Network Working Group
Request for Comments: 3455
Category: Informational

M. Garcia-Martin
Ericsson
E. Henrikson
Lucent
D. Mills
Vodafone

January 2003

Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the 3rd-Generation Partnership Project (3GPP)

Status of this Memo

Copyright Notice

Abstract

This document describes a set of private Session Initiation Protocol (SIP) headers (P-headers) used by the 3rd-Generation Partnership Project (3GPP), along with their applicability, which is limited to particular environments. The P-headers are for a variety of purposes within the networks that the partners use, including charging and information about the networks a call traverses.

Table of Contents

1. Overall Applicability

The SIP extensions specified in this document make certain assumptions regarding network topology, linkage between SIP and lower layers, and the availability of transitive trust. These assumptions are generally NOT APPLICABLE in the Internet as a whole. The mechanisms specified here were designed to satisfy the requirements specified in the 3GPP Release 5 requirements on SIP [4] for which either no general-purpose solution was planned, where insufficient operational experience was available to understand if a general solution is needed, or where a more general solution is not yet mature. For more details about the assumptions made about these extensions, consult the Applicability subsection for each extension.

2. Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14, RFC 2119 [2].

3. Overview

The Third Generation Partnership Project (3GPP) has selected SIP as the protocol used to establish and tear down multimedia sessions in the context of its IP Multimedia Subsystem (IMS). (For more information on the IMS, a detailed description can be found in 3GPP TS 23.228 [14] and 3GPP TS 24.229 [15]). 3GPP notified the IETF SIP and SIPPING working groups that existing SIP documents provided almost all the functionality needed to satisfy the requirements of the IMS, but that they required some additional functionality in order to use SIP for this purpose. These requirements [4] are documented in an Internet Draft which was submitted to the SIPPING Working Group. Some of these requirements are satisfied by chartered extensions, while other requirements were applicable to SIP, but not sufficiently general for the SIP Working Group to adopt. This document describes private extensions to address those requirements. Each extension, or set of related extensions is described in its own section below.

4. SIP Private Headers

4.1 The P-Associated-URI header

This extension allows a registrar to return a set of associated URIs for a registered address-of-record. We define the P-Associated-URI header field, used in the 200 OK response to a REGISTER request. The P-Associated-URI header field transports the set of Associated URIs to the registered address-of-record.

An associated URI is a URI that the service provider has allocated to a user for his own usage. A registrar contains information that allows an address-of-record URI to be associated with zero or more URIs. Usually, all these URIs (the address-of-record URI and the associated URIs) are allocated for the usage of a particular user. This extension to SIP allows the UAC to know, upon a successful authenticated registration, which other URIs, if any, the service provider has associated to an address-of-record URI.

Note that, generally speaking, the registrar does not register the associated URIs on behalf of the user. Only the address-of-record which is present in the To header field of the REGISTER is registered and bound to the contact address. The only information conveyed is that the registrar is aware of other URIs to be used by the same user.

It may be possible, however, that an application server (or even the registrar itself) registers any of the associated URIs on behalf of the user by means of a third party registration. However, this third party registration is out of the scope of this document. A UAC MUST NOT assume that the associated URIs are registered.

If a UAC wants to check whether any of the associated URIs is registered, it can do so by mechanisms specified outside this document, e.g., the UA may send a REGISTER request with the To header field value set to any of the associated URIs and without a Contact header. The 200 OK response will include a Contact header with the list of registered contact addresses. If the associated URI is not registered, the UA MAY register it prior to its utilization.

4.1.1 Applicability statement for the P-Associated-URI header

The P-Associated-URI header is applicable in SIP networks where the SIP provider is allocating the set of identities that a user can claim (in headers like the From field) in requests that the UA generates. It furthermore assumes that the provider knows the entire set of identities that a user can legitimately claim, and that the user is willing to restrict its claimed identities to that set. This is in contrast to normal SIP usage, where the From field is explicitly an end-user specified field.

4.1.2 Usage of the P-Associated-URI header

The registrar inserts the P-Associated-URI header field into the 200 OK response to a REGISTER request. The header field value is populated with a list containing zero or more URIs that are associated to the address-of-record.

If the registrar supports the P-Associated-URI header extension, then the registrar MUST always insert the P-Associated-URI header field in all the 200 OK responses to a REGISTER request, regardless of whether the REGISTER was an initial registration, re-registration, or de-registration and regardless of whether there are zero or more associated URIs.

4.1.2.1 Procedures at the UA

A UAC may receive a P-Associated-URI header field in the 200 OK response for a REGISTER. The presence of the header field in the 200 OK response for a REGISTER request implies that the extension is supported at the registrar.

The header value contains a list of zero or more associated URIs to the address-of-record URI. The UAC MAY use any of the associated URIs to populate the From header value, or any other SIP header value that provides information of the identity of the calling party, in a subsequent request.

The UAC MAY check whether the associated URI is registered or not. This check can be done, e.g., by populating the To header value in a REGISTER sent to the registrar and without a Contact header. The 200 OK response will include a Contact header with the list of registered contact addresses. As described in SIP [1], the 200 OK response may contain a Contact header field with zero or more values (zero meaning the address-of-record is not registered).

4.1.2.2 Procedures at the registrar

A registrar that receives and authorizes a REGISTER request, may associate zero or more URIs with the address-of-record.

A registrar that supports this specification MUST include a P-Associated-URI header field in the 200 OK response to a REGISTER request. The header MUST be populated with a comma-separated list of SIP or SIPS URIs which are associated to the address-of-record under registration.

In case the address-of-record under registration does not have any other SIP or SIPS URIs associated, the registrar MUST include an empty P-Associated-URI header value.

4.1.2.3 Procedures at the proxy

This memo does not define any procedure at the proxy.

4.2 The P-Called-Party-ID header

A proxy server inserts a P-Called-Party-ID header, typically in an INVITE request, en-route to its destination. The header is populated with the Request-URI received by the proxy in the request. The UAS identifies which address-of-record, out of several registered address-of-records, the invitation was sent to (for example, the user may be simultaneously using a personal and a business SIP URIs to receive invitation to sessions). The UAS may use the information to render different distinctive audiovisual alerting tones, depending on the URI used to receive the invitation to the session.

Users in the 3GPP IP Multimedia Subsystem (IMS) may get one or several SIP URIs (address-of-record) to identify the user. For instance, a user may get a business SIP URI and a personal one. As an example of utilization, the user may make available the business SIP URI to co-workers and may make available the personal SIP URI to members of the family.

At a certain point in time, both the business SIP URI and the personal SIP URI are registered in the SIP registrar, so both URIs can receive invitations to new sessions. When the user receives an invitation to join a session, he/she should be aware of which of the several registered SIP URIs this session was sent to.

This requirement is stated in the 3GPP Release 5 requirements on SIP [4].

The problem arises during the terminating side of a session establishment, when the SIP proxy that is serving a UA gets an INVITE, and the SIP server retargets the SIP URI which is present in the Request-URI field, and replaces it by the SIP URI published by the user in the Contact header field of the REGISTER request at registration time. When the UAS receives the SIP INVITE, it cannot determine which address-of-record the request was sent to.

One can argue that the To header field conveys the semantics of the called user, and therefore, this extension to SIP is not needed. Although the To header field in SIP may convey the called party ID in most situations, there are two particular cases when the above assumption is not correct:

1. The session has been forwarded, redirected, etc., by previous SIP proxies, before arriving to the proxy which is serving the called user.

2. The UAC builds an INVITE request and the To header field is not the same as the Request-URI.

The problem of using the To header field is that this field is populated by the UAC and not modified by proxies in the path. If the UAC, for any reason, did not populate the To header field with the address-of-record of the destination user, then the destination user is not able to distinguish which address-of-record the session was destined.

Another possible solution to the problem is built upon the differentiation of the Contact header value between different address-of-record at registration time. The UA can differentiate each address-of-record it registers by assigning a different Contact header value. For instance, when the UA registers the address-of-record sip:id1, the Contact header value can be sip:id1@ua; the registration of sip:id2 can be bound to the Contact value sip:id2@ua.

The solution described above assumes that the UA explicitly registers each of its address-of-record URIs, and therefore, it has full control over the contact address values assigned to each registration. However, in the case the UA does not have full control of its registered address-of-record, because of, e.g., a third party registration, the solution does not work. This may be the case of the 3GPP registration, where the UA may have previously indicated the network, by means outside of SIP, that some other address-of-record URIs may be automatically registered when the UA registers a particular address-of-record. The requirement is covered in the 3GPP Release 5 requirements on SIP [4].

In the next paragraphs we show an example of the problem, in the case there has been some sort of call forwarding in the session, so that the UAC is not aware of the intended destination URI in the current INVITE.

We assume that a User Agent (UA) is registering to his proxy (P1).

```
      Scenario                      UA --- P1

   F1 Register UA -> P1
        REGISTER sip:example.com SIP/2.0
        Via: SIP/2.0/UDP 192.0.2.4:5060;branch=z9hG4bKnashds7
        To: sip:user1-business@example.com
        From: sip:user1-business@example.com;tag=456248
        Call-ID: 843817637684230998sdasdh09
        CSeq: 1826 REGISTER
        Contact: <sip:user1@192.0.2.4>
```

The user also registers his personal URI to his/her registrar.

F2 Register UA -> P1

```
REGISTER sip:example.com SIP/2.0
Via: SIP/2.0/UDP 192.0.2.4:5060;branch=z9hG4bKnashdt8
To: sip:user1-personal@example.com
From: sip:user1-personal@example.com;tag=346249
Call-ID: 2Q38176376842309 98sdasdh10
CSeq: 1827 REGISTER
Contact: <sip:user1@192.0.2.4>
```

Later, the proxy/registrar (P1) receives an INVITE from another proxy (P2) destined to the user's business SIP address-of-record. We assume that this SIP INVITE has undergone some sort of forwarding in the past, and as such, the To header field is not populated with the SIP URI of the user. In this case we assume that the session was initially addressed to sip:other-user@othernetwork.com. The SIP server at othernetwork.com has forwarded this session to sip:user1-business@example.com

Scenario UA --- P1 --- P2

F3 Invite P2 -> P1

```
INVITE sip:user1-business@example.com SIP/2.0
Via: SIP/2.0/UDP 192.0.2.20:5060;branch=z9hG4bK03djaoe1
To: sip:other-user@othernetwork.com
From: sip:another-user@anothernetwork.com;tag=938s0
Call-ID: 843817637684230998sdasdh09
CSeq: 101 INVITE
```

The proxy P1 retargets the user and replaces the Request-URI with the SIP URI published during registration time in the Contact header value.

F4 Invite P1 -> UA

```
INVITE sip:user1@192.0.2.4 SIP/2.0
Via: SIP/2.0/UDP 192.0.2.10:5060;branch=z9hG4bKg48sh128
Via: SIP/2.0/UDP 192.0.2.20:5060;branch=z9hG4bK03djaoe1
To: sip:other-user@othernetwork.com
From: sip:another-user@anothernetwork.com;tag=938s0
Call-ID: 843817637684230998sdasdh09
CSeq: 101 INVITE
```

When the UAS receives the INVITE, it cannot determine whether it got the session invitation due to his registration of the business or the personal address-of-record. Neither the UAS nor proxies or application servers can provide this user a service based on the destination address-of-record of the session.

We solve this problem by allowing the proxy that is responsible for the home domain (as defined in SIP) of the user to insert a P-Called-Party-ID header that identifies the address-of-record to which this session is destined.

If this SIP extension is used, the proxy serving the called user will get the message flow F5, it will populate the P-Called-Party-ID header in message flow F6 with the contents of the Request-URI in F4. This is show in flows F5 and F6 below:

```
F5 Invite P2 -> P1
     INVITE sip:user1-business@example.com SIP/2.0
     Via: SIP/2.0/UDP 192.0.2.20:5060;branch=z9hG4bK03djaoe1
     To: sip:other-user@othernetwork.com
     From: sip:another-user@anothernetwork.com;tag=938s0
     Call-ID: 843817637684230998sdasdh09
     CSeq: 101 INVITE

F6 Invite P1 -> UA
     INVITE sip:user1@192.0.2.4 SIP/2.0
     Via: SIP/2.0/UDP 192.0.2.10:5060;branch=z9hG4bKg48sh128
     Via: SIP/2.0/UDP 192.0.2.20:5060;branch=z9hG4bK03djaoe1
     To: sip:other-user@othernetwork.com
     From: sip:another-user@anothernetwork.com;tag=938s0
     Call-ID: 843817637684230998sdasdh09
     P-Called-Party-ID: sip:user1-business@example.com
     CSeq: 101 INVITE
```

When the UA receives the INVITE request F6 it can determine the intended address-of-record of the session, and apply whatever service is needed for that address-of-record.

4.2.1 Applicability statement for the P-Called-Party-ID header

The P-Called-Party-ID is applicable when the UAS needs to be aware of the intended address-of-record that was present in the Request-URI of the request, before the proxy retargets to the contact address. The UAS may be interested in applying different audiovisual alerting effects or other filtering services, depending on the intended destination of the request. It is specially valuable when the UAS has registered several address-of-record URIs to his registrar, and therefore, the UAS is not aware of the address-of-record that was present in the INVITE request when it hit his proxy/registrar, unless this extension is used.

Requirements for a more general solution are proposed in [12], but have not been adopted by SIP, nor a solution has been developed.

4.2.2 Usage of the P-Called-Party-ID header

The P-Called-Party-ID header field provides proxies and the UAS with the address-of-record that was present in the Request-URI of the request, before a proxy retargets the request. This information is intended to be used by subsequent proxies in the path or by the UAS.

Typically, a SIP proxy inserts the P-Called-Party-ID header prior to retargetting the Request-URI in the SIP request. The header value is populated with the contents of Request-URI, prior to replacing it with the Contact address.

4.2.2.1 Procedures at the UA

A UAC MUST NOT insert a P-Called-Party-ID header field in any SIP request or response.

A UAS may receive a SIP request that contains a P-Called-Party-ID header field. The header will be populated with the address-of-record received by the proxy in the Request-URI of the request, prior to its forwarding to the UAS.

The UAS may use the value in the P-Called-Party-ID header field to provide services based on the called party URI, such as, e.g., filtering of calls depending on the date and time, distinctive presentation services, distinctive alerting tones, etc.

4.2.2.2 Procedures at the proxy

A proxy that has access to the Contact information of the user, MAY insert a P-Called-Party-ID header field in any of the requests indicated in the Table 1 (Section 5.7). The proxy MUST populate the header value with the contents of the Request-URI present in the SIP request that the proxy received.

It is necessary that the proxy which inserts the P-Called-Party-ID header has information about the user, in order to prevent a wrong delivery of the called party ID. This information may have been learned through a registration process, for instance.

A proxy or application server that receives a request containing a P-Called-Party-ID header may use the contents of the header to provide a service to the user based on the URI of that header value.

A SIP proxy MUST NOT insert a P-Called-Party-ID header in REGISTER requests.

4.3 The P-Visited-Network-ID header

3GPP networks are composed of a collection of so called home networks, visited networks and subscribers. A particular home network may have roaming agreements with one or more visited networks. This has the effect that when a mobile terminal is roaming, it can use resources provided by the visited network in a transparent fashion.

One of the conditions for a home network to accept the registration of a UA roaming to a particular visited network, is the existence of a roaming agreement between the home and the visited network. There is a need to indicate to the home network which one is the visited network that is providing services to the roaming UA.

3GPP user agents always register to the home network. The REGISTER request is proxied by one or more proxies located in the visited network towards the home network. For the sake of a simple approach, it seems sensible that the visited network includes an identification that is known at the home network. This identification should be globally unique, and takes the form of a quoted text string or a token. The home network may use this identification to verify the existence of a roaming agreement with the visited network, and to authorize the registration through that visited network.

4.3.1 Applicability statement for the P-Visited-Network-ID header

The P-Visited-Network-ID is applicable whenever the following circumstances are met:

1. There is transitive trust in intermediate proxies between the UA and the home network proxy via established relationships between the home network and the visited network, and generally supported by the use of standard security mechanisms, e.g., IPsec, AKA, or TLS.

2. An endpoint is using resources provided by one or more visited networks (a network to which the user does not have a direct business relationship).

3. A proxy that is located in one of the visited networks wants to be identified at the user’s home network.

4. There is no requirement that every visited network needs to be identified at the home network. Those networks that want to be identified make use of the extension defined in this document. Those networks that do not want to be identified do nothing.

5. A commonly pre-agreed text string or token identifies the visited network at the home network.

6. The UAC sends a REGISTER or dialog-initiating request (e.g., INVITE) or a standalone request outside a dialog (e.g., OPTIONS) to a proxy in a visited network.

7. The request traverses, en route to its destination, a first proxy located in the visited network, and a second proxy located in the home network or its destination is the registrar in the home network.

8. The registrar or home proxy verifies and authorizes the usage of resources (e.g., proxies) in the visited network.

4.3.2 Usage of the P-Visited-Network-ID header

The P-Visited-Network-ID header field is used to convey to the registrar or home proxy in the home network the identifier of a visited network. The identifier is a text string or token that is known by both the registrar or the home proxy at the home network and the proxies in the visited network.

Typically, the home network authorizes the UA to roam to a particular visited network. This action requires an existing roaming agreement between the home and the visited network.

While it is possible for a home network to identify one or more visited networks by inspecting the domain name in the Via header fields, this approach has a heavy dependency on DNS. It is an option for a proxy to populate the via header with an IP address, for example, and in the absence of a reverse DNS entry, the IP address will not convey the desired information.

Any SIP proxy that receives any of the requests indicated in Table 1 (Section 5.7) MAY insert a P-Visited-Network-ID header when it forwards the request. In case a REGISTER or other request is traversing different administrative domains (e.g., different visited networks), a SIP proxy MAY insert a new P-Visited-Network-ID header if the request does not contain a P-Visited-Network-ID header with the same network identifier as its own network identifier (e.g., if the request has traversed other different administrative domains).

Note also that, there is not requirement for the header value to be readable in the proxies. Therefore, a first proxy may insert an encrypted header that only the registrar can decrypt. If the request traverses a second proxy located in the same administrative domain as the first proxy, the second proxy may not be able to read the

contents of the P-Visited-Network-ID header. In this situation, the second proxy will consider that its visited network identifier is not already present in the value of the header, and therefore, it will insert a new P-Visited-Network-ID header value (hopefully with the same identifier that the first proxy inserted, although perhaps, not encrypted). When the request arrives at the registrar or proxy in the home network, it will notice that the header value is repeated (both the first and the second proxy inserted it). The decrypted values should be the same, because both proxies where part of the same administrative domain. While this situation is not desirable, it does not create any harm at the registrar or proxy in the home network.

The P-Visited-Network-ID is normally used at registration. However, this extension does not preclude other usages. For instance, a proxy

located in a visited network that does not maintain registration state may insert a P-Visited-Network-ID header into any standalone request outside a dialog or a request that creates a dialog. At the time of writing this document, the only requests that create dialogs are INVITE [1], SUBSCRIBE [6] and REFER [11].

In order to avoid conflicts with identifiers, especially when the number of roaming agreements between networks increase, care must be taken when selecting the value of the P-Visited-Network-ID. The identifier should be a globally unique to avoid duplications. Although there are many mechanism to create globally unique identifiers across networks, one of such as mechanisms is already in operation, and that is DNS. The P-Visited-Network-ID does not have any connection to DNS, but the values in the header can be chosen from the own DNS entry representing the domain name of the network. This guarantees the uniqueness of the value.

4.3.2.1 Procedures at the UA

User agent clients SHOULD NOT insert a P-Visited-Network-ID header in any SIP message.

4.3.2.2 Procedures at the registrar and proxy

A SIP proxy which is located in a visited network MAY insert a P-Visited-Network-ID header field in any of the requests indicated in the Table 1 (Section 5.7). The header MUST be populated with the contents of a text string or a token that identifies the administrative domain of the network where the proxy is operating at the user’s home network.

A SIP proxy or registrar which is located in the home network may use the contents of the P-Visited-Network-ID as an identifier of one or more visited networks that the request traversed. The proxy or registrar in the home network may take local policy driven actions based on the existence or not of a roaming agreement between the home and the visited networks. This means, for instance, authorize the actions of the request based on the contents of the P-Visited-Network-ID header.

A SIP proxy which is located in the home network MUST delete this header when forwarding the message outside the home network administrative domain, in order to retain the user's privacy.

A SIP proxy which is located in the home network SHOULD delete this header when the home proxy has used the contents of the header or the request is routed based on the called party, even when the request is not forwarded outside the home network administrative domain.

4.3.2.3 Examples of Usage

We present example in the context of the scenario presented in the following network diagram:

```
Scenario            UA --- P1 --- P2 --- REGISTRAR
```

This example shows the message sequence for an REGISTER transaction originating from UA1 eventually arriving at REGISTRAR. P1 is an outbound proxy for UA1. In this case P1 also inserts the P-Visited-Network-ID header. P1 then routes the REGISTER request to the Registrar via P2.

Message sequence for REGISTER using P-Visited-Network-ID header:

```
F1 Register UA -> P1
     REGISTER sip:example.com SIP/2.0
     Via: SIP/2.0/UDP 192.0.2.4:5060;branch=z9hG4bKnashds7
     To: sip:user1-business@example.com
     From: sip:user1-business@example.com;tag=456248
     Call-ID: 843817637684230998sdasdh09
     CSeq: 1826 REGISTER
     Contact: <sip:user1@192.0.2.4>
```

In flow F2, proxy P2 adds its own identifier to the P-Visited-Network-ID header.

```
F2 Register P1 -> P2
     REGISTER sip:example.com SIP/2.0
     Via: SIP/2.0/UDP p1.visited.net;branch=z9hG4bK203igld
     Via: SIP/2.0/UDP 192.0.2.4:5060;branch=z9hG4bKnashdt8
     To: sip:user1-personal@example.com
     From: sip:user1-personal@example.com;tag=346249
     Call-ID: 2Q38176376842309 98sdasdh10
     CSeq: 1826 REGISTER
     Contact: <sip:user1@192.0.2.4>
     P-Visited-Network-ID: "Visited network number 1"
```

Finally, in flow F3, proxy P2 decides to insert his own identifier, derived from its own domain name.

```
F3 Register P2 -> REGISTRAR
     REGISTER sip:example.com SIP/2.0
     Via: SIP/2.0/UDP p2.other.net;branch=z9hG4bK2bndnvk
     Via: SIP/2.0/UDP p1.visited.net;branch=z9hG4bK203igld
     Via: SIP/2.0/UDP 192.0.2.4:5060;branch=z9hG4bKnashdt8
     To: sip:user1-personal@example.com
     From: sip:user1-personal@example.com;tag=346249
     Call-ID: 2Q38176376842309 98sdasdh10
     CSeq: 1826 REGISTER
     Contact: <sip:user1@192.0.2.4>
     P-Visited-Network-ID: other.net, "Visited network number 1"
```

4.4 The P-Access-Network-Info header

This section describes the P-Access-Network-Info header. This header is useful in SIP-based networks that also provide layer 2/layer 3 connectivity through different access technologies. SIP User Agents may use this header to relay information about the access technology to proxies that are providing services. The serving proxy may then use this information to optimize services for the UA. For example, a 3GPP UA may use this header to pass information about the access network such as radio access technology and radio cell identity to its home service provider.

For the purpose of this extension, we define an access network as the network providing the layer 2/layer 3 IP connectivity which in turn provides a user with access to the SIP capabilities and services provided.

In some cases, the SIP server that provides the user with services may wish to know information about the type of access network that the UA is currently using. Some services are more suitable or less

suitable depending on the access type, and some services are of more value to subscribers if the access network details are known by the SIP proxy which provides the user with services.

In other cases, the SIP server that provides the user with services may simply wish to know crude location information in order to provide certain services to the user. For example, many of the location based services available in wireless networks today require the home network to know the identity of the cell the user is being served by.

Some regulatory requirements exist mandating that for cellular radio systems, the identity of the cell where an emergency call is established is made available to the emergency authorities.

The SIP server that provides services to the user may desire knowledge about the access network. This is achieved by defining a new private SIP extension header, P-Access-Network-Info. This header carries information relating to the access network between the UAC and its serving proxy in the home network.

4.4.1 Applicability Statement for the P-Access-Network-Info header

This mechanism is appropriate in environments where SIP services are dependent on SIP elements knowing details about the IP and lower layer technologies used by a UA to connect to the SIP network. Specifically, the extension requires that the UA know the access technology it is using, and that a proxy desires such information to provide services. Generally, SIP is built on the "Everything over IP and IP over everything" principle, where the access technology is not relevant for the operation of SIP. Since SIP systems generally should not care or even know about the access technology, this SIP extension is not for general SIP usage.

The information revealed in the P-Access-Network-Info header is potentially very sensitive. Proper protection of this information depends on the existence of specific business and security relationships amongst the proxies that will see SIP messages containing this header. It also depends on explicit knowledge of the UA of the existence of those relationships. Therefore, this mechanism is only suitable in environments where the appropriate relationships are in place, and the UA has explicit knowledge that they exist.

4.4.2 Usage of the P-Access-Network-Info header

When a UA generates a SIP request or response which it knows is going to be securely sent to its SIP proxy that is providing services, the UA inserts a P-Access-Network-Info header into the SIP message. This header contains information on the access network that the UA is using to get IP connectivity. The header is typically ignored by intermediate proxies between the UA and the SIP proxy that is providing services. The proxy providing services can inspect the header and make use of the information contained there to provide appropriate services, depending on the value of the header. Before proxying the request onwards, this proxy strips the header from the message.

4.4.2.1 UA behavior

A UA that supports this extension and is willing to disclose the related parameters MAY insert the P-Access-Network-Info header in any SIP request or response.

The UA inserting this information MUST trust the proxy that is providing services to protect its privacy by deleting the header before forwarding the message outside of the proxy's domain. This proxy is typically located in the home network.

In order to do the deletion of the header, there must also be a transitive trust in intermediate proxies between the UA and the proxy that provides the services. This trust is established by business agreements between the home network and the access network, and generally supported by the use of standard security mechanisms, e.g., IPsec, AKA, and TLS.

4.4.2.2 Proxy behavior

A proxy MUST NOT insert or modify the value of the P-Access-Network-Info header.

A proxy which is providing services to the UA, may act upon any information present in the P-Access-Network-Info header value, if is present, to provide a different service depending on the network or the location through which the UA is accessing the server. For example, for cellular radio access networks the SIP proxy located in the home network may use the cell ID to provide basic localized services.

A proxy that provides services to the user, the proxy typically located in the home network, and therefore trusted, MUST delete the header when the SIP signaling is forwarded to a SIP server located in

a non-trusted administrative network domain. The SIP server providing services to the UA uses the access network information and is of no interest to other proxies located in different administrative domains.

4.5 The P-Charging-Function-Addresses header

3GPP has defined a distributed architecture that results in multiple network entities becoming involved in providing access and services. There is a need to inform each SIP proxy involved in a transaction about the common charging functional entities to receive the generated charging records or charging events.

The solution provided by 3GPP is to define two types of charging functional entities: Charging Collection Function (CCF) and Event Charging Function (ECF). CCF is used for off-line charging (e.g., for postpaid account charging). ECF is used for on-line charging (e.g., for pre-paid account charging). There may be more than a single instance of CCF and ECF in a network, in order to provide redundancy in the network. In case there are more than a single instance of either the CCF or the ECF addresses, implementations SHOULD attempt sending the charging data to the ECF or CCF address, starting with the first address of the sequence (if any) in the P-Charging-Function-Addresses header. The CCF and ECF addresses may be passed during the establishment of a dialog or in a standalone transaction. More detailed information about charging can be found in 3GPP TS 32.200 [16] and 3GPP TS 32.225 [17].

We define the SIP private header P-Charging-Function-Addresses. A proxy MAY include this header, if not already present, in either the initial request or response for a dialog, or in the request and response of a standalone transaction outside a dialog. Only one instance of the header MUST be present in a particular request or response.

The mechanisms by which a SIP proxy collects the values to populate the P-Charging-Function-Addresses header values are outside the scope of this document. However, as an example, a SIP proxy may have preconfigured these addresses, or may obtain them from a subscriber database.

4.5.1 Applicability Statement for the P-Charging-Function-Addresses header

The P-Charging-Function-Addresses header is applicable within a single private administrative domain where coordination of charging is required, for example, according to the architecture specified in 3GPP TS 32.200 [16].

The P-Charging-Function-Addresses header is not included in a SIP message sent outside of the own administrative domain. The header is not applicable if the administrative domain does not provide a charging function.

The P-Charging-Function-Addresses header is applicable whenever the following circumstances are met:

1. A UA sends a REGISTER or dialog-initiating request (e.g., INVITE) or a standalone transaction request outside a dialog to a proxy located in the administrative domain of a private network.

2. A registrar, proxy or UA that is located in the administrative domain of the private network wants to generate charging records.

3. A registrar, proxy or UA that is located in the private network has access to the addresses of the charging function entities for that network.

4. There are other proxies located in the same administrative domain of the private network, that are generated charging records or charging events. The proxies want to send, by means outside SIP, the charging information to the same charging collecting entities than the first proxy.

4.5.2 Usage of the P-Charging-Function-Addresses header

A SIP proxy that receives a SIP request may insert a P-Charging-Function-Addresses header prior to forwarding the request, if the header was not already present in the SIP request. The header value contains one or more parameters that contain the hostnames or IP addresses of the nodes that are willing to receive charging information.

A SIP proxy that receives a SIP request that includes a P-Charging-Function-Addresses may use the hostnames or IP addresses included in the value, as the destination of charging information or charging events. The means to send those charging information or events are outside the scope of this document, and usually, do not use SIP for that purpose.

4.5.2.1 Procedures at the UA

This document does not specify any procedure at the UA, with regard to the P-Charging-Function-Addresses header. UAs need not understand this header.

However, it might be possible that a UA is located within the administrative domain of a private network (e.g., a PSTN gateway, or conference mixer), and it may have access to the addresses of the charging entities. In this cases, a UA MAY insert the P-Charging-Function-Addresses header in a SIP request or response when the next hop for the message is a proxy located in the same administrative domain.

4.5.2.2 Procedures at the Proxy

A SIP proxy that supports this extension and receives a request or response without the P-Charging-Function-Addresses MAY insert a P-Charging-Function-Addresses header prior to forwarding the message. The header is populated with a list of the addresses of one or more charging entities where the proxy should send charging related information.

If a proxy that supports this extension receives a request or response with the P-Charging-Function-Addresses, it may retrieve the information from the header value to use with application specific logic, i.e., charging. If the next hop for the message is within the administrative domain of the proxy, then the proxy SHOULD include the P-Charging-Function-Addresses header in the outbound message. However, if the next hop for the message is outside the administrative domain of the proxy, then the proxy MUST remove the P-Charging-Function-Addresses header.

4.5.2.3 Examples of Usage

We present example in the context of the scenario presented in the following network diagram:

```
   Scenario                    UA1 --- P1 --- P2 --- UA2
```

In the scenario we assume that P1 and P2 belong to the same administrative domain.

The example below shows the message sequence for an INVITE transaction originating from UA1 eventually arriving at UA2. P1 is an outbound proxy for UA1. In this case P1 also inserts charging information. P1 then routes the call via P2 to UA2.

Message sequence for INVITE using P-Charging-Function-Addresses:

```
F1 Invite UA1 -> P1
   INVITE sip:ua2@home1.net SIP/2.0
   Via: SIP/2.0/UDP 192.0.2.4:5060;branch=z9hG4bKnashds7
   To: sip:ua2@home1.net
   From: sip:ua1@home1.net;tag=456248
   Call-ID: 843817637684230998sdasdh09
   CSeq: 18 INVITE
   Contact: sip:ua1@192.0.2.4

F2 Invite P1 -> P2
   INVITE sip:ua2@home1.net SIP/2.0
   Via: SIP/2.0/UDP p1.home1.net:5060;branch=z9hG4bK34ghi7ab04
   Via: SIP/2.0/UDP 192.0.2.4:5060;branch=z9hG4bKnashds7
   To: sip:ua2@home1.net
   From: sip:ua1home1.net;tag=456248
   Call-ID: 843817637684230998sdasdh09
   CSeq: 18 INVITE
   Contact: sip:ua1@192.0.2.4
   P-Charging-Function-Addresses: ccf=192.1.1.1; ccf=192.1.1.2;
                                  ecf=192.1.1.3; ecf=192.1.1.4
```

Now both P1 and P2 are aware of the IP addresses of the entities that collect charging record or charging events. Both proxies can send the charging information to the same entities.

4.6 The P-Charging-Vector header

3GPP has defined a distributed architecture that results in multiple network entities becoming involved in providing access and services. Operators need the ability and flexibility to charge for the access and services as they see fit. This requires coordination among the network entities (e.g., SIP proxies), which includes correlating charging records generated from different entities that are related to the same session.

The correlation information includes, but it is not limited to, a globally unique charging identifier that makes easy the billing effort.

A charging vector is defined as a collection of charging information. The charging vector may be filled in during the establishment of a dialog or standalone transaction outside a dialog. The information inside the charging vector may be filled in by multiple network entities (including SIP proxies) and retrieved by multiple network entities. There are three types of correlation information to be transferred: the IMS Charging Identity (ICID) value, the address of the SIP proxy that creates the ICID value, and the Inter Operator Identifiers (IOI).

ICID is a charging value that identifies a dialog or a transaction outside a dialog. It is used to correlate charging records. ICID MUST be a globally unique value. One way to achieve globally uniqueness is to generate the ICID using two components: a locally unique value and the host name or IP address of the SIP proxy that generated the locally unique value.

The IOI identifies both the originating and terminating networks involved in a SIP dialog or transaction outside a dialog. There may an IOI generated from each side of the dialog to identify the network associated with each side.

There is also expected to be access network charging information, which consists of network specific identifiers for the access level (e.g., UMTS radio access network or IEEE 802.11b). The details of the information for each type of network are not described in this memo.

We define the SIP private header P-Charging-Vector. A proxy MAY include this header, if not already present, in either the initial request or response for a dialog, or in the request and response of a standalone transaction outside a dialog. Only one instance of the header MUST be present in a particular request or response.

The mechanisms by which a SIP proxy collects the values to populate in the P-Charging-Vector are outside the scope of this document.

4.6.1 Applicability Statement for the P-Charging-Vector header

The P-Charging-Vector header is applicable within a single private administrative domain or between different administrative domains where there is a trust relationship between the domains.

The P-Charging-Vector header is not included in a SIP message sent to another network if there is no trust relationship. The header is not applicable if the administrative domain manages charging in a way that does not require correlation of records from multiple network entities (e.g., SIP proxies).

The P-Charging-Vector header is applicable whenever the following circumstances are met:

1. A UA sends a REGISTER or dialog-initiating request (e.g., INVITE) or a standalone transaction request outside a dialog to a proxy located in the administrative domain of a private network.

2. A registrar, proxy or UA that is located in the administrative domain of the private network wants to generate charging records.

3. A proxy or UA that is located in the administrative domain of the private network has access to the charging correlation information for that network.

4. Optionally, a registrar, proxy or UA that is part of a second administrative domain in another private network, whose SIP request and responses are traversed through, en-route to the first private network, wants to generate charging records and correlate those records with those of the first private network. This assumes that there is a trust relationship between both private networks.

4.6.2 Usage of the P-Charging-Vector header

The P-Charging-Vector header is used to convey charging related information, such as the globally unique IMS charging identifier (ICID) value.

Typically, a SIP proxy that receives a SIP request that does not contain a P-Charging-Vector header may insert it, with those parameters that are available at the SIP proxy.

A SIP proxy that receives a SIP request that contains a P-Charging-Vector header may use the values, such as the globally unique ICID, to produce charging records.

4.6.2.1 Procedures at the UA

This document does not specify any procedure at the UA, with regard to the P-Charging-Vector header. UAs need not understand this header.

4.6.2.2 Procedures at the Proxy

A SIP proxy that supports this extension and receives a request or response without the P-Charging-Vector header MAY insert a P-Charging-Vector header prior to forwarding the message. The header is populated with one ore more parameters, as described in the syntax, including but not limited to, a globally unique charging identifier.

If a proxy that supports this extension receives a request or response with the P-Charging-Vector header, it may retrieve the information from the header value to use with application specific logic, i.e., charging. If the next hop for the message is within the trusted domain, then the proxy SHOULD include the P-Charging-Vector

header in the outbound message. If the next hop for the message is outside the trusted domain, then the proxy MAY remove the P-Charging-Function-Addresses header.

Per local application specific logic, the proxy MAY modify the contents of the P-Charging-Vector header prior to sending the message.

4.6.2.3 Examples of Usage

We present example in the context of the scenario presented in the following network diagram:

Scenario UA1 --- P1 --- P2 --- UA2

This example shows the message sequence for an INVITE transaction originating from UA1 eventually arriving at UA2. P1 is an outbound proxy for UA1. In this case P1 also inserts charging information. P1 then routes the call via P2 to UA2.

Message sequence for INVITE using P-Charging-Vector:

F1 Invite UA1 -> P1

```
INVITE sip:joe@example.com SIP/2.0
Via: SIP/2.0/UDP 192.0.2.4:5060;branch=z9hG4bKnashds7
To: sip:joe@example.com
From: sip:ua1@home1.net;tag=456248
Call-ID: 843817637684230998sdasdh09
CSeq: 18 INVITE
Contact: sip:ua1@192.0
```

F2 Invite P1 -> P2

```
INVITE sip:joe@example.com SIP/2.0
Via: SIP/2.0/UDP P1.home1.net:5060;branch=z9hG4bK34ghi7a
Via: SIP/2.0/UDP 192.0.2.4:5060;branch=z9hG4bKnashds7
To: sip:joe@example.com
From: sip:ua1@home1.net;tag=456248
Call-ID: 843817637684230998sdasdh09
CSeq: 18 INVITE
Contact: sip:ua1@192.0.2.4
P-Charging-Vector: icid-value=1234bc9876e;
                   icid-generated-at=192.0.6.8;
                    orig-ioi=home1.net
```

5. Formal Syntax

All of the mechanisms specified in this document are described in both prose and an augmented Backus-Naur Form (BNF) defined in RFC 2234 [3]. Further, several BNF definitions are inherited from SIP and are not repeated here. Implementors need to be familiar with the notation and contents of SIP [1] and RFC 2234 [3] to understand this document.

5.1 P-Associated-URI header syntax

The syntax of the P-Associated-URI header is described as follows:

```
P-Associated-URI       = "P-Associated-URI" HCOLON
                         (p-aso-uri-spec)
                         *(COMMA p-aso-uri-spec)
p-aso-uri-spec         = name-addr *(SEMI ai-param)
ai-param               = generic-param
```

5.2 P-Called-Party-ID header syntax

The syntax of the P-Called-Party-ID header is described as follows:

```
P-Called-Party-ID      = "P-Called-Party-ID" HCOLON
                         called-pty-id-spec
called-pty-id-spec     = name-addr *(SEMI cpid-param)
cpid-param             = generic-param
```

5.3 P-Visited-Network-ID header syntax

The syntax of the P-Visited-Network-ID header is described as follows:

```
P-Visited-Network-ID   = "P-Visited-Network-ID" HCOLON
                          vnetwork-spec
                          *(COMMA vnetwork-spec)
vnetwork-spec          = (token / quoted-string)
                          *(SEMI vnetwork-param)
vnetwork-param         = generic-param
```

5.4 P-Access-Network-Info header syntax

The syntax of the P-Access-Network-Info header is described as follows:

```
P-Access-Network-Info  = "P-Access-Network-Info" HCOLON
                         access-net-spec
access-net-spec        = access-type *(SEMI access-info)
```

```
access-type            = "IEEE-802.11a" / "IEEE-802.11b" /
                         "3GPP-GERAN" / "3GPP-UTRAN-FDD" /
                         "3GPP-UTRAN-TDD" /
                         "3GPP-CDMA2000" / token
access-info            = cgi-3gpp / utran-cell-id-3gpp /
                         extension-access-info
extension-access-info  = gen-value
cgi-3gpp               = "cgi-3gpp" EQUAL
                         (token / quoted-string)
utran-cell-id-3gpp     = "utran-cell-id-3gpp" EQUAL
                         (token / quoted-string)
```

The access-info may contain additional information relating to the access network. The values for "cgi-3gpp" and "utran-cell-id-3gpp" are defined in 3GPP TS 24.229 [15].

5.5 P-Charging-Function-Addresses header syntax

The syntax for the P-Charging-Function-Addresses header is described as follows:

```
P-Charging-Addr        = "P-Charging-Function-Addresses" HCOLON
                         charge-addr-params
                         *(SEMI charge-addr-params)
charge-addr-params     = ccf / ecf / generic-param
ccf                    = "ccf" EQUAL gen-value
ecf                    = "ecf" EQUAL gen-value
```

5.6 P-Charging-Vector header syntax

The syntax for the P-Charging-Vector header is described as follows:

```
P-Charging-Vector      = "P-Charging-Vector" HCOLON icid-value
                         *(SEMI charge-params)
charge-params          = icid-gen-addr / orig-ioi /
                         term-ioi / generic-param
icid-value             = "icid-value" EQUAL gen-value
icid-gen-addr          = "icid-generated-at" EQUAL host
orig-ioi               = "orig-ioi" EQUAL gen-value
term-ioi               = "term-ioi" EQUAL gen-value
```

The P-Charging-Vector contains icid-value mandatory parameter. The icid-value represents the IMS charging ID, and contains an identifier used for correlating charging records and events. The first proxy that receives the request generates this value.

The icid-gen-addr parameter contains the host name or IP address of the proxy that generated the icid-value.

The orig-ioi and term-ioi parameters represent, respectively, the originating and terminating interoperator identifiers. They are used to correlate charging records between different operators. The originating ioi represents the network responsible for the charging records in the originating part of the session or standalone request. Similarly, the terminating ioi represents the network responsible for the charging records in the terminating part of the session or standalone request.

5.7 Table of new headers

Table 1 extends the headers defined in this document to Table 2 in SIP [1], section 7.1 of the SIP-specific event notification [6], tables 1 and 2 in the SIP INFO method [8], tables 1 and 2 in Reliability of provisional responses in SIP [7], tables 1 and 2 in the SIP UPDATE method [9], tables 1 and 2 in the SIP extension for Instant Messaging [10], and table 1 in the SIP REFER method [11]:

Header field	where	proxy	ACK	BYE	CAN	INV	OPT	REG
P-Associated-URI	2xx		-	-	-	-	-	o
P-Called-Party-ID	R	amr	-	-	-	o	o	-
P-Visited-Network-ID	R	ad	-	-	-	o	o	o
P-Access-Network-Info		dr	-	o	-	o	o	o
P-Charging-Vector		admr	-	o	-	o	o	o
P-Charging-Function-Addresses		adr	-	o	-	o	o	o

Header field	SUB	NOT	PRA	INF	UPD	MSG	REF
P-Associated-URI	-	-	-	-	-	-	-
P-Called-Party-ID	o	-	-	-	-	o	o
P-Visited-Network-ID	o	-	-	-	-	o	o
P-Access-Network-Info	o	o	o	o	o	o	o
P-Charging-Vector	o	o	o	o	o	o	o
P-Charging-Function-Addresses	o	o	o	o	o	o	o

Table 1: Header field support

6. Security Considerations

6.1 P-Associated-URI

The information returned in the P-Associated-URI header is not viewed as particularly sensitive. Rather, it is simply informational in nature, providing openness to the UAC with regard to the automatic association performed by the registrar. If end-to-end protection is not used at the SIP layer, it is possible for proxies between the registrar and the UA to modify the contents of the header value. This attack, while potentially annoying, should not have significant impacts.

The lack of encryption, either end-to-end or hop-by-hop, may lead to leak some privacy regarding the list of authorized identities. For instance, a user who registers an address-of-record of sip:user1@example.com may get another SIP URI associated as sip:first.last@example.com returned in the P-Associated-URI header value. An eavesdropper could collect this information. If the user does not want to disclose the associated URIs, the eavesdropper could have gain access to private URIs. Therefore it is RECOMMENDED that this extension is used in a secured environment, where encryption of SIP messages is provided either end-to-end or hop-by-hop.

6.2 P-Called-Party-ID

Due to the nature of the P-Called-Party-ID header, this header does not introduce any significant security concern. It is possible for an attacker to modify the contents of the header. However, this modification will not cause any harm to the session establishment.

An eavesdropper may collect the list of identities a user is registered. This may have privacy implications. To mitigate this problem, this extension SHOULD only be used in a secured environment, where encryption of SIP messages is provided either end-to-end or hop-by-hop.

6.3 P-Visited-Network-ID

The P-Visited-Network-ID header assumes that there is trust relationship between a home network and one or more transited visited networks. It is possible for other proxies between the proxy in the visited network that inserts the header, and the registrar or the home proxy, to modify the value of P-Visited-Network-ID header. Therefore intermediaries participating in this mechanism MUST apply a hop-by-hop integrity protection mechanism such us IPsec or other available mechanisms in order to prevent such attacks.

6.4 P-Access-Network-Info

A Trust Domain is formally defined in the Short term requirements for Network Asserted Identity [13] document. For the purpose of this document, we refer to the 3GPP trust domain as the collection of SIP proxies and application servers that are operated by a 3GPP network operator and are compliant with the requirements expressed in 3GPP TS 24.229 [15].

This extension assumes that the access network is trusted by the UA (because the UA's home network has a trust relationship with the access network), as described earlier in this document.

This extension assumes that the information added to the header by the UAC should be sent only to trusted entities and should not be used outside of the trusted administrative network domain.

The SIP proxy that provides services to the user, utilizes the information contained in this header to provide additional services and UAs are expected to provide correct information. However, there are no security problems resulting from a UA inserting incorrect information. Networks providing services based on the information carried in the P-Access-Network-Info header will therefore need to trust the UA sending the information. A rogue UA sending false access network information will do no more harm than to restrict the user from using certain services.

The mechanism provided in this document is designed primarily for private systems like 3GPP. Most security requirements are met by way of private standardized solutions.

For instance, 3GPP will use the P-Access-Network-Info header to carry relatively sensitive information like the cell ID. Therefore the information MUST NOT be sent outside of the 3GPP domain.

The UA is aware - if it is a 3GPP UA - that it is operating within a trusted domain.

The 3GPP UA is aware of whether or not a secure association to the home network domain for transporting SIP signaling, is currently available, and as such the sensitive information carried in the P-Access-Network-Info header SHOULD NOT be sent in any initial unauthenticated and unprotected requests (e.g., REGISTER).

Any UA that is using this extension and is not part of a private trusted domain should not consider the mechanism as secure and as such SHOULD NOT send sensitive information in the P-Access-Network-Info header.

Any proxy that is operating in a private trust domain where the P-Access-Network-Info header is supported is required to delete the header, if it is present, from any message prior to forwarding it outside of the trusted domain.

Therefore, a network that requires its UA to send information in the P-Access-Network-Info header must ensure that either that information is not of a sensitive nature or that the information is not sent outside of the trust domain.

A proxy receiving a message containing the P-Access-Network-Info header from a non-trusted entity is not able to guarantee the validity of the contents.

6.5 P-Charging-Function-Addresses

It is expected as normal behavior that proxies within a closed network will modify the values of the P-Charging-Function-Addresses and insert it into a SIP request or response. However, these proxies that share this information MUST have a trust relationship.

If an untrusted entity were inserted between trusted entities, it could potentially substitute a different charging function address. Therefore, an integrity protection mechanism such as IPsec or other available mechanisms MUST be applied in order to prevent such attacks. Since each trusted proxy may need to view or modify the values in the P-Charging-Function-Addresses header, the protection should be applied on a hop-by-hop basis.

6.6 P-Charging-Vector

It is expected as normal behavior that proxies within a closed network will modify the values of the P-Charging-Vector and insert it into a SIP request or response. However, these proxies that share this information MUST have a trust relationship.

If an untrusted entity were inserted between trusted entities, it could potentially interfere with the charging correlation mechanism. Therefore, an integrity protection mechanism such as IPsec or other available mechanisms MUST be applied in order to prevent such attacks. Since each trusted proxy may need to view or modify the values in the P-Charging-Vector header, the protection should be applied on a hop-by-hop basis.

7. IANA Considerations

This document defines several private SIP extension header fields (beginning with the prefix "P-").

These extension headers have been included in the registry of SIP header fields defined in SIP [1]. Expert review as required for this process was provided by the SIP Working Group.

The following extensions are registered as private extension header fields:

RFC Number: RFC3455
Header Field Name: P-Associated-URI
Compact Form: none

RFC Number: RFC3455
Header Field Name: P-Called-Party-ID
Compact Form: none

RFC Number: RFC3455
Header Field Name: P-Visited-Network-ID
Compact Form: none

RFC Number: RFC3455
Header Field Name: P-Access-Network-Info
Compact Form: none

RFC Number: RFC3455
Header Field Name: P-Charging-Function-Addresses
Compact Form: none

RFC Number: RFC3455
Header Field Name: P-Charging-Vector
Compact Form: none

8. Contributors

The extensions described in this document were originally specified in several documents. Miguel Garcia-Martin authored the P-Associated-URI, P-Called-Party-ID, and P-Visited-Network-ID headers. Duncan Mills authored the P-Access-Network-Info header. Eric Henrikson authored the P-Charging-Function-Addresses and P-Charging-Vector headers. Rohan Mahy assisted in the incorporation of these extensions into a single document.

9. Acknowledgments

The authors would like to thank Andrew Allen, Gabor Bajko, Gonzalo Camarillo, Keith Drage, Georg Mayer, Dean Willis, Rohan Mahy, Jonathan Rosenberg, Ya-Ching Tan and the 3GPP CN1 WG members for their comments on this document.

10. Normative References

[1] 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.

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

[3] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", RFC 2234, November 1997.

11. Informative References

[4] Garcia-Martin, M., "3rd-Generation Partnership Project (3GPP) Release 5 requirements on the Session Initiation Protocol (SIP)", Work in Progress.

[5] Mankin, A., Bradner, S., Mahy, R., Willis, D., Ott, J. and B. Rosen, "Change Process for the Session Initiation Protocol (SIP)", BCP 67, RFC 3427, December 2002.

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

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

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

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

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

[11] Sparks, R., "The SIP Refer Method", Work in Progress.

[12] Barnes, M., "SIP Generic Request History Capability Requirements", Work in Progress.

[13] Watson, M., "Short Term Requirements for Network Asserted Identity", RFC 3324, November 2002.

[14] 3GPP, "TS 23.228: IP Multimedia Subsystem (IMS); Stage 2 (Release 5)", 3GPP 23.228, September 2002, <ftp://ftp.3gpp.org/Specs/archive/23_series/23.228/>.

[15] 3GPP, "TS 24.229: IP Multimedia Call Control Protocol based on SIP and SDP; Stage 3 (Release 5)", 3GPP 24.229, September 2002, <ftp://ftp.3gpp.org/Specs/archive/24_series/24.229/>.

[16] 3GPP, "TS 32.200: Telecommunication Management; Charging management; Charging principles (Release 5)", 3GPP 32.200, June 2002, <ftp://ftp.3gpp.org/Specs/archive/32_series/32.200/>.

[17] 3GPP, "TS 32.225: Telecommunication Management; Charging management; Charging Data Description for IP Multimedia Subsystem (Release 5)", 3GPP 32.225, September 2002, <ftp://ftp.3gpp.org/Specs/archive/32_series/32.225/>.

Authors' Addresses

Miguel A. Garcia-Martin
Ericsson
Hirsalantie 11
Jorvas FIN-02420
Finland
EMail: miguel.a.garcia@ericsson.com

Eric Henrikson
Lucent
11601 Willows Rd, Suite 100
Redmond, WA 98052
USA
EMail: ehenrikson@lucent.com

Duncan Mills
Vodafone
The Courtyard, 2-4 London Road
Newbury, Berkshire RG14 1JX
UK
EMail: duncan.mills@vf.vodafone.co.uk

Full Copyright Statement

Acknowledgement

Funding for the RFC Editor function is currently provided by the Internet Society.