SIP Working Group
Internet Draft
Document: <draft-ietf-sip-privacy-02.txt>

W. Marshall
AT&T

K. Ramakrishnan
TeraOptic Networks

E. Miller
G. Russell
CableLabs

B. Beser
Pacific Broadband

M. Mannette
K. Steinbrenner
3Com

D. Oran
F. Andreasen
Cisco

J. Pickens
Com21

P. Lalwaney
Nokia

J. Fellows
Motorola

D. Evans
D. R. Evans Consulting

K. Kelly
NetSpeak

M. Watson
Nortel Networks

May 20, 2001

SIP Extensions for Caller Identity and Privacy

Status of this Memo

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.

The distribution of this memo is unlimited. It is filed as <draft-ietf-sip-privacy-02.txt>, and expires November 20, 2001. Please send comments to the authors.

1. Abstract

This document describes extensions to the Session Initiation Protocol (SIP) that allow parties in a SIP session to be identified by different types of party information such as calling and called party. For each type of party information, different types of identity information, e.g. subscriber, or terminal, can be provided. The extensions can furthermore be used in an environment where one or more proxies serve as trusted intermediaries which can provide and verify the identity of the parties either directly or indirectly, while still respecting desired privacy. Each party can specify the level of privacy that should be afforded them, from simple suppression of party information to full IP address privacy.

2. Conventions used in this document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC-2119 [2].

3. Introduction

In order for SIP [4] to be a viable alternative to the current PSTN,

SIP must support certain popular telephony services as well as some regulatory and public safety requirements. These include Calling Identity Delivery services, Calling Identity Delivery Blocking, and the ability to trace the originator of a call. While SIP can support each of these services independently, certain combinations cannot be supported. For example, a caller that wants to maintain privacy and consequently provides unintelligible information in the SIP From header field will not be identifiable. However, since the contents of the From header field cannot be modified, this will prevent certain services, e.g. return call or call trace, to be performed by entities more than a single hop away. We note that this problem is not telephony specific but applies to other forms of session

initiation as well. Furthermore, the issue of privacy in an IP environment is more complicated than in the PSTN. The caller and callee will normally exchange IP traffic directly, and IP address information itself may reveal some privacy. The issue of IP address privacy for both the caller and callee consequently needs to be addressed as well.

In order to solve the above we assume an architecture where a SIP User Agent is associated with a trusted proxy, and proxies in turn communicate with other proxies and user agents which may or may not be trusted. This is illustrated in the Figure below:

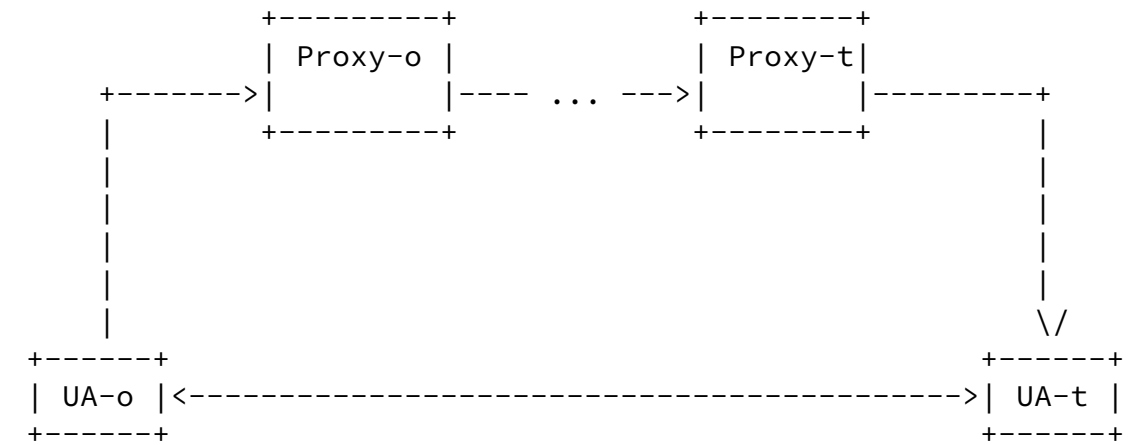


Figure 1 - Basic Architecture

Calls utilizing the services of this architecture must both be placed and received through the trusted proxy for the UA.

This document defines two extensions to SIP that allow the parties to be identified by a trusted intermediary while still being able to maintain their privacy.

The first one is a new general header, Remote-Party-ID, which identifies each party. Different types of party information can be provided, e.g. calling, or called party, and for each type of party, different types of identity information, e.g. subscriber, or terminal, can be provided. Since a party may not wish to reveal some or all of this information to an untrusted entity, the party can request a specific level of privacy for each. The intermediary also has the ability to specify a required level of privacy.

The second extension is a new general header, Anonymity, which defines other types of privacy requested by the party. Currently, the only such type is IP address privacy.

The trusted intermediary verifies the Remote-Party-ID information supplied and ensures the privacy requested, including IP address privacy, is provided when forwarding a message across an untrusted boundary. Remote-Party-ID information that was not successfully

verified, is tagged with an indication of this, so the receiver knows whether it should trust the information or not.

This document defines a set of party types and identity information. New types of party and identity information as well as other attributes may be introduced, thereby allowing new services to make use of the generic party information, privacy and authenticity handling defined here.

4. Protocol Overview

UACs that wish to use the extensions defined here MUST initiate calls through their trusted proxy and include a Proxy-Require header in the initial INVITE request containing the option tag "privacy". When such a UAC initiates a call, it SHOULD include a calling subscriber Remote-Party-ID header field in the initial INVITE request in order to identify the originator of the call. This Remote-Party-ID MUST contain a SIP-URL identifying the UAC and MAY contain a "display-name" for the UAC as well. The party type SHOULD be set to "calling" and the identity type SHOULD be set to "subscriber", however other types of party and identity information may be included as well. If Remote-Party-ID privacy is desired, the

UAC MUST include a privacy token set to one or more of "uri", "name" or "full". If IP address privacy is desired, the UAC MUST include an Anonymity header set to "ipaddr". Note that if the UAC does not initiate the call through its trusted proxy, the requested privacy may not be provided.

When a proxy supporting this extension receives an INVITE from an untrusted entity, it looks for the presence of a Remote-Party-ID header with calling subscriber information. If one is found, the proxy determines if the previous hop was a UA the proxy serves. If so, the calling subscriber Remote-Party-ID information is verified and modified if needed. If the request instead came from another untrusted entity, the proxy either removes the calling subscriber Remote-Party-ID information or marks it as being untrusted. Alternatively, the proxy MAY reject the request, e.g. with a 403 or 407. Other types of Remote-Party-ID information may be present as well. For each of these, the proxy MUST mark the information as being untrusted, if the request was received from an untrusted entity and the proxy itself can not verify the information supplied. Additional procedures may be defined as well.

Prior to forwarding the request to an untrusted entity, the proxy MUST look for the presence of a privacy request indication in each Remote-Party-ID header field. If one is found, the privacy requested MUST be provided for that Remote-Party-ID header field prior to forwarding the request. For uri and name privacy, this typically involves encrypting and possibly removing information provided in the Remote-Party-ID. The proxy MUST also look for the presence of an Anonymity header requesting IP address privacy. If IP Address privacy is requested, the proxy MUST ensure that IP address privacy

is provided through a level of indirection for signaling and media. We refer to the function that provides this level of indirection as an Anonymizer. The Anonymity header MUST be removed as well.

Once a UAS supporting this extension receives the INVITE through its trusted proxy, it can use the calling subscriber Remote-Party-ID information provided to identify the originator of the call, unless the originator had requested privacy. If the INVITE contained a Proxy-Require with an option tag of "privacy", the UAS SHOULD include a called subscriber Remote-Party-ID identifying itself in the first non-100 response. The party information SHOULD be set to "called" and the identity information SHOULD be set to "subscriber". Additional Remote-Party-ID header fields may be provided as well. If the UAS desires privacy for a Remote-Party-ID, it MUST include a

privacy request indication in that Remote-Party-ID header. If the UAS desires IP address privacy, the UAS MUST include an Anonymity header indicating this. Note that if the UAS did not receive the call through its trusted proxy, any Remote-Party-Id information provided may be false, and any privacy requested by the UAS may not be provided.

The UAS MAY also include Remote-Party-ID headers in subsequent provisional and final responses to the INVITE. The UAS SHOULD include a called party Remote-Party-ID header if the contents are different than sent in a previous response. The party information SHOULD be set to "called" and the identity information SHOULD be set to "subscriber". Additional Remote-Party-ID header fields may be provided as well.

When a proxy supporting this extension receives a non-100 response to the initial INVITE, it looks for a Remote-Party-ID header field and applies similar processing as for the initial INVITE with one difference. If the INVITE did not contain a Proxy-Require with an option tag of "privacy", the proxy MUST ensure that any privacy requested in the response is provided prior to forwarding it, irrespective of whether the previous hop is trusted or not.

Finally, when the UAC receives the first non-100 response from the UAS through the UAC's trusted proxy, the UAC can use the called subscriber Remote-Party-ID information provided to identify the called party, unless the terminator had requested privacy. Subsequent non-100 responses MAY contain Remote-Party-ID information as well. When the UAC receives the final 200 response from its trusted proxy, it MAY contain a called subscriber Remote-Party-ID header identifying the party the UAC was connected to.

5. Header Field Definitions

Table 1 below is an extension of tables 4 and 5 in [4] for the new headers defined here:

	where	enc.	e-e	ACK	BYE	CAN	INV	OPT	REG
Anonymity	g	n	h	-	-	-	o	-	-
Remote-Party-ID	g	n	h	-	-	-	o	-	-

Table 1: Summary of header fields.

The headers can be used in an INVITE as well as any response to an INVITE. Note that any privacy requested may not be honored unless the request or response is sent through the UA's trusted proxy. Similarly, Remote-Party-ID information may not be trustworthy if it was received in a request or response from anybody else than the trusted proxy.

5.1 Remote-Party-ID Header Field Definitions

The Remote-Party-ID header field provides information about the remote party. Different types of party information can be provided, e.g. calling and called, and for each, different types of identity information can be provided as well. A request or response MAY contain more than one Remote-Party-ID header field with privacy requested independently for each. Remote-Party-ID is defined by the following ABNF [3]:

```
  Remote-Party-ID    = "Remote-Party-ID" ":" [display-name]
                             "<" addr-spec ">" *(";" rpi-token)

  rpi-token          = rpi-screen | rpi-pty-type |
                        rpi-id-type | rpi-privacy | other-rpi-token

  rpi-screen         = "screen" "=" ("no" | "yes" )

  rpi-pty-type       = "party" "=" ( "calling" | "called"  | token )

  rpi-id-type        = "id-type" "=" ( "subscriber" | "user" |
                             "alias" | "return" | "term"  | token )

  rpi-privacy        = "privacy" "=" 1#(
                         ("full" | "name" | "uri" | "off" | token )
                         [ "-" ( "network" | token ) ]  )

  other-rpi-token    = ["-"] token ["=" (token | quoted-string)]
```

Furthermore, we define the value "private" for "other-user" in an "addr-spec", to indicate that the user part of an "addr-spec" is in a non-intelligible form. The syntax for "other-user" is therefore refined to:

```
  other-user         = token  | "private"
```

Comparisons follow the case-sensitivity rules defined by SIP [4].

The "display-name" in Remote-Party-ID is a text string that identifies the name of the party. The "addr-spec" contains information identifying the party either in clear-text or encrypted form. In the latter case, the "user" part of the "addr-spec" typically contains the encrypted party information, whereas the "hostport" identifies the entity that can decrypt the information. Furthermore, an "other-user" value of "private" will then be present to indicate that the "addr-spec" is non-intelligible. Depending on the rpi-pty-type, the "addr-spec" can be used as the Request-URI by the UA to initiate certain call control functions or subsequent calls that are required to reference the party.

The rpi-screen parameter describes what verification the Remote-Party-ID information has undergone. The value "yes" indicates the Remote-Party-ID was verified successfully by the proxy itself or the proxy received the message from a trusted entity with this indication. The value "no" (assumed by default) indicates the Remote-Party-ID was either not verified successfully by the proxy or the proxy received the message from an untrusted entity. Multiple rpi-screen parameters MAY be present in a Remote-Party-ID - if both "yes" and "no" are present, "no" will take precedence.

The rpi-pty-type describes the type of party to which this header refers. There MUST NOT be more than one rpi-pty-type present in a Remote-Party-ID. If absent, the parameter describes the party from which it was received, i.e. "calling" party in the case of INVITE, and "called" in the case of responses. Additional values may be defined as extensions. Such extensions shall be registered with IANA.

The rpi-id-type describes the nature of the identity provided. Several types of identity may be provided for each party, however there MUST NOT be more than one rpi-id-type present in a given Remote-Party-ID. If the rpi-id-type parameter is absent, the Remote-Party-ID contains the subscriber identity.

This document defines five identity types - additional values may be defined as extensions in which case they shall be registered with IANA. The types are defined based on the nature of the information they represent, as opposed to a particular application. Individual applications requiring identity information will specify which identities should be used for that application:

Subscriber identity (rpi-id-type="subscriber"):

  This identifies the owner of the subscription which is being used for the session.

User identity (rpi-id-type="user"):

This identifies the individual participating in the session. For example when multiple individuals are able to originate sessions

under the same subscription. If absent, this is assumed to be the same as the subscriber identity.

Alias identity (rpi-id-type="alias"):

This is an identity which the party has asked to be identified by. For example a user may have a 'business' and a 'personal' address, and may ask to be identified by one of these when initiating a session using the other subscription. If absent, this is assumed to be the same as the subscriber identity.

Return session identity (rpi-id-type="return")

This identifies the point to which the party wishes return sessions to be addressed. For example a business may wish to provide a 'freephone' identity to encourage return calls. If absent, this is assumed to be the same as the subscriber identity.

Terminal identity (rpi-id-type="term")

This identifies the terminal being used for the session. For example several users may be able to 'log in' to a single terminal, in which case the identity of the terminal will differ from that of the user, subscriber, etc. If absent, this is assumed to be the same as the subscriber identity.

UAs SHOULD NOT include multiple Remote-Party-ID headers containing the same identity information marked with different rpi-id-types.

The rpi-privacy parameter describes whether the identity information must be hidden from untrusted entities. There MAY be multiple rpi-privacy parameters in a Remote-Party-ID. If privacy is requested, it MUST be one or more of "full", "uri", or "name". The value "full" requests that both the "display-name" and the "addr-spec" must be hidden. The values "name" and "uri" request that the "display-name" or the "addr-spec" must be hidden respectively. The value "off" indicates that lack of privacy is explicitly requested, and MUST be the only value if present. The values may be postfixed with a string indicating that the privacy request was made by an entity other than

the party itself. Postfixing with the value "-network" indicates that the network proxies have requested that the information be hidden. Additional values may be defined as extensions. Such extensions shall be registered with IANA.

It should be noted, that an entity requesting only Remote-Party-ID privacy will not receive complete privacy. The values "uri" and "name" merely affect information that may be displayed as opposed to truly hiding the identity of the requesting entity since the identity of the host, e.g. IP address, is not hidden. For full privacy, the entity should request IP address privacy as well - see Section 5.2.

Finally, the "other-rpi-token" parameter allows Remote-Party-ID to be extended with other types of parameters which shall be registered with IANA. By default, such extensions will be assumed to contain information that may be of importance to the verification, and hence have to be supported for verification to pass successfully. The prefix "-" is used to indicate that a parameter extension does not need to be supported by a given entity in order for the Remote-Party-ID to be verified successfully. Consequently, such extensions MUST NOT begin with the character "-".

5.2 Anonymity Header Field Definition

The Anonymity header field allows a SIP user agent to indicate the degree of other privacy that should be provided to its session. The only type of other privacy defined here is IP address privacy.

The ABNF for the header field follows:

```
Anonymity          = "Anonymity" ":"  1#privacy-tag
privacy-tag        = "ipaddr" | "off" | token
```

Comparisons follow the case-sensitivity rules defined by SIP [4].

If privacy is requested, it MUST currently be "ipaddr" (extensions may change this). The value "off" indicates that no privacy is requested, and MUST be the only value if present. Additional values may be defined as extensions. Such extensions shall be registered with IANA.

The value "ipaddr" requests IP privacy such that the other party

does not learn the IP address of this party.

It should be noted, that an entity requesting only IP address privacy merely hides its IP address without suppressing its identity. For full privacy, the entity should thus also request privacy for its Remote-Party-ID information. Note however, that the use of extensions that do not consider privacy impacts, may in turn violate privacy.

The value "off" indicates no privacy is requested, and MUST be the only value if present.

Absence of the Anonymity header in a request or response is identical to the value "off", unless provisioned otherwise.

It should be noted, that the Anonymity header field allows both the originating and terminating UA to indicate its desire for IP address privacy.

6. Protocol Semantics

Below, we provide the protocol semantics for a UAC, a UAS, and a proxy.

6.1 UAC Behavior

When a UAC supporting this extension initiates a call through its trusted proxy, it SHOULD include a calling subscriber Remote-Party-ID header in the initial INVITE request in order to identify the originator of the call. The Remote-Party-ID header MUST at a minimum contain an "addr-spec" to uniquely identify the calling subscriber. The "addr-spec" SHOULD be the same string as appears in the Request-URI for incoming call attempts. The Remote-Party-ID SHOULD include an rpi-pty-type set to "calling" and an rpi-id-type set to "subscriber" - we refer to this as a calling subscriber Remote-Party-ID. The rpi-screen parameter SHOULD NOT be included. The Remote-Party-ID MAY optionally include a "display-name" which SHOULD be set to a name that the proxy has associated with the calling subscriber, e.g. the subscribers full name. The UAC MAY include other Remote-Party-ID information as well.

If the UAC desires Remote-Party-ID privacy for the call, it MUST include an rpi-privacy parameter for each relevant Remote-Party-ID. Note that if the UAC does not initiate the call through its trusted proxy, the requested privacy may not be provided. The rpi-privacy parameter MUST specify the desired level of privacy, e.g. "uri", to maintain privacy of the "addr-spec". As honoring the privacy requested depends on the proxy, the UAC MUST furthermore include a Proxy-Require header with an option-tag of "privacy". Should a 420 response listing "privacy" as an unsupported option be returned, then privacy can not be provided for this call. The UA must then either initiate a new session without requiring privacy, or the session initiation attempt must be abandoned.

If the UAC desires "name" or "full" privacy, the UAC MUST NOT reveal the originating subscriber's name in the "display-name" portion of any other header than Remote-Party-ID. This can be achieved by either not providing a "display-name" or setting the "display-name" to "anonymous" in such fields, e.g. From and Contact.

If the UAC desires "uri" or "full" privacy, the UAC MUST NOT reveal the subscriber's identity in any other header field than Remote-Party-ID. In particular, the contents of header fields needs to be considered as described below:

* From: The UAC SHOULD supply a cryptographically random identifier for the userinfo, and a non-identifying hostname, e.g. "localhost", in the host name.

* To: If a telephone number is used in the addr-spec, the telephone number SHOULD be a full E.164 number including the country code that is different from the From header field. If a

host name is included, it SHOULD be the non-identifying name "localhost".

* Contact: The same cryptographically random identifier used in the From header field SHOULD be supplied for the userinfo, and an IP-address SHOULD be used in the host name.

* All other headers that may contain either an IP address or a domain name, e.g. Call-ID, and Via, SHOULD use the IP-address form. It should however be noted, that this simple privacy step may be overcome fairly easily in many cases.

The UAC may also explicitly request that privacy is not to be provided for a Remote-Party-ID by setting the rpi-privacy parameter to "off". This is also the default value, unless provisioned otherwise.

If the UAC desires IP address privacy, it MUST include an Anonymity header field set to "ipaddr". The value "off", which is the default unless provisioned otherwise, may be provided if IP address privacy is explicitly not requested. Note that if the UAC does not initiate the call through its trusted proxy, the requested privacy may not be provided. As honoring the privacy requested depends on the proxy, the UAC MUST furthermore include a Proxy-Require header with an option-tag of "privacy". Should a 420 response listing "privacy" as an unsupported option be returned, then privacy can not be provided for this call. The UA must then either initiate a new session without requiring privacy, or the session initiation attempt must be abandoned.

If the UAC desires "ipaddr" privacy, then the following header field requirements apply:

* From: The UAC MUST use the non-identifying host name "localhost".

* Call-ID: The UAC MUST NOT base the Call-ID on the originator's IP address.

The first non-100 response received by the UAC through its trusted proxy MAY contain one or more Remote-Party-ID header fields. A Remote-Party-ID with party type "called" will identify the called party. If such a Remote-Party-ID header field either does not contain an rpi-screen parameter, or it contains an rpi-screen parameter with the value "no" (this includes the case where both "yes" and "no" is provided), the UAC SHOULD NOT trust the validity of the information provided. An end-to-end encrypted Remote-Party-ID header field can of course also not be trusted, regardless of the value of the rpi-screen parameter.

Subsequent responses received by the UAC MAY also contain Remote-Party-ID header fields. Such Remote-Party-Ids with party type

"called" identify other parties to which the session has been directed, for whatever reason.

Remote-Party-ID headers contained in the final response, with rpi-pty-type set to "called" identify the party which has answered the session. Again, end-to-end encrypted Remote-Party-ID header fields can not be trusted.

6.2 UAS Behavior

A UAS supporting this extension and receiving an INVITE from its trusted proxy looks for a Remote-Party-ID header field with rpi-pty-type "calling" and rpi-id-type "subscriber", i.e. a calling subscriber Remote-Party-ID, to identify the originator of the request. If rpi-pty-type is omitted from a Remote-Party-ID in the INVITE, "calling" is assumed, and if rpi-id-type is omitted, "subscriber" is assumed. If a calling subscriber Remote-Party-ID either does not contain an rpi-screen parameter or it contains an rpi-screen parameter with a value of "no" (this includes the case where both "yes" and "no" is provided), the UAS SHOULD NOT trust the validity of the information provided. An end-to-end encrypted Remote-Party-ID header field can of course also not be trusted, regardless of the value of the rpi-screen parameter. Otherwise, the UAS SHOULD use the information provided to identify the calling party rather than any information provided in the From or any other header field. Note that the INVITE MAY contain other Remote-Party-ID header fields.

If the initial INVITE contained a Proxy-Require header field with an option tag of "privacy", the UAS SHOULD insert a called subscriber Remote-Party-ID header field identifying itself into the first non-100 response it sends back through its trusted proxy. The called subscriber Remote-Party-ID SHOULD contain an rpi-pty-type of "called" and an rpi-id-type of "subscriber. Otherwise, the rules for the Remote-Party-ID are similar to those for the initial INVITE sent by a UAC. In addition, the UAS MAY insert Remote-Party-ID headers in any further responses. The UAS SHOULD insert a new called subscriber Remote-Party-ID header if the called party information changed from the called party information sent in the previous response. Note that the UAS may request privacy for the Remote-Party-ID information as well. Again, such privacy requests can only be assumed to be honored if the initial INVITE was received through its trusted proxy.

The UAS MAY request IP-address privacy by including an Anonymity header set to "ipaddr" in the first non-100 response it sends back through its trusted proxy. It should be noted though, that the UAS can not depend on this privacy being honored, if the original INVITE did not contain a Proxy-Require with an option tag of "privacy".

6.3 Proxy Behavior

When a proxy supporting this extension receives an INVITE from an untrusted entity, the proxy first determines if the request came from a UAC that it serves. If so, the proxy examines the INVITE for the presence of calling subscriber Remote-Party-ID header fields. If a calling subscriber Remote-Party-ID header field is present, the information supplied is verified and, if needed, rewritten. The proxy MUST verify that the "addr-spec" provided in a calling subscriber Remote-Party-ID is a valid "addr-spec" for that UAC; if not, the proxy MUST rewrite the "addr-spec" with a valid "addr-spec" for that UAC. If "display-name" is provided in a calling subscriber Remote-Party-ID, the proxy MUST verify that the "display-name" is a valid string for the UAC; if not or if the "display-name" is omitted, the proxy MUST rewrite the "display-name" with a valid string for the UAC or remove the "display-name". The proxy MUST also add an rpi-screen parameter with a value of "yes". If an rpi-screen parameter was already present in the calling subscriber Remote-Party-ID, it MUST be discarded.

If a Remote-Party-ID header was not present in the INVITE, but the proxy is able to identify the originating UAC anyway, the proxy inserts a Remote-Party-ID header with the correct information.

If the request instead came from an untrusted entity, and it was not a UAC the proxy serves or the proxy is unable to identify the entity, the proxy MUST either remove any calling subscriber Remote-Party-ID header or add "screen=no" before the request is forwarded. In the latter case, the proxy SHOULD furthermore ensure this is the only rpi-screen parameter. Alternatively, the proxy MAY reject the request, e.g. with a 403 or 407.

The INVITE MAY contain additional Remote-Party-ID header fields. When the request was received from an untrusted entity, each of these MUST be verified by the proxy. If the proxy is able to successfully verify the information in a Remote-Party-ID header field, the proxy MUST add an rpi-screen parameter set to "yes" for that Remote-Party-ID. Furthermore, this MUST be the only rpi-screen parameter for that Remote-Party-ID. If verification fails however, further processing depends on the reason for the failure. Two different failure reasons are defined here:

* The information provided could not be verified because the proxy does not support verification of this particular Remote-Party-ID.

* The information provided is incorrect and the proxy detected that.

In the first case, the proxy MUST add an rpi-screen parameter set to "no". The proxy SHOULD furthermore ensure this is the only rpi-screen parameter. In the second case, the proxy MUST by default add an rpi-screen parameter set to "no" and ensure this is the only rpi-screen parameter, however individual extensions and local procedures

MAY specify a different behavior, for example rewrite or removal of the offending Remote-Party-ID header field.

Note, that the proxy does not check a "display-name" provided in the From header field.

The proxy MUST furthermore look for the presence of a privacy request in any of the Remote-Party-ID headers as well as an Anonymity header. If privacy was requested, and the next hop is trusted, the proxy MUST ensure that a Proxy-Require header with an option-tag of "privacy" is present.

If the proxy forwards the request to an untrusted entity, and privacy was requested, the proxy MUST ensure the privacy requested will be honored.

For each Remote-Party-ID requesting privacy, the proxy MUST do the following:

* If rpi-privacy contains the value "full" or "uri", the proxy MUST replace the "addr-spec" in that Remote-Party-ID header field with a private "addr-spec". The private "addr-spec" MUST list the proxy itself in the hostport and include a "user=private" user parameter.

* If rpi-privacy contains the value "full" or "name", the proxy MUST delete the "display-name" in that Remote-Party-ID header field.

Generation of the user part of a private "addr-spec" is a proxy internal issue as long as the requested privacy is honored. However, it is RECOMMENDED to construct the user part by including:

* the initial "addr-spec",

* the value of rpi-privacy, and

* sufficient checksum information to prevent tampering by the untrusted party.

All of this information MUST then be encoded or encrypted such that the next hop is unable to discern the original Remote-Party-ID. It is RECOMMENDED that the string be encrypted with a symmetric private key, and converted to a printable string using Base64 encoding. The proxy MAY include other information in the user part as well.

For IP-address privacy, the proxy MUST ensure that the request is rewritten in a way that ensures that the IP-address of the originating UAC will not be revealed. This implies that neither SIP signaling nor IP media streams are exchanged directly between the UAC and UAS. A level of indirection which we call an Anonymizer MUST be provided.

Prior to forwarding the request to an untrusted entity, the proxy MAY remove any "privacy" option tag present in a Proxy-Require header field to prevent unnecessary failure of the request if downstream proxies do not support this extension. Note that this will unfortunately also prevent downstream proxies and UASs from determining if their previous hop supports the extension.

When receiving the first non-100 response to the initial INVITE from an untrusted entity, the proxy first determines if the response came from a UAS that it serves.

If it did, the proxy examines the response for the presence of a called subscriber Remote-Party-ID and privacy requests (incl. IP address privacy) and applies similar processing as for an INVITE received from a UAC served by the proxy. Furthermore, if the original INVITE did not contain a Proxy-Require header field with an option tag of "privacy", the proxy can not determine if the previous hop supports the privacy extension or not. Consequently, if the response contains a request for privacy, the privacy MUST be applied by this proxy, irrespective of whether the upstream hop is trusted or not.

If the response came from an untrusted entity, and it was not a UAS the proxy serves, the proxy MUST either remove any called subscriber Remote-Party-ID header fields provided or set "screen=no" before the response is forwarded upstream. The same action MUST be taken when

the initial INVITE did not contain a Proxy-Require with an option tag of "privacy", irrespective of whether the downstream hop was trusted or not. The reason for this is, that without the proxy-require in the initial INVITE, the proxy does not know if the downstream proxies performed proper privacy handling.

Finally, the response MAY contain additional Remote-Party-ID header fields. The proxy MUST apply similar processing as for an INVITE received from a UAC served by the proxy. In particular, the proxy MUST ensure it provides the proper screening indication of any Remote-Party-ID information received and performs the correct privacy handling as well.

6.4 Additional proxy behavior

A proxy supporting this extension SHOULD be prepared to receive a request containing a SIP-URL with a user parameter of "private". If the "hostport" part of the SIP-URL identifies the proxy handling the request, the proxy MUST recover the private information. For proxies that use the encryption recommendation provided earlier, this implies decrypting the "user" portion of the SIP-URL and replacing it with the decrypted SIP-URL that was contained in the "user" portion as well as any other SIP information included. Note that the decrypted SIP-URL may itself contain a "private" SIP-URL.

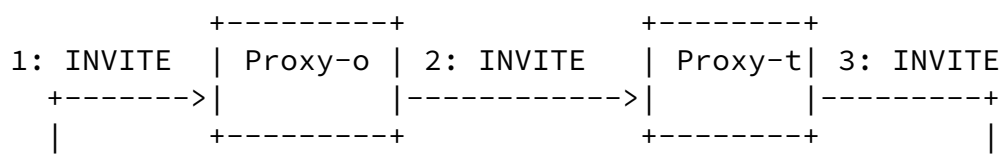
If the proxy is unable to recover a "private" SIP-URL, it MUST fail the request with a 4xx error code.

7 Example of Use

In this Section, we illustrate how the request for privacy may work in practice. It should be noted that the privacy service described can be implemented in a number of ways; we merely describe one possible solution in this section.

7.1 Basic Privacy Example

The Figure below illustrates a basic privacy example scenario



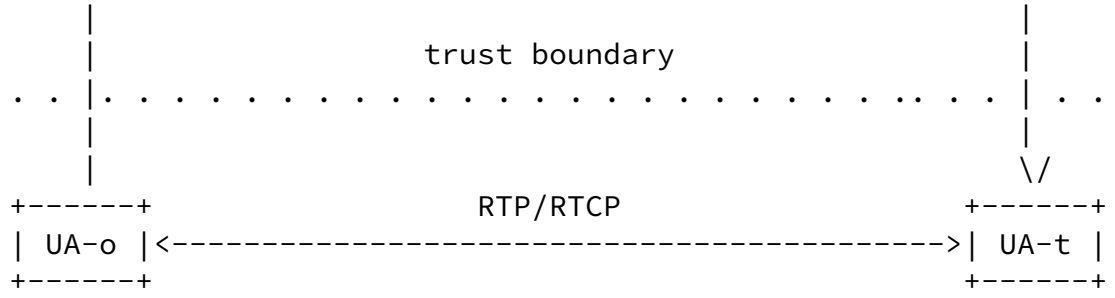


Figure 2 - Basic Privacy Example

The originating user agent (UA-o) sends an INVITE (1) to Proxy-o where it identifies itself and requests uri and name, i.e. full, privacy. Since the From header field contains calling identity information, UA-o supplies a cryptographically random identifier for the user info, and the non-identifying hostname "localhost" rather than its true identity:

```
INVITE
From:             sip:xyz@localhost
Remote-Party-ID: "John Doe" <sip:jdoe@foo.com>;party=calling;
                        id-type=subscriber;privacy=full
Proxy-Require:   privacy
```

Proxy-o verifies the calling subscriber information before it sends INVITE (2) to Proxy-t, which in this case is trusted. Since the calling subscriber Remote-Party-ID was verified successfully, Proxy-o adds an rpi-screen parameter set to "yes". When Proxy-t receives the INVITE, it examines the privacy request included in the INVITE and sees that uri and name privacy is requested. Proxy-t therefore removes the "display-name" from the calling subscriber Remote-Party-ID, encrypts the "addr-spec" and rpi-privacy, puts the result in the "user" part, inserts itself as the "hostport" and adds a "user=private" user parameter. Also, Proxy-t removes the Proxy-Require "privacy":


```
INVITE
From:             sip:xyz@localhost
Remote-Party-ID: <sip:e(<sip:jdoe@foo.com>;privacy=full
                        )@proxy-t.foo.com;user=private
                 >;party=calling;id-type=subscriber;
                                privacy=full;screen=yes
```

UA-t notes the presence of the Remote-Party-ID, but since it indicates full privacy, UA-t can only identify the calling subscriber as private, however it knows that the subscribers identity has been verified since the rpi-screen parameter is set to "yes". UA-t decides to accept the call setup, and responds with a 180 Ringing. In this case, there is no request for privacy, so only the called subscriber Remote-Party-ID of the called party is added:

```
180
Remote-Party-ID: <sip:mdoe@foo.com>;party=called;
                                          id-type=subscriber
```

Proxy-t verifies the information provided, adds the omitted "display-name" to the Remote-Party-ID as well as an rpi-screen parameter set to "yes". Since no privacy was requested, proxy-t can provide the Remote-Party-ID information to proxy-o in clear:

```
180
Remote-Party-ID: "Mary Doe" <sip:mdoe@foo.com>;party=called;
                                  id-type=subscriber;screen=yes
```

Proxy-o forwards the response to UA-o as is.

While this illustrates the basic operation of the service, there are additional issues that need to be considered. In SIP, there are several fields that can reveal the identity of the calling party, either in part or completely. Other protocols used, e.g. SDP and RTP may reveal identity information as well. A user agent wishing to not reveal its identity should consider each of these. The next example looks more closely at this.

7.2 Full Privacy Example

The second example we look at is one where IP-address privacy is requested. The Figure below illustrates how IP address privacy can be achieved by inserting a trusted intermediary, an anonymizer, for the media streams between UA-o and UA-t. The interface between the proxies and the media anonymizer is purposely not defined here:

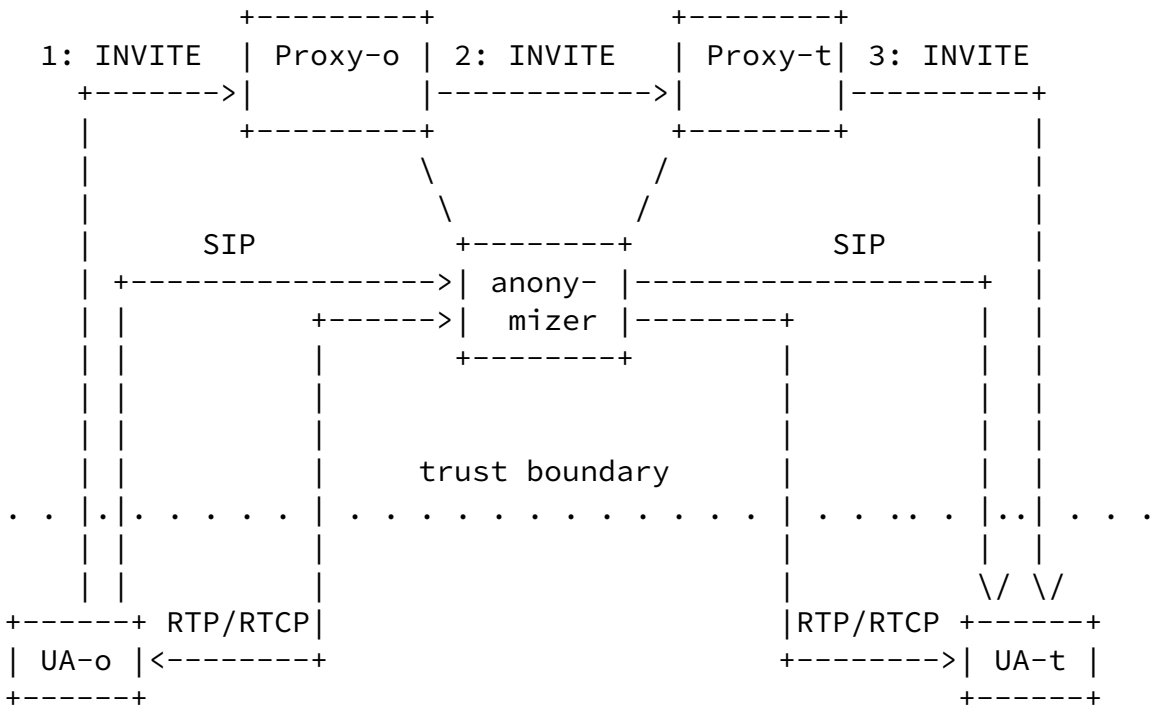


Figure 3 - Full Privacy Example

For all signaling and media exchange purposes, the anonymizer adds a level of indirection thereby hiding the IP address(es) of UA-o from UA-t. This indirection is used both for the media streams and SIP signaling, beyond the initial INVITE, exchanged directly between UA-o and UA-t.

In addition to the requirements listed earlier, the following commonly used header fields may reveal privacy information as well, which can be remedied as described:

* A Contact header field must be set to point to the anonymizer to prevent any direct signaling between UA-o and UA-t
* Via, Record-Route, Route, and any other header fields identifying either UA-o or Proxy-o must be hidden, e.g. by encryption or simple stateful removal and re-insertion by Proxy-t.

An alternative to the media anonymizer function shown above is to implement the anonymizer as a back to back User Agent thereby trivially hiding IP address information in the SIP signaling itself.

Furthermore, when SDP is used to describe the media in the session, the session descriptions exchanged by the user agents need to be modified to direct the media streams to the anonymizer. The use of SDP fields revealing calling identity information needs to be considered as well. Similar concerns apply to the use of RTCP.

8. Security Considerations

A user requesting complete privacy must still authenticate himself to the proxy, and therefore the SIP messages between the UA and the proxy MUST be protected. Likewise, it is necessary that the proxies take precautions to protect the user identification information from eavesdropping and interception. Use of IPSec between the UA and proxy as well as between proxies is recommended.

As noted above, Remote-Party-ID information received can only be trusted if it is received in clear-text through the UA's trusted proxy. Also, any privacy requested can only be assumed to be honored when the request is made through the trusted proxy. Thus if it is unknown whether a given request or response is sent or received through the trusted proxy, Remote-Party-ID information should be considered untrustworthy, and request for privacy should not be assumed to be honored.

9. Notice Regarding Intellectual Property Rights

AT&T may seek patent or other intellectual property protection for some or all of the technologies disclosed in the document. If any standards arising from this disclosure are or become protected by one or more patents assigned to AT&T, AT&T intends to disclose those patents and license them on reasonable and non-discriminatory terms. Future revisions of this draft may contain additional information regarding specific intellectual property protection sought or received.

3COM may seek patent or other intellectual property protection for some or all of the technologies disclosed in the document. If any standards arising from this disclosure are or become protected by one or more patents assigned to 3COM, 3COM intends to disclose those patents and license them on reasonable and non-discriminatory terms. Future revisions of this draft may contain additional information regarding specific intellectual property protection sought or received.

10. References

1. Bradner, S., "The Internet Standards Process -- Revision 3", BCP 9, RFC 2026, October 1996.

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

3 Crocker, D. and Overell, P.(Editors), "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, Internet Mail Consortium and Demon Internet Ltd., November 1997

4 M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg, "SIP: session initiation protocol," Request for Comments (Proposed Standard) 2543, Internet Engineering Task Force, Mar. 1999.

11. Acknowledgments

The Distributed Call Signaling work in the PacketCable project is the work of a large number of people, representing many different companies. The authors would like to recognize and thank the following for their assistance: John Wheeler, Motorola; David Boardman, Daniel Paul, Arris Interactive; Bill Blum, Jon Fellows, Jay Strater, Jeff Ollis, Clive Holborow, Motorola; Doug Newlin, Guido Schuster, Ikhlaq Sidhu, 3Com; Jiri Matousek, Bay Networks; Farzi Khazai, Nortel; John Chapman, Bill Guckel, Michael Ramalho, Cisco; Chuck Kalmanek, Doug Nortz, John Lawser, James Cheng, Tung-Hai Hsiao, Partho Mishra, AT&T; Telcordia Technologies; and Lucent Cable Communications.

12. Author's Addresses

Bill Marshall
AT&T
Florham Park, NJ 07932
Email: wtm@research.att.com

K. K. Ramakrishnan
TeraOptic Networks
Sunnyvale, CA

Email: kk@teraoptic.com

Ed Miller
CableLabs
Louisville, CO 80027
Email: E.Miller@Cablelabs.com

Glenn Russell
CableLabs
Louisville, CO 80027
Email: G.Russell@Cablelabs.com

Burcak Beser
Pacific Broadband Communications
San Jose, CA
Email: Burcak@pacband.com

Mike Mannette
3Com
Rolling Meadows, IL 60008
Email: Michael-Mannette@3com.com

Kurt Steinbrenner
3Com
Rolling Meadows, IL 60008
Email: Kurt-Steinbrenner@3com.com

Dave Oran
Cisco
Acton, MA 01720
Email: oran@cisco.com

Flemming Andreasen
Cisco
Edison, NJ
Email: fandreas@cisco.com

John Pickens
Com21
San Jose, CA
Email: jpickens@com21.com

Poornima Lalwaney
Nokia

San Diego, CA 92121
Email: poornima.lalwaney@nokia.com

Jon Fellows
Motorola
San Diego, CA 92121
Email: jfellows@gi.com

Doc Evans
D. R. Evans Consulting
Boulder, CO 80303
Email: n7dr@arrl.net

Keith Kelly
NetSpeak
Boca Raton, FL 33587
Email: keith@netspeak.com

Mark Watson
Nortel Networks
Maidenhead, UK
Email: mwatson@nortelnetworks.com

Appendix A: Nature of Party

This document defines a new "other-rpi-token" to identity the nature of the party in the Remote-Party-id. The Remote-Party-ID Nature of Party information (rpi-np) is supplied on a "token = value" form as defined by the following grammar:

```
rpi-np = "np" "=" ("ordinary" | "residential" | "business" |
                   "priority" | "hotel" | "failure" | "hospital" |
                   "prison" | "police" | "test" | "payphone" |
                   "coin" | "payphone-public" | "payphone-private" |
                   "coinless" | "restrict" | "coin-restrict" |
```

```
                    "coinless-restrict" | "reserved" | "operator" |
                    "trans-freephone" | "isdn-res" | "isdn-bus" |
                    "unknown" | "emergency" | token )
```

The rpi-np describes the nature of the party identified. Typically, this information will come from information digits (ID) which are used in the Public switched network in the US. Information digits are two digit codes that precede the Called Party Number and provide information to the Exchange carriers and IECs about the type of line that originated the call or any special characteristics of the Billing number. In the non-US environments, a parameter called the Calling Party Category (CPC) usually plays the role of Information digits in the US and provides similar information. Additional values can be defined as extensions.

The following example illustrates the use of the Nature of Party:

```
     Remote-Party-ID: "Mary Doe" <sip:mdoe@foo.com>;party=called;
                     id-type=subscriber;np=ordinary;screen=yes
```


Appendix B: IANA Considerations

The extensions defined in this document are extensible themselves.

Any extensions defined shall be registered with IANA as follows:

* rpi-id-type: A literal name must be provided.

* rpi-pty-type: A literal name must be provided.

* rpi-privacy: A new privacy indication or a new privacy postfix can be defined - each of these have a separate name space:

  * privacy indication: A literal name must be provided.

  * privacy postfix: A literal name must be provided.

* other-rpi-token: A literal name, which MUST NOT start with the dash character ("-"), must be provided. If the extension is on the form "type = value", then a description of the permissible values should furthermore be provided.

* privacy-tag: A literal name must be provided.

Also, each extension must have a designated contact person. Furthermore, if such extensions are to see any widespread and/or interoperable use, they should be properly defined and described in a publicly available document.

Full Copyright Statement

Expiration Date This memo is filed as <draft-ietf-sip-privacy-02.txt>, and expires November 20, 2001.