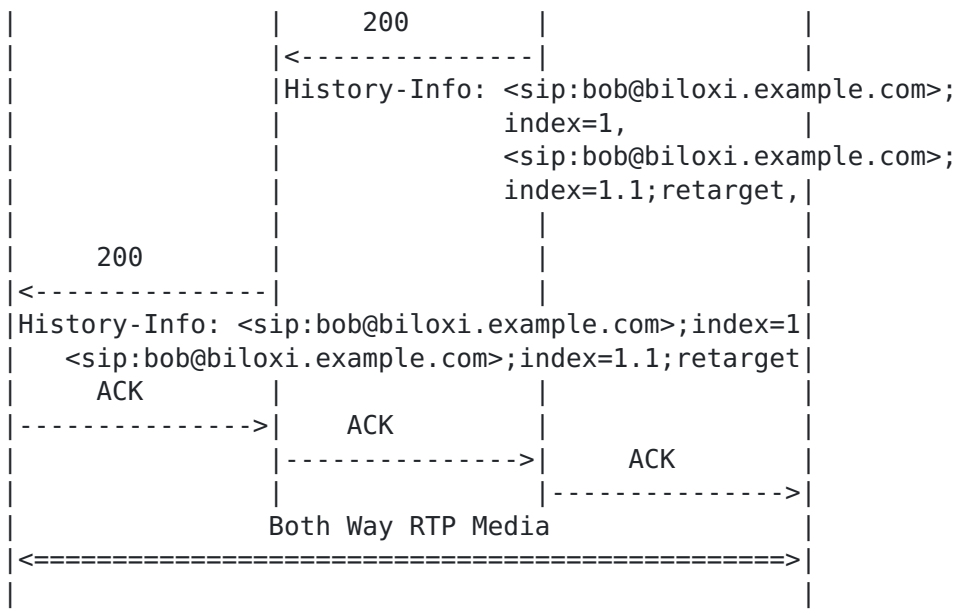


Figure 3: Example with Privacy Header for Specific URI at Proxy2

5. Application Considerations

As seen by the example scenarios in the Appendix A, 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.

4. The security associated with the History-Info header requires the use of TLS. In the case of TLS not being available for a connection over which a request is being forwarded, the History-Info header may be removed from a request. The impact of lack of having the information depends upon the nature of the specific application (e.g., Is the information something that appears on a display or is it processed by automata which could have negative impacts on the subsequent processing of a request?). It is suggested that the impact of an intermediary not supporting the security recommendations should be evaluated by the application to ensure that the impacts have been sufficiently addressed by the application.

6. Security Considerations

The threat model and related security and privacy requirements for the History-Info header are described in Sections 2.1 and 2.2 of this document. Sections 3.2, 3.3, and 4.4 provide normative recommendations related to security and privacy fulfilling these requirements. The use of TLS is mandated between the entities (i.e., UAC to Proxy, Proxy to Proxy, and Proxy to UAS) that use the History-Info header. The appropriate handling of a request in the case that TLS is not available for a specific connection is described in Section 5.

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.

7. IANA Considerations

This document requires several IANA registrations detailed in the following sections.

7.1. Registration of New SIP History-Info Header

This document defines a new SIP header field name: History-Info and a new 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:

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 support for 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.

7.2. Registration of "history" for SIP Privacy Header

This document defines a new 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.

8. Contributors

Cullen Jennings, Mark Watson, and Jon Peterson contributed to the development of the initial requirements for [RFC4244].

Cullen Jennings and Mark Watson provided substantial input in the form of email discussion in the development of the initial version of

the individual solution document which provided the basis for [RFC4244].

Jonathan Rosenberg produced the initial document that provided the basis for the addition of the "target" parameter to the History-Info header, as well as some content for this document.

9. Acknowledgements

The editor would like to acknowledge the constructive feedback provided by Robert Sparks, Paul Kyzivat, Scott Orton, John Elwell, Nir Chen, Francois Audet, 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] The editor would like to acknowledge the significant input from Rohan Mahy on some of the normative aspects of the ABNF for [RFC4244], particularly around the need for and format of the index and around the security aspects.

10. Changes since last Version

NOTE TO THE RFC-Editor: Please remove this section prior to publication as an RFC.

Changes from RFC4244 to individual -00:

1. Clarified that HI captures both retargeting as well as cases of just forwarding a request. Added descriptions of the usage of the terms "retarget", "forward" and "redirect" to the terminology section.
2. Added additional examples for the functionalityy provided by HI for core SIP.
3. Added "retarget" parameter to HI header to ABNF and protocol description, as well as defining proxy, UAC and UAS behavior for the parameter.
4. Simplified example call flow in section 4.5 and added "retarget" parameter. Moved previous call flow to appendix.
5. Fixed ABNF per RFC4244 errata "dot" -> "."

11. References

11.1. Normative References

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

June 2002.

[RFC3326] Schulzrinne, H., Oran, D., and G. Camarillo, "The Reason Header Field for the Session Initiation Protocol (SIP)", RFC 3326, December 2002.

[RFC3323] Peterson, J., "A Privacy Mechanism for the Session Initiation Protocol (SIP)", RFC 3323, November 2002.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC5246] Dierks, T. and E. Rescorla, "The Transport Layer Security (TLS) Protocol Version 1.2", RFC 5246, August 2008.

[RFC4244] Barnes, M., "An Extension to the Session Initiation Protocol (SIP) for Request History Information", RFC 4244, November 2005.

11.2. Informative References

[RFC3665] Johnston, A., Donovan, S., Sparks, R., Cunningham, C., and K. Summers, "Session Initiation Protocol (SIP) Basic Call Flow Examples", BCP 75, RFC 3665, December 2003.

[I-D.ietf-sip-gruu]
Rosenberg, J., "Obtaining and Using Globally Routable User Agent (UA) URIs (GRUU) in the Session Initiation Protocol (SIP)", draft-ietf-sip-gruu-15 (work in progress), October 2007.

[I-D.rosenberg-sip-target-uri-delivery]
Rosenberg, J., "Delivery of Request-URI Targets to User Agents", draft-rosenberg-sip-target-uri-delivery-00 (work in progress), October 2008.

[RFC3087] Campbell, B. and R. Sparks, "Control of Service Context using SIP Request-URI", RFC 3087, April 2001.

[RFC4240] Burger, E., Van Dyke, J., and A. Spitzer, "Basic Network Media Services with SIP", RFC 4240, December 2005.

[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.

Appendix A. Detailed call flows

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.

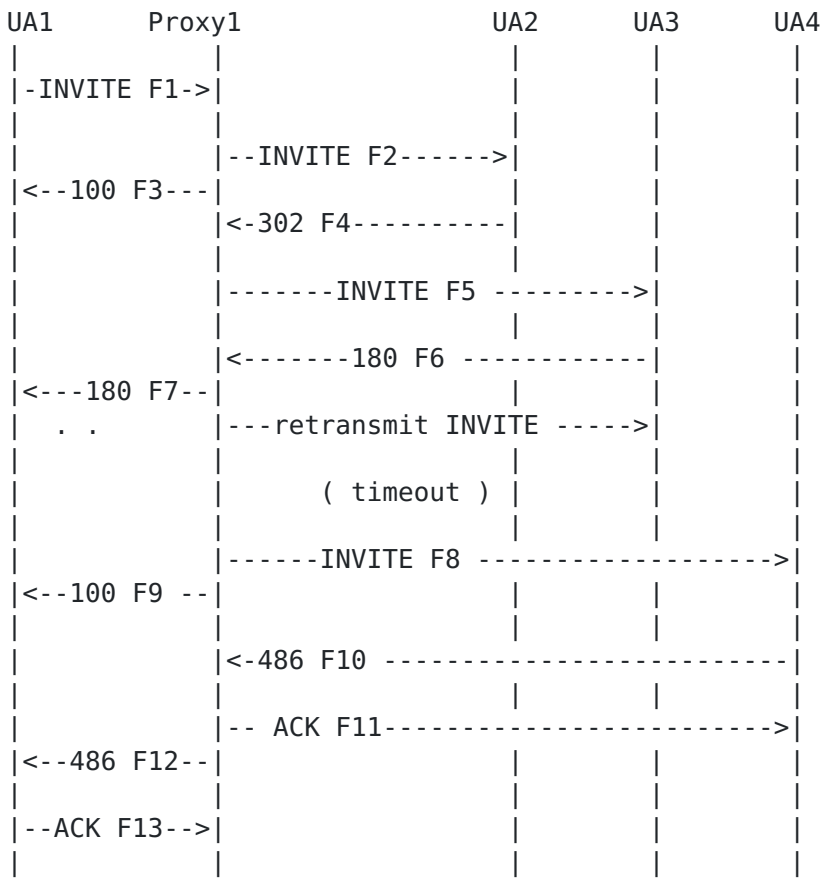
A.1. Sequentially Forking (History-Info in Response)

This scenario highlights an example where the History-Info in the response is useful to an application or user that originated the request.

Alice at UA1 sends a call to Bob via Proxy1. Proxy1 sequentially tries several places (UA2, UA3 and UA4) unsuccessfully before sending a response to Alice.

This scenario is provided to show that by providing the History-Info to UA1, the end-user or an application at UA1 could make a decision on how best to attempt finding Bob. Without this mechanism, UA1 might well attempt UA3 (and thus UA4) and then re-attempt UA4 on a third manual attempt at reaching Bob. With this mechanism, either the end-user or application could know that Bob is busy on his home phone and is physically not in the office. 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.

In this scenario, since UA1 has not included the original Request-URI in the INVITE, the proxy adds a hi-entry to capture the original Request-URI to provide the complete set of information, as discussed in Section 4.3.3.1.



Message Details

F1 INVITE UA1 ->Proxy1

```
INVITE sip:UserA@example.com SIP/2.0
Via: SIP/2.0/UDP example.net:5060
From: Alice <sip:User1@example.net>
To: Bob <sip:UserA@example.com>
Call-Id: 12345600@example.net
CSeq: 1 INVITE
Contact: Alice <sip:User1@example.net>
Content-Type: application/sdp
Content-Length: <appropriate value>

v=0
o=UserA 2890844526 2890844526 IN IP4 client.example.net
s=Session SDP
c=IN IP4 192.0.2.3
t=0 0
m=audio 49170 RTP/AVP 0
```

```
   a=rtpmap:0 PCMU/8000


   /* Client for UA1 prepares to receive data on port 49170
      from the network. */

   F2 INVITE  Proxy1 ->UA2

   INVITE sip:UserA@ims.example.com SIP/2.0
   Via: SIP/2.0/UDP ims.example.com:5060;branch=1
   Via: SIP/2.0/UDP example.net:5060
   Record-Route: <sip:UserA@example.com>
   From: Alice <sip:User1@example.net>
   To: Bob <sip:UserA@example.com>
   Call-Id: 12345600@example.net
   CSeq: 1 INVITE
   History-Info: <sip:UserA@example.com>;index=1,\
                 <sip:UserA@ims.example.com>;index=1.1
   Contact: Alice <sip:User1@example.net>
   Content-Type: application/sdp
   Content-Length: <appropriate value>

   v=0
   o=UserA 2890844526 2890844526 IN IP4 client.example.net
   s=Session SDP
   c=IN IP4 192.0.2.3
   t=0 0
   m=audio 49170 RTP/AVP 0
   a=rtpmap:0 PCMU/8000


   F3 100 Trying Proxy1 ->UA1

   SIP/2.0 100 Trying
   Via: SIP/2.0/UDP example.net:5060
   From: Alice <sip:User1@example.net>
   To: Bob <sip:UserA@example.com>
   Call-Id: 12345600@example.net
   CSeq: 1 INVITE
   Content-Length: 0
```

F4 302 Moved Temporarily UA2 ->Proxy1

```
SIP/2.0 302 Moved Temporarily
Via: SIP/2.0/UDP ims.example.com:5060;branch=1
Via: SIP/2.0/UDP example.net:5060
From: Alice <sip:User1@example.net>
To: Bob <sip:UserA@example.com>;tag=3
Call-Id: 12345600@example.net
CSeq: 1 INVITE
Contact: <sip:UserB@example.com>
Content-Length: 0
```

F5 INVITE Proxy1 -> UA3

```
INVITE sip:UserB@example.com SIP/2.0
Via: SIP/2.0/UDP ims.example.com:5060;branch=2
Via: SIP/2.0/UDP example.net:5060
From: Alice <sip:User1@example.net>
To: Bob <sip:UserA@example.com>
Call-Id: 12345600@example.net
History-Info: <sip:UserA@example.com>; index=1,\
              <sip:UserA@ims.example.com?Reason=SIP;\
              cause=302;text="Moved Temporarily">;index=1.1,\
              <sip:UserB@example.com>;index=1.2
CSeq: 1 INVITE
Contact: Alice <sip:User1@example.net>
Content-Type: application/sdp
Content-Length: <appropriate value>

v=0
o=User1 2890844526 2890844526 IN IP4 client.example.net
s=Session SDP
c=IN IP4 192.0.2.3
t=0 0
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

F6 180 Ringing UA3 ->Proxy1

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP example.net:5060
From: Alice <sip:User1@example.net>
To: Bob <sip:UserA@example.com>;tag=5
Call-ID: 12345600@example.net
CSeq: 1 INVITE
Content-Length: 0
```

```
   F7 180 Ringing Proxy1 -> UA1

   SIP/2.0 180 Ringing
   SIP/2.0/UDP example.net:5060
   From: Alice <sip:User1@example.net>
   To: Bob <sip:UserA@example.com>
   Call-Id: 12345600@example.net
   CSeq: 1 INVITE
   Content-Length: 0


   /* User B is not available.  INVITE is sent multiple
      times until it times out. */
   /* The proxy forwards the INVITE to UA4 after adding the
      additional History Information entry. */

   F8 INVITE Proxy1 -> UA4

   INVITE sip:UserC@example.com SIP/2.0
   Via: SIP/2.0/UDP ims.example.com:5060;branch=3
   Via: SIP/2.0/UDP example.net:5060
   From: Alice <sip:User1@example.net>
   To: Bob <sip:UserA@example.com>
   Call-Id: 12345600@example.net
   History-Info: <sip:UserA@example.com>; index=1,\
                 <sip:UserA@ims.example.com?Reason=SIP;\
                 cause=302; text="Moved Temporarily">;index=1.1,\
                 <sip:UserB@example.com?Reason=SIP;cause=480;\
                 text="Temporarily Unavailable" >;index=1.2,\
                 <sip:UserC@example.com>;index=1.3
   CSeq: 1 INVITE
   Contact: Alice <sip:User1@example.net>
   Content-Type: application/sdp
   Content-Length: <appropriate value>

   v=0
   o=User1 2890844526 2890844526 IN IP4 client.example.net
   s=Session SDP
   c=IN IP4 192.0.2.3
   t=0 0
   m=audio 49170 RTP/AVP 0
   a=rtpmap:0 PCMU/8000
```

F9 100 Trying Proxy1 ->UA1

```
SIP/2.0 100 Trying
Via: SIP/2.0/UDP example.net:5060
From: Alice <sip:User1@example.net>
To: Bob <sip:UserA@example.com>
Call-Id: 12345600@example.net
CSeq: 1 INVITE
Content-Length: 0
```

F10 486 Busy Here UA4 -> Proxy1

```
SIP/2.0  486 Busy Here
Via: SIP/2.0/UDP ims.example.com:5060;branch=3
Via: SIP/2.0/UDP example.net:5060
From: Alice <sip:User1@example.net>
To: Bob <sip:UserA@example.com>
Call-Id: 12345600@example.net
CSeq: 1 INVITE
Content-Length: 0
```

F11 ACK Proxy1 -> UA4

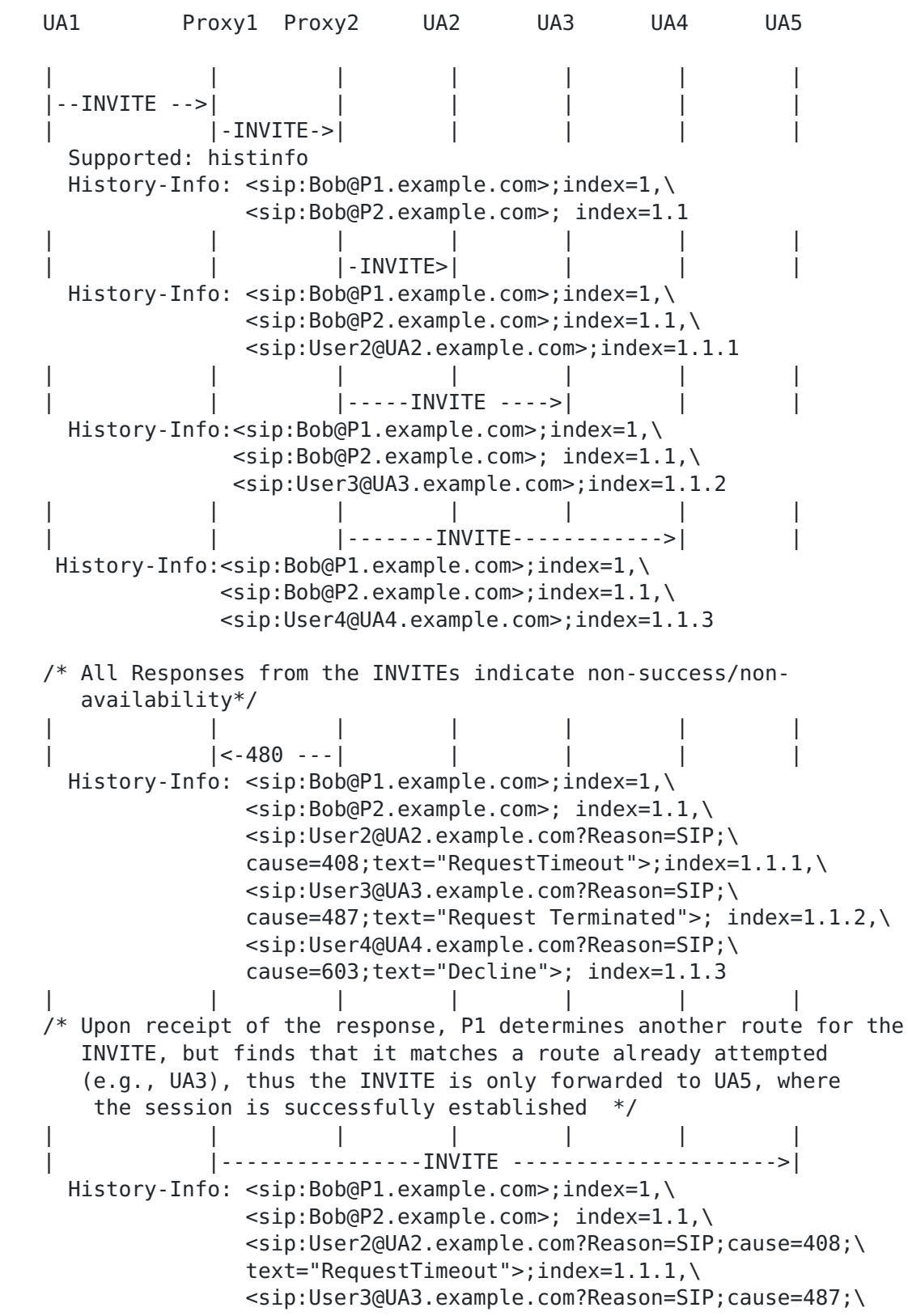
```
ACK sip:UserC@example.com SIP/2.0
Via: SIP/2.0/UDP example.net:5060
From: Alice <sip:User1@example.net>
To: Bob <sip:UserA@example.com>
Call-Id: 12345600@example.net
CSeq: 1 ACK
Content-Length: 0
```

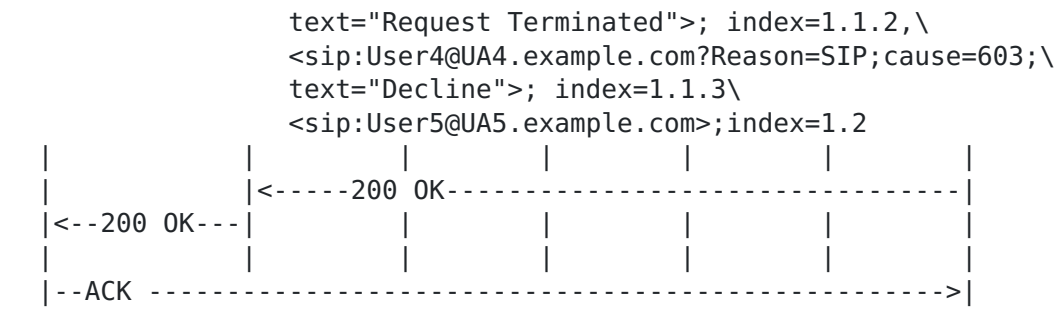
```
   /* The proxy forwards the 486 to Alice after adding the
      associated History Information entries from the series of
      INVITES */

         F12 486 Busy Here Proxy1 -> UA1

   SIP/2.0  486 Busy Here
   Via: SIP/2.0/UDP example.net:5060
   From: Alice <sip:User1@example.net>
   To: Bob <sip:UserA@example.com>
   Call-Id: 12345600@example.net
   History-Info: <sip:UserA@example.com>; index=1,\
                 <sip:UserA@ims.example.com?Reason=SIP;\
                 cause=302; text="Moved Temporarily">;index=1.1,\
                 <sip:UserB@example.com?Reason=SIP;cause=480;\
                 text="Temporarily Unavailable" >;index=1.2,\
                 <sip:UserC@example.com>;index=1.3
   CSeq: 1 INVITE
   Content-Length: 0


   F13 ACK Alice -> Proxy 1

   ACK sip:UserA@example.com SIP/2.0
   Via: SIP/2.0/UDP example.net:5060
   From: Alice <sip:User1@example.net>
   To: Bob <sip:UserA@example.com>
   Call-Id: 12345600@example.net
   CSeq: 1 ACK
   Content-Length: 0
```

A.2. Parallel Forking

This scenario highlights an example where the History-Info in the response is primarily of use in not retrying routes that have already been tried by another proxy. Note that this is just an example and that there may be valid reasons why a Proxy would want to retry the routes, and thus, this would likely be a local proxy or even user-specific policy.

UA1 sends a call to Bob to proxy 1. Proxy 1 forwards the request to Proxy 2. Proxy 2 sends the requests in parallel and tries several places (UA2, UA3, and UA4) before sending a response to Proxy 1 that all the places are busy. Proxy 1, without the History-Info, would try some of the same places (e.g., UA3) based upon registered contacts for Bob, before completing at UA5. However, with the History-Info, Proxy 1 determines that UA3 has already received the invite; thus, the INVITE goes directly to UA5.


UA1 Proxy1 Proxy2 UA2 UA3 UA4 UA5
|--INVITE -->|
|-INVITE->|
Supported: histinfo
History-Info: <sip:Bob@P1.example.com>;index=1,\
<sip:Bob@P2.example.com>; index=1.1
|-INVITE>|
History-Info: <sip:Bob@P1.example.com>;index=1,\
<sip:Bob@P2.example.com>;index=1.1,\
<sip:User2@UA2.example.com>;index=1.1.1
|-----INVITE ---->|
History-Info:<sip:Bob@P1.example.com>;index=1,\
<sip:Bob@P2.example.com>; index=1.1,\
<sip:User3@UA3.example.com>;index=1.1.2
|-------INVITE------------>|
History-Info:<sip:Bob@P1.example.com>;index=1,\
<sip:Bob@P2.example.com>;index=1.1,\
<sip:User4@UA4.example.com>;index=1.1.3
/* All Responses from the INVITEs indicate non-success/non-
availability*/
|<-480 ---|
History-Info: <sip:Bob@P1.example.com>;index=1,\
<sip:Bob@P2.example.com>; index=1.1,\
<sip:User2@UA2.example.com?Reason=SIP;\
cause=408;text="RequestTimeout">;index=1.1.1,\
<sip:User3@UA3.example.com?Reason=SIP;\
cause=487;text="Request Terminated">; index=1.1.2,\
<sip:User4@UA4.example.com?Reason=SIP;\
cause=603;text="Decline">; index=1.1.3
/* Upon receipt of the response, P1 determines another route for the
INVITE, but finds that it matches a route already attempted
(e.g., UA3), thus the INVITE is only forwarded to UA5, where
the session is successfully established */
|----------------INVITE --------------------->|
History-Info: <sip:Bob@P1.example.com>;index=1,\
<sip:Bob@P2.example.com>; index=1.1,\
<sip:User2@UA2.example.com?Reason=SIP;cause=408;\
text="RequestTimeout">;index=1.1.1,\
<sip:User3@UA3.example.com?Reason=SIP;cause=487;\

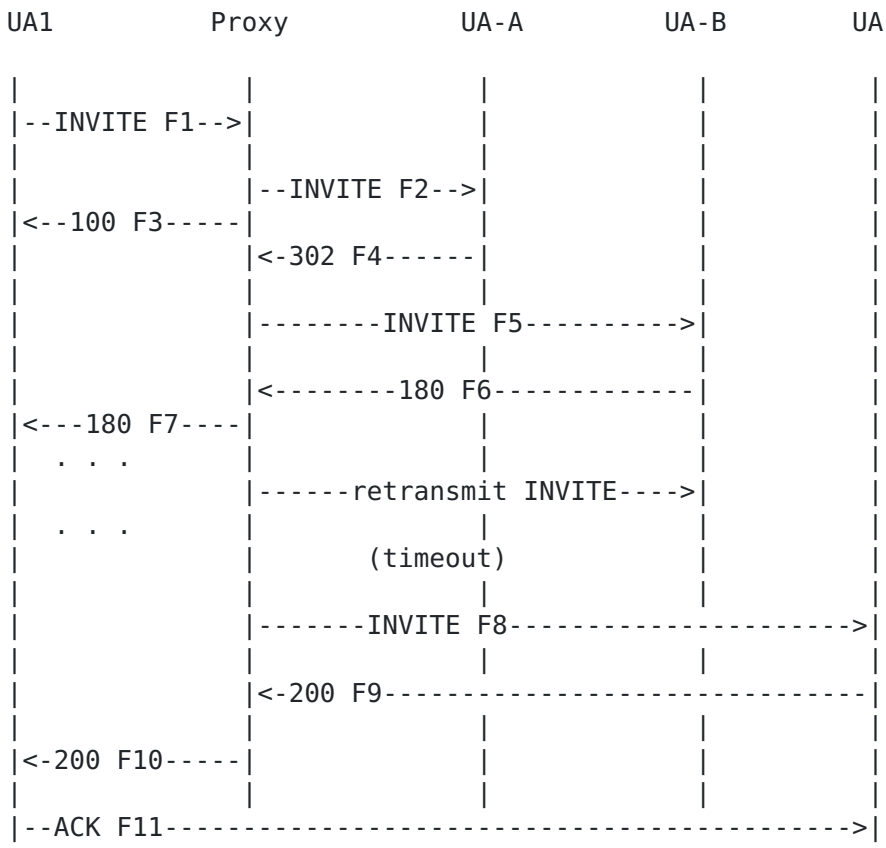


Parallel Forking Call Flow with History-Info

A.3. 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 isn't 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 makes use of.

UA1 called UA A, which had been forwarded to UA B, which forwarded to a UA 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.



Message Details

F1 INVITE UA1->Proxy

```
INVITE sip:UserA@example.com SIP/2.0
Via: SIP/2.0/UDP example.net:5060
From: BigGuy <sip:User1@example.net>
To: LittleGuy <sip:UserA@example.com>
Call-Id: 12345600@example.net
CSeq: 1 INVITE
Contact: BigGuy <sip:User1@example.net>
Content-Type: application/sdp
Content-Length: <appropriate value>

v=0
o=UserA 2890844526 2890844526 IN IP4 client.example.net
s=Session SDP
c=IN IP4 192.0.2.3
t=0 0
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

```
   /*Client for UA1 prepares to receive data on port 49170
     from the network. */

   F2 INVITE Proxy->UA-A

   INVITE sip:UserA@ims.example.com SIP/2.0
   Via: SIP/2.0/UDPims.example.com:5060;branch=1
   Via: SIP/2.0/UDP example.net:5060
   Record-Route: <sip:UserA@example.com>
   From: BigGuy <sip:User1@example.net>
   To: LittleGuy <sip:UserA@example.com>
   Call-Id: 12345600@example.net
   CSeq: 1 INVITE
   History-Info: <sip:UserA@ims.example.com>; index=1
   Contact: BigGuy <sip:User1@example.net>
   Content-Type: application/sdp
   Content-Length: <appropriate value>

   v=0
   o=UserA 2890844526 2890844526 IN IP4 client.example.net
   s=Session SDP
   c=IN IP4 192.0.2.3
   t=0 0
   m=audio 49170 RTP/AVP 0
   a=rtpmap:0 PCMU/8000


   F3 100 Trying Proxy->UA1

   SIP/2.0 100 Trying
   Via: SIP/2.0/UDP example.net:5060
   From: BigGuy <sip:User1@example.net>
   To: LittleGuy <sip:UserA@example.com>
   Call-Id: 12345600@example.net
   CSeq: 1 INVITE
   Content-Length: 0


   F4 302 Moved Temporarily UserA->Proxy

   SIP/2.0 302 Moved Temporarily
   Via: SIP/2.0/UDP ims.example.com:5060;branch=1
   Via: SIP/2.0/UDP example.net:5060
   From: BigGuy <sip:User1@example.net>
   To: LittleGuy<sip:UserA@example.com>;tag=3
   Call-Id: 12345600@example.net
   CSeq: 1 INVITE
   Contact: <sip:UserB@example.com>
```

```
   Content-Length: 0


   F5 INVITE Proxy-> UA-B

   INVITE sip:UserB@example.com SIP/2.0
   Via: SIP/2.0/UDP ims.example.com:5060;branch=2
   Via: SIP/2.0/UDP example.net:5060
   From: BigGuy <sip:User1@example.net>
   To: LittleGuy <sip:UserA@example.com>
   Call-Id: 12345600@example.net
   History-Info: <sip:UserA@ims.example.com?Reason=SIP;\
                 cause=302;text="Moved Temporarily">;index=1,\
                 <sip:UserB@example.com>;index=2
   CSeq: 1 INVITE
   Contact: BigGuy <sip:User1@example.net>
   Content-Type: application/sdp
   Content-Length: <appropriate value>

   v=0
   o=User1 2890844526 2890844526 IN IP4 client.example.net
   s=Session SDP
   c=IN IP4 192.0.2.3
   t=0 0
   m=audio 49170 RTP/AVP 0
   a=rtpmap:0 PCMU/8000


   F6 180 Ringing UA-B ->Proxy

   SIP/2.0 180 Ringing
   Via: SIP/2.0/UDP example.net:5060
   From: BigGuy <sip:User1@example.net>
   To: LittleGuy <sip:UserA@example.com>;tag=5
   Call-ID: 12345600@example.net
   CSeq: 1 INVITE
   Content-Length: 0


   F7 180 Ringing Proxy-> UA1

   SIP/2.0 180 Ringing
   SIP/2.0/UDP example.net:5060
   From: BigGuy <sip:User1@example.net>
   To: LittleGuy <sip:UserA@example.com>
   Call-Id: 12345600@example.net
   CSeq: 1 INVITE
   Content-Length: 0
```

```
   /* User B is not available.  INVITE is sent multiple
      times until it times out. */

   /* The proxy forwards the INVITE to UA-VM after adding the
      additional History Information entry. */

   F8 INVITE Proxy -> UA-VM

   INVITE sip:VM@example.com SIP/2.0
   Via: SIP/2.0/UDP ims.example.com:5060;branch=3
   Via: SIP/2.0/UDP example.net:5060
   From: BigGuy <sip:User1@example.net>
   To: LittleGuy <sip:UserA@example.com>
   Call-Id: 12345600@example.net
   History-Info: <sip:UserA@ims.example.com?Reason=SIP;\
                 cause=302; text="Moved Temporarily">;index=1,\
                 <sip:UserB@example.com?Reason=SIP;cause=480;\
                 text="Temporarily Unavailable">;index=2,\
                 <sip:VM@example.com>;index=3
   CSeq: 1 INVITE
   Contact: BigGuy <sip:User1@example.net>
   Content-Type: application/sdp
   Content-Length: <appropriate value>

   v=0
   o=User1 2890844526 2890844526 IN IP4 client.example.net
   s=Session SDP
   c=IN IP4 192.0.2.3
   t=0 0
   m=audio 49170 RTP/AVP 0
   a=rtpmap:0 PCMU/8000
```

F9 200 OK UA-VM -> Proxy

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ims.example.com:5060;branch=3
Via: SIP/2.0/UDP example.net:5060
From: BigGuy <sip:User1@example.net>
To: LittleGuy <sip:UserA@example.com>;tag=3
Call-Id: 12345600@example.net
CSeq: 1 INVITE
Contact: TheVoiceMail <sip:VM@example.com>
Content-Type: application/sdp
Content-Length: <appropriate value>

v=0
o=UserA 2890844527 2890844527 IN IP4 vm.example.com
s=Session SDP
c=IN IP4 192.0.2.4
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

F10 200 OK Proxy -> UA1

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ims.example.com:5060;branch=3
Via: SIP/2.0/UDP example.net:5060
From: BigGuy <sip:User1@example.net>
To: LittleGuy <sip:UserA@example.com>;tag=3
Call-Id: 12345600@example.net
CSeq: 1 INVITE
Contact: TheVoiceMail <sip:VM@example.com>
Content-Type: application/sdp
Content-Length: <appropriate value>

v=0
o=UserA 2890844527 2890844527 IN IP4 vm.example.com
s=Session SDP
c=IN IP4 192.0.2.4
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

```
F11 ACK UA1 -> UA-VM

ACK sip:VM@example.com SIP/2.0
Via: SIP/2.0/UDP example.net:5060
From: BigGuy <sip:User1@example.net>
To: LittleGuy<sip:UserA@example.com>;tag=3
Call-Id: 12345600@example.net
CSeq: 1 ACK
 Content-Length: 0

/* RTP streams are established between UA1 and
   UA-VM. UA-VM starts announcement for UA1 */
```

A.4. Automatic Call Distribution

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 (ACDGRP1) were busy, by retargeting the request to the Silver Group. 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 the calling UA.

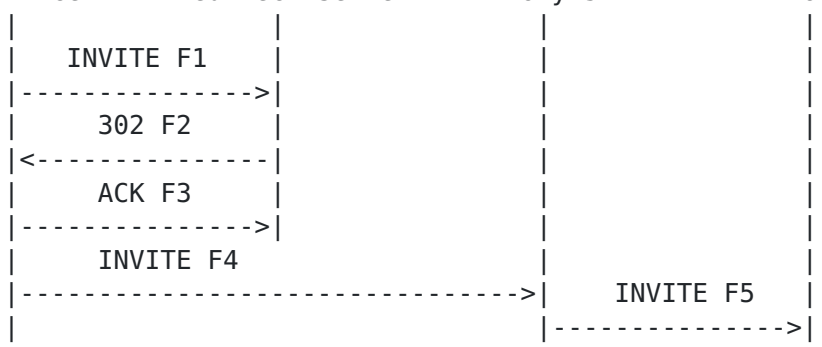
As with the other examples, this is not prescriptive of how one would do this type of service but an example of a subset of processing that might be associated with such a service. In addition, this example is not addressing any aspects of Agent availability, which might also be done via a SIP interface.

A.5. Session via Redirect and Proxy Servers

The scenarios in this Appendix A provide sample use cases for the History-Info header for informational purposes only. They are not intended to be normative and the formatting is for visual purposes; thus, the headers in the URI are not shown properly formatted for escaping. Refer to Section 4.2 examples with the proper formatting.

In this scenario, Alice places a call to Bob using first a Redirect server then a Proxy Server. The INVITE message is first sent to the

Redirect Server. The Server returns a 302 Moved Temporarily response (F2) containing a Contact header with Bob's current SIP address. Alice then generates a new INVITE with Bob's current SIP address included in another History-Info entry. The INVITE is then sent to Bob via the Proxy Server, with Bob receiving the complete History information; the call then proceeds normally. The complete call flow for this scenario, without the use of History-Info, is described in Section 3.6 of the SIP Basic Call Flow Examples [RFC3665].



Message Details

F1 INVITE Alice -> Redirect Server

```
INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060;branch=z9hG4bKbf9f44
Max-Forwards: 70
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@atlanta.example.com
CSeq: 1 INVITE
History-Info: <sip:bob@biloxi.example.com>; index=1
Contact: <sip:alice@client.atlanta.example.com>
Content-Length: 0
```

F2 302 Moved Temporarily Redirect Proxy -> Alice

```
SIP/2.0 302 Moved Temporarily
Via: SIP/2.0/UDP client.atlanta.example.com:5060;\
     branch=z9hG4bKbf9f44;received=192.0.2.1
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>;tag=53fHlqlQ2
Call-ID: 2xTb9vxSit55XU7p8@atlanta.example.com
CSeq: 1 INVITE
History-Info: <sip:bob@biloxi.example.com>; index=1
Contact: <sip:bob@chicago.example.com;transport=tcp>
Content-Length: 0
```

F3 ACK Alice -> Redirect Server

```
ACK sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060;branch=z9hG4bKbf9f44
Max-Forwards: 70
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>;tag=53fHlqlQ2
Call-ID: 2xTb9vxSit55XU7p8@atlanta.example.com
CSeq: 1 ACK
Content-Length: 0
```

F4 INVITE Alice -> Proxy 3

```
INVITE sip:bob@chicago.example.com SIP/2.0
Via: SIP/2.0/TCP client.atlanta.example.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@atlanta.example.com
CSeq: 2 INVITE
History-Info: <sip:bob@biloxi.example.com?Reason=SIP;cause=302>\
              text="Moved Temporarily">; index=1,
              <sip:bob@chicago.example.com>; index=2
Contact: <sip:alice@client.atlanta.example.com;transport=tcp>
Content-Length: 0
```

F5 INVITE Proxy 3 -> Bob

```
INVITE sip:bob@client.chicago.example.com SIP/2.0
Via: SIP/2.0/TCP ss3.chicago.example.com:5060;branch=z9hG4bK721e.1
Via: SIP/2.0/TCP client.atlanta.example.com:5060;\
     branch=z9hG4bK74bf9;received=192.0.2.1
Max-Forwards: 69
Record-Route: <sip:ss3.chicago.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@atlanta.example.com
CSeq: 2 INVITE
History-Info: <sip:bob@biloxi.example.com?Reason=SIP;cause=302>\
              text="Moved Temporarily">; index=1,
              <sip:bob@chicago.example.com>; index=2,
              <sip:bob@client.chicago.example.com>; index=2.1
Contact: <sip:alice@client.atlanta.example.com;transport=tcp>
Content-Length: 0
```

Authors' Addresses

Mary Barnes
Nortel
Richardson, TX

Email: mary.barnes@nortel.com

Francois Audet
Nortel
4655 Great America Parkway
Santa Clara, CA 95054
US

Email: audet@nortel.com