SIP Working Group
Internet Draft
Document: <draft-ietf-sip-manyfolks-resource-02>

W. Marshall
AT&T

K. Ramakrishnan
TeraOptic Networks

E. Miller
Terayon

G. Russell
CableLabs

B. Beser
Pacific Broadband

M. Mannette
K. Steinbrenner
3Com

D. Oran
F. Andreasen
M. Ramalho
Cisco

J. Pickens
Com21

P. Lalwaney
Nokia

J. Fellows
Copper Mountain Networks

D. Evans
D. R. Evans Consulting

K. Kelly
NetSpeak

A. Roach
Ericsson

J. Rosenberg
D. Willis
S. Donovan
dynamicsoft

H. Schulzrinne
Columbia University

February, 2001

Integration of Resource Management and SIP

Status of this Memo

This document is an Internet-Draft and is in full conformance with all provisions of Section 10 of RFC2026[1].

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-manyfolks-resource-02.txt>, and expires February 28, 2002. Please send comments to the authors.

1. Abstract

This document discusses how network QoS and security establishment can be made a precondition to sessions initiated by the Session Initiation Protocol (SIP), and described by SDP. These preconditions require that the participant reserve network resources (or establish a secure media channel) before continuing with the session. We do not define new QoS reservation or security mechanisms; these pre-conditions simply require a participant to use existing resource reservation and security mechanisms before beginning the session.

This results in a multi-phase call-setup mechanism, with the resource management protocol interleaved between two phases of call signaling. The objective of such a mechanism is to enable deployment of robust IP Telephony services, by ensuring that resources are made available before the phone rings and the participants of the call are "invited" to participate.

This document also proposes an extension to the Session Initiation Protocol (SIP) to add a new COMET method, which is used to confirm the completion of all pre-conditions by the session originator.

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[4].

3. Table of Contents

4. Introduction

For an Internet Telephony service to be successfully used by a large number of subscribers, it must offer few surprises to those accustomed to the behavior of existing telephony services. One expectation is that of connection quality, implying resources must be set aside for each call.

A key contribution is a recognition of the need for coordination between call signaling, which controls access to telephony specific services, and resource management, which controls access to network-layer resources. This coordination is designed to meet the user expectations and human factors associated with telephony.

While customers may expect, during times of heavy load, to receive a "fast busy" or an announcement saying "all circuits are busy now," the general expectation is that once the destination phone rings that the connection can be made. Blocking a call after ringing the destination is considered a "call defect" and is a very undesirable exception condition.

This draft addresses both "QoS-Assured" and "QoS-Enabled" sessions. A "QoS-Assured" session will complete only if all the required resources are available and assigned to the session. A provider may choose to block a call when adequate resources for the call are not available. Public policy demands that the phone system provide adequate quality at least in certain cases: e.g., for emergency communications during times of disasters. Call blocking enables a provider to meet such requirements.

A "QoS-Enabled" session allows the endpoints to complete the session establishment either with or without the desired resources. Such session will use dedicated resources if available, and use a best-effort connection as an alternative if resources can not be dedicated. In cases where resources are not available, the originating and/or terminating User Agent might consult with the customer to obtain guidance on whether the session should complete.

Coordination between call signaling and resource management is also needed to prevent fraud and theft of service. The coordination between call-signaling and QoS setup protocols ensures that users are authenticated and authorized before receiving access to the enhanced QoS associated with the telephony service.

This coordination, referred to in this draft as "preconditions," require that the participant reserve network resources (or establish a secure media channel) before continuing with the session. We do not define new QoS reservation or security mechanisms; these pre-conditions simply require a participant to use existing resource reservation and security mechanisms before beginning the session.

In the case of SIP [2], this effectively means that the "phone won't

ring" until the preconditions are met. These preconditions are described by new SDP parameters, defined in this document. The parameters can mandate end-to-end QoS reservations based on RSVP [5] or any other end-to-end reservation mechanism (such as YESSIR [6],

or PacketCable's Dynamic Quality of Service (D-QoS) [7]), and security based on IPSEC [8]. The preconditions can be defined independently for each media stream.

The QoS architecture of the Internet separates QoS signaling from application level signaling. Application layer devices (such as web proxies and SIP servers) are not well suited for participation in network admission control or QoS management, as this is fundamentally a network layer issue, independent of any particular application. In addition, since application devices like SIP servers are almost never on the "bearer path" (i.e., the network path the RTP [9] takes), and since the RTP path and signaling paths can be completely different (even traversing different autonomous systems), these application servers are generally not capable of managing QoS for the media. Keeping QoS out of application signaling also means that there can be a single infrastructure for QoS across all applications. This eliminates duplication of functionality, reducing management and equipment costs. It also means that new applications, with their own unique QoS requirements, can be easily supported.

This loose coupling works very well for a wide range of applications. For example, in an interactive game, one can establish the game using an application signaling protocol, and then later on use RSVP to reserve network resources. The separation is also effective for applications which have no explicit signaling. However, certain applications may require tighter coupling. In the case of Internet telephony, the following is an important requirement:

> When A calls B, B's phone should not ring unless resources have been reserved from A to B, and B to A.

This could be achieved without coupling if A knew B's address, port, and codecs before the telephony signaling took place. However, since telephony signaling is used largely to obtain this information in the first place, the coupling cannot be avoided.

A similar model exists for security. Rather than inventing new security mechanisms for each new application, common security tools (such as IPSEC) can be used across all applications. As with QoS, a means in application level protocols is needed to indicate that a security association is needed for the application to execute.

To solve both of these problems, we propose an extension to SDP which allows indication of pre-conditions for sessions. These preconditions indicate that participation in the session should not proceed until the preconditions are met. The preconditions we define are (1) success of end-to-end resource reservation, and (2) success

of end- to-end security establishment. We chose to implement these extensions in SDP, rather than SIP [2] or SAP [10], since they are fundamentally a media session issue. SIP is session agnostic; information about codecs, ports, and RTP [9] are outside the scope of SIP. Since it is the media sessions that the reservations and security refer to, SDP is the appropriate venue for the extensions.

Furthermore, placement of the extensions in SDP rather than SIP or SAP allows specification of preconditions for individual media streams. For example, a multimedia lecture might require reservation for the audio, but not the video (which is less important).

Our extensions are completely backward compatible. If a recipient does not understand them, normal SIP or SAP processing will occur, at no penalty of call setup latency.

4.1 Requirements

The basic motivation in this work is to meet and possibly exceed the user expectations and human factors associated with telephony.

In this section, we briefly describe the application requirements that led to the set of DCS signaling design principles. In its basic implementation, DCS supports a residential telephone service comparable to the local telephone services offered today. Some of the requirements for resource management, in concert with call signaling, are as follows:

The system must minimize call defects. These are situations where either the call never completes, or an error occurs after the destination is alerted. Requirements on call defects are typically far more stringent than call blocking. Note that we expect the provider and the endpoints to attempt to use lower bandwidth codecs as the first line of defense against limited network capacity, and to avoid blocking calls.

The system must minimize the post-dial-delay, which is the time between the user dialing the last digit and receiving positive confirmation from the network. This delay must be short enough that users do not perceive a difference with post-dial delay in the circuit switched network or conclude that the network connectivity no longer exists.

Call signaling needs to provide enough information to the resource management protocol so as to enable resources to be allocated in the network. This typically requires most if not all of the components of a packet classifier (source IP, destination IP, source port, destination port, protocol) to be available for resource allocation.

4.2 Overview

For acceptable interactive voice communication it is important to

achieve end-to-end QoS. The end-to-end QoS assurance implies achieving low packet delay and packet loss. End-to-end packet delay must be small enough that it does not interfere with normal voice conversations. The ITU recommends no greater than 300 ms roundtrip delay for telephony service. Packet loss must be small enough to not perceptibly impede voice quality or the performance of fax and voice band modems.

If it is found that the network cannot guarantee end-to-end QoS resources, there are two alternatives: either (1) allow call signaling to proceed with high probability of excessive delay and packet loss which could impair any interactive real-time communication between the participants, or (2) block the call prior to the called party being alerted. When calls are blocked because of a lack of resources in a particular segment of the network, it is highly desirable that such blocking occur before the called party picks up.

We do expect the endpoints to attempt to use lower bandwidth codecs, thereby avoiding blocking calls, as the first line of defense against limited network capacity.

The call signaling and resource reservation must be achieved in such a way that the post-dial-delay must be minimized without increasing the probability of call defects. This means that the number of round-trip messages must be kept at an absolute minimum and messages must be sent directly end-system to end-system if possible.

The general idea behind the extension is simple. We define two new SDP attributes, "qos" and "security". The "qos" attribute indicates whether end-to-end resource reservation is optional or mandatory, and in which direction (send, recv, or sendrecv). When the attribute indicates mandatory, this means that the participant who has received the SDP does not proceed with participation in the session until resource reservation has completed in the direction indicated. In this case, "not proceeding" means that the participant behaves as if they had not received the SDP at all. If the attribute indicates that QoS for the stream is optional, then the participant proceeds normally with the session, but should reserve network resources in the direction indicated, if they are capable. Absence of the "qos" attribute means the participant reserves resources for this stream, and proceeds normally with the session. This behavior is the normal behavior for SDP.

Resource reservation takes place using whatever protocols participants must use, based on support by their service provider. If the ISP's of the various participants are using differing resource reservation protocols, translation is necessary, but this is done within the network, without knowledge of the participants.

The direction attribute indicates in which direction reservations should be reserved. If "send", it means reservations should be made

in the direction of media flow from the session originator to participants. If "recv", it means reservations should be made in the direction of media flow from participants to the session originator. In the case of "sendrecv", it means reservations should be made in both directions. If the direction attribute is "sendrecv" but the endpoints only support a single-direction resource reservation protocol, then both the originator and participants cooperate to ensure the agreed precondition is met.

In the case of security, the same attributes are defined - optional/mandatory, and send/recv/sendrecv. Their meaning is identical to the one above, except that a security association should be established in the given direction. The details of the security association are not signaled by SDP; these depend on the Security Policy Database of the participant.

Either party can include a "confirm" attribute in the SDP. When the "Confirm" attribute is present, the recipient sends a COMET message to the sender, with SDP attached, telling the status of each precondition as "success" or "failure." If the "confirm" attribute is present in the SDP sent by the session originator to the participant (e.g. in the SIP INVITE message), then the participant sends the COMET message to the originator. If the "confirm" attribute is present in the SDP sent by the recipient to the originator (e.g. in a SIP response message), then the originator sends the COMET message to the participant.

When the "Confirm" attribute is present in both the SDP sent by the session originator to the participant (e.g. in the SIP INVITE message), and in the SDP sent by the recipient back to the originator (e.g. in a SIP response message), the session originator would wait for the COMET message with the success/failure notification before responding with a COMET message, and responds instead with a CANCEL if a mandatory precondition is not met, or if a sufficient combination of optional preconditions are not met. The recipient does not wait for the COMET message from the originator before sending its COMET message.

The "confirm" attribute is typically used if the direction attribute is "sendrecv" and the originator or participant only supports a single-direction resource reservation protocol. In such a case, the originator (or participant) would reserve resources for one direction of media flow, and send a confirmation with a direction attribute of "send." The participant (or originator) would reserve resources for the other direction. On receipt of the COMET message, they would know that both directions have been reserved, and the precondition is met.

5. SDP Extension

The formatting of the qos attribute in the Session Description

Protocol (SDP)[3] is described by the following BNF:

```
qos-attribute    = "a=qos:" strength-tag SP direction-tag
                           [SP confirmation-tag]
strength-tag     = ("mandatory" | "optional" | "success" |
                           "failure")
direction-tag    = ("send" | "recv" | "sendrecv")
confirmation-tag = "confirm"
```

and the security attribute:

```
security-attribute = "a=secure:" SP strength-tag SP direction-tag
```



```
                           [SP confirmation-tag]
```

5.1 SDP Example

The following example shows an SDP description carried in a SIP INVITE message from A to B:

```
v=0
o=mhandley 2890844526 2890842807 IN IP4 126.16.64.4
s=SDP Seminar
i=A Seminar on the session description protocol
u=http://www.cs.ucl.ac.uk/staff/M.Handley/sdp.03.ps
e=mjh@isi.edu (Mark Handley)
c=IN IP4 224.2.17.12/127
t=2873397496 2873404696
m=audio 49170 RTP/AVP 0
a=qos:mandatory recv confirm
m=video 51372 RTP/AVP 31
a=secure:mandatory sendrecv
m=application 32416 udp wb
a=orient:portrait
a=qos:optional sendrecv
a=secure:optional sendrecv
```

This SDP indicates that B should not continue its involvement in the session until resources for the audio are reserved from B to A, and a bi-directional security association is established for the video. B can join the sessions once these preconditions are met, but should reserve resources and establish a bi-directional security association for the whiteboard.

5.2 SDP Allowable Combinations

If the recipient of the SDP (e.g. the UAS) is capable and willing to honor the precondition(s), it returns a provisional response containing SDP, along with the qos/security attributes, for each such stream. This SDP MUST be a subset of the preconditions indicated in the INVITE.

Table 1 illustrates the allowed values for the direction tag in the

response. Each row represents a value of the direction in the SIP INVITE, and each column the value in the response. An entry of N/A means that this combination is not allowed. A value of A->B (B->A) implies that the precondition is for resources reserved (or security established) from A to B (B to A). A value of A<->B means that the precondition is for resource reservation or security establishment in both directions. The value in the response is the one used by both parties.

	B: response			
A: request	send	recv	sendrecv	none
send	N/A	A->B	N/A	--
recv	B->A	N/A	N/A	--
sendrecv	A<-B	B<-A	A<->B	--
none	--	--	--	--

Table 1: Allowed values of coupling

Table 2 illustrates the allowed values for the strength tag in the request and response. A "Y" means the combination is allowed, and a "N" means it is not. The value in the response is the one used by both parties.

	B: response		
A: request	mandatory	optional	none
mandatory	Y	Y	Y
optional	N	Y	Y
none	N	N	Y

Table 2: Allowed values of strength parameter

Table 3 illustrates the allowed values for the direction tag in a confirmation message (COMET) sent from the originator to a participant, based on the value of the direction tag negotiated in the initial request and response. A "Y" means the combination is allowed, and a "N" means it is not and SHOULD be ignored by the participant.

	A: confirmation		
B: reponse	send	recv	sendrecv
A->B	Y	N	N
A<-B	N	Y	N
A<->B	Y	Y	Y

Table 3: Allowed values of direction in confirmation from A

Table 4 illustrates the allowed values for the direction tag in a confirmation message (COMET) sent from the participant to the originator, based on the value of the direction tag negotiated in the initial request and response. A "Y" means the combination is allowed, and a "N" means it is not and SHOULD be ignored by the originator.

	B: confirmation		
B: reponse	send	recv	sendrecv
A->B	N	Y	N
A<-B	Y	N	N
A<->B	Y	Y	Y

Table 4: Allowed values of direction in confirmation from B

6. SIP Extension: The COMET Method

The COMET method is used for communicating successful completion of preconditions between the user agents.

The signaling path for the COMET method is the signaling path established as a result of the call setup. This can be either direct signaling between the calling and called user agents or a signaling path involving SIP proxy servers that were involved in the call setup and added themselves to the Record-Route header on the initial INVITE message.

The precondition information is communicated in the message body, which MUST contain an SDP. For every agreed precondition, the strength-tag MUST indicate "success" or "failure".

If the initial request contained Record-Route headers, the provisional response MUST contain a copy of those headers, as if the response were a 200 OK to the initial request. Since provisional responses MAY contain Record-Route and Contact headers, the COMET request MUST contain Route headers if the Record-Route headers were present in the provisional response. The Route header is constructed as specified in [2]. The Route header that is constructed from some provisional response MUST NOT be placed in any other request except for the COMET for that provisional response.

A UAC MUST NOT insert a Route header into a COMET request if no Record-Route header was present in the response.

If the initial request was sent with credentials, the COMET request SHOULD contain those credentials as well.

The Call-ID in the COMET MUST match that of the provisional response. The CSeq in this request MUST be larger than the last request sent by this UAC for this call leg. The To, From, and Via headers MUST be present, and MUST be constructed as they would be

for a re-INVITE or BYE as specified in [2]. In particular, if the provisional response contained a tag in the To field, this tag MUST be mirrored in the To field of the COMET.

Once the COMET request is created, it is sent by the UAC. It is sent as would any other non-INVITE request for a call. In particular, when sent over UDP, the COMET request is retransmitted with an exponentially increasing interval, starting at 500 milliseconds and increasing to 4 seconds. Note that a UAC SHOULD NOT retransmit the COMET request when it receives a retransmission of the provisional response being acknowledged, although doing so does not create a protocol error. As with any other non-INVITE request, the UAC continues to retransmit the COMET request until it receives a final response.

A COMET request MAY be cancelled. However, whilst allowed for purposes of generality, usage of CANCEL with COMET is NOT RECOMMENDED.

6.1 Header Field Support for COMET Method

Tables 3 and 4 are extensions of tables 4 and 5 in the SIP specification[2]. Refer to Section 6 of [2] for a description of the content of the tables.

6.2 Responses to the COMET Request Method

If a server receives a COMET request it MUST send a final response.

A 200 OK response MUST be sent by a UAS for a COMET request if the COMET request was successfully received for an existing call. Beyond that, no additional operations are required.

A 481 Call Leg/Transaction Does Not Exist message MUST be sent by a UAS if the COMET request does not match any existing call leg.

Header	Where	COMET
Accept	R	o
Accept-Encoding	R	o
Accept-Language	R	o
Allow	200	-
Allow	405	o
Authorization	R	o
Call-ID	gc	m
Contact	R	o
Contact	1xx	-

Contact	2xx	-
Contact	3xx	-
Contact	485	-
Content-Encoding	e	o
Content-Length	e	o
Content-Type	e	*
CSeq	gc	m
Date	g	o
Encryption	g	o
Expires	g	o
From	gc	m
Hide	R	o
Max-Forwards	R	o
Organization	g	o

Table 3 Summary of header fields, A-0

Header	Where	COMET
Priority	R	o
Proxy-Authenticate	407	o
Proxy-Authorization	R	o
Proxy-Require	R	o
Require	R	o
Retry-After	R	-
Retry-After	404,480,486	o
Retry-After	503	o
Retry-After	600,603	o
Response-Key	R	o
Record-Route	R	o
Record-Route	2xx	o
Route	R	o
Server	r	o
Subject	R	o
Timestamp	g	o
To	gc(1)	m
Unsupported	420	o
User-Agent	g	o
Via	gc(2)	m
Warning	r	o
WWW-Authenticate	401	o

Table 4 Summary of header fields, P-Z

Other request failure (4xx), Server Failure (5xx) and Global Failure (6xx) responses MAY be sent for the COMET Request.

6.3 Message Body Inclusion

The COMET request MUST contain a message body, with Content-type "application/sdp".

6.4 Behavior of SIP User Agents

Unless stated otherwise, the protocol rules for the COMET request governing the usage of tags, Route and Record-Route, retransmission and reliability, CSeq incrementing and message formatting follow those in [2] as defined for the BYE request.

A COMET request MAY be cancelled. A UAS receiving a CANCEL for a COMET request SHOULD respond to the COMET with a "487 Request Cancelled" response if a final response has not been sent to the COMET and then behave as if the request were never received.

6.5 Behavior of SIP Proxy and Redirect Servers

6.5.1 Proxy Server

Unless stated otherwise, the protocol rules for the COMET request at a proxy are identical to those for a BYE request as specified in [2].

6.5.2 Forking Proxy Server

Unless stated otherwise, the protocol rules for the COMET request at a proxy are identical to those for a BYE request as specified in [2].

6.5.3 Redirection Server

Unless stated otherwise, the protocol rules for the COMET request at a proxy are identical to those for a BYE request as specified in [2].

7. SIP Extension: The 183-Session-Progress Response

An additional provisional response is defined by this draft, which is returned by a UAS to convey information not otherwise classified.

7.1 Status Code and Reason Phrase

The following is to be added to Figure 4 in Section 5.1.1, Informational and success Status codes.

```
Informational = "183"  ;Session-Progress
```

7.2 Status Code Definition

The following is to be added to a new section 7.1.5

7.1.5 183 Session Progress

The 183 (Session Progress) response is used to convey information about the progress of the call which is not otherwise classified. The Reason-Phrase MAY be used to convey more details about the call progress.

The Session Progress response MAY contain a message body. If so, it MUST contain a "Content-Disposition" header indicating the proper treatment of the message body.

8. SIP Extension: The 580-Precondition-Failure Response

An additional error response is defined by this draft, which is returned by a UAS if it is unable to perform the mandatory preconditions for the session.

8.1 Status Code and Reason Phrase

The following is to be added to Figure 8, Server error status codes

```
Server-Error =  "580"  ;Precondition-Failure
```

8.2 Status Code Definition

The following is to be added to a new section 7.5.7.

7.5.7 580 Precondition Failure

The server was unable to establish the qos or security association mandated by the SDP precondition.

The Precondition Failure response MUST contain a message body, with Content-Type "application/sdp", giving the specifics of the precondition that could not be met.

9. SIP Extension: Content-Disposition header

An additional entity header is defined by this draft, which is returned by a UAS in a provisional response indicating preconditions for the session.

The following is to be added to Table 3, SIP headers, in Section 3.

```
Entity-header   = Content-Disposition   ; Section 6.14a
```

The following entry is to be added to Table 4, Summary of header fields, A-O, in Section 6.

	where	enc	e-e	ACK	BYE	CAN	INV	OPT	REG
Content-Disposition	e		e	o	o	-	o	o	o

The following is to be added to a new section after 6.14.

6.14a Content-Disposition

```
Content-disposition  = "Content-Disposition" ":"
                        Disposition-type *( ";" disp-param)
Disposition-type     = "precondition" | disp-extension-token
Disp-extension-token = token
Disp-param           = "handling" "=" "optional" | "required"
                        | other-handling
Other-handling       = token
```

The Content-Disposition header field describes how the message body is to be interpreted by the UAC or UAS.

The value "precondition" indicates the body part describes QoS and/or security preconditions that SHOULD be established prior to the start of the session.

The handling parameter, disp-param, describes how the UAC or UAS should react if it receives a message body whose content type or disposition type it does not understand. If the parameter has the value "optional" the UAS MUST ignore the message body; if it has the value "required" the UAS MUST return 415 (Unsupported Media Type). If the handling parameter is missing, the value "required" is to be assumed.

10. Option tag for Requires and Supported headers

This draft defines the option tag "precondition" for use in the Require and Supported headers [12].

A UAS that supports this extension MUST respond to an OPTION request with a Supported header that includes the "precondition" tag.

A UAC MAY include a "Require: precondition" in an INVITE request if it wants the session initiation to fail if the UAS does not support this feature.

Presence of the precondition entries in the SDP message body of an INVITE request or response indicates support of this feature. The UAC or UAS MAY in addition include a "Supported: precondition" header in the request or response.

11. SIP Usage Rules

11.1 Overview

The session originator (UAC) prepares an SDP message body for the INVITE describing the desired QoS and security preconditions for each media flow, and the desired directions. The token value "send" means the direction of media from originator (whichever entity created the SDP) to recipient (whichever entity received the SDP in

a SIP message), and "recv" is from recipient to originator. In an INVITE, the UAC is the originator, and the UAS is the recipient. The roles are reversed in the response.

The recipient of the INVITE (UAS) returns a 18x provisional response containing a Content-Disposition of "precondition," and SDP with the qos/security attribute for each stream having a precondition. The preconditions in this SDP (i.e. strength tag and direction tag) are equal to, or a subset of, the preconditions indicated in the INVITE. The UAS would typically include a confirmation request in this SDP. Unlike normal SIP processing, the UAS MUST NOT alert the called user at this point. The UAS now attempts to reserve the qos resources and establish the security associations.

The 18x provisional response is received by the UAC. If the 18x contained SDP with mandatory qos/security parameters, the UAC does not let the session proceed until the mandatory preconditions are met. The UAC attempts to reserve the needed resources and establish the security associations.

If either party requests a confirmation, a COMET message is sent to that party. The COMET message contains the success/failure indication for each precondition. For a precondition with a direction value of "sendrecv," the COMET indicates whether the sender is able to confirm both directions or only one direction. Upon receipt of the COMET message, the UAC/UAS continues normal SIP call handling. For a UAS this includes alerting the user and

sending a 180-Ringing or 200-OK response. The UAC SIP transaction completes normally.

Note that this extension requires usage of reliable provisional responses [11]. This is because the 18x contains SDP with information required for the session originator to initiate reservations towards the participant.

11.2 Behavior of Originator (UAC)

The session originator (UAC) MAY include QoS and security preconditions (including the desired direction) for each media flow in the SDP sent with the INVITE. The token value "send" means the direction of media from originator (whichever entity created the SDP) to recipient (whichever entity received the SDP in a SIP message). The token value "recv" means the direction of media from recipient to originator. If preconditions are included in the INVITE request, the UAC MUST indicate support for reliable provisional responses [11].

If the UAC receives a 18x provisional response without a Content-Disposition of "precondition," or without SDP, or with SDP but without any qos/security preconditions in any stream, UAC treats it as an indication that the UAS is unable or unwilling to perform the

preconditions requested. As such, the UAC SHOULD proceed with normal call setup procedures.

If the 18x provisional response contained a Content-Disposition of "precondition" and contained SDP with mandatory qos/security parameters, the UAC SHOULD NOT let the session proceed until the mandatory preconditions are met.

If the 18x provisional response with preconditions requested an acknowledgement (using the methods of [11]), the UAC MUST include an updated SDP in the PRACK if the UAC modified the original SDP based on the response from the UAS. Such a modification MAY be due to negotiation of compatible codecs, or MAY be due to negotiation of mandatory preconditions. If the provisional response received from the UAS is a 180-Ringing, and the direction value of a mandatory precondition is "sendrecv," and the UAC uses a one-way QoS mechanism (such as RSVP), the updated SDP in the PRACK SHOULD request a confirmation from the UAS so that the bi-directional precondition can be satisfied before performing the requested alerting function.

Upon receipt of the 18x provisional response with preconditions, the UAC MUST initiate the qos reservations and establish the security associations to the best of its capabilities.

If the UAC had requested confirmation in the initial SDP, it MAY wait for the COMET message from the UAS containing the success/failure status of each precondition. The UAC MAY set a local timer to limit the time waiting for preconditions to complete. The UAC SHOULD merge the success/failure status in the COMET message with its local status. For example, if the COMET message indicated

success in the "send" direction, and the UAC was also able to meet the precondition in the "send" direction, they combine to meet the precondition in the "sendrecv" direction.

If any of the mandatory preconditions cannot be met, and a confirmation was not requested by the UAS, the UAC MUST send a CANCEL and terminate the session. If any of the optional preconditions cannot be met, the UAC MAY consult with the originating customer for guidance on whether to complete the session.

When all the preconditions have either been met or have failed, and the SDP received from the UAS included a confirmation request, the UAC MUST send a COMET message to the UAS with SDP. Each precondition is updated to indicate success/failure and the appropriate direction tag is updated based on local operations performed combined with the received COMET message, if any.

The session now completes normally, as per [2].

11.3 Behavior of Destination (UAS)

On receipt of an INVITE request containing preconditions, the UAS MUST generate a 18x provisional response containing a subset of the preconditions supported by the UAS. In the response, the token value "send" means the direction of the media from the UAS to the UAC, and "recv" is from the UAC to the UAS. This is reversed from the SDP in the initial INVITE. The 18x provisional response MUST include a Content-Disposition header with parameter "precondition." If the "confirm" attribute is present in the SDP received from the UAC, or if the direction tag of a mandatory QoS precondition is "sendrecv" and the UAS only supports a one-way QoS reservation scheme (e.g. RSVP), then the UAS SHOULD include a "confirm" attribute. If the UAS is able to satisfy the preconditions immediately, and no confirmation is requested by the UAC, then a 180-Ringing response is appropriate. Otherwise a 183-Session-Progress response SHOULD be used.

If the INVITE request did not contain any preconditions, but did indicate support for reliable provisional responses[11], the UAS MAY include preconditions in a 18x provisional response to the INVITE. The 18x provisional response MUST include a Content-Disposition header with the parameter "precondition." The 18x provisional response MUST request an acknowledgement using the mechanism of [11]. If the PRACK includes an SDP without any preconditions, the UAS MAY complete the session without the preconditions, or MAY reject the INVITE request.

The UAS SHOULD request an acknowledgement to the 18x provisional response, using the mechanism of [11]. The UAS SHOULD wait for the PRACK message before initiating resource reservation to allow the resource reservation to reflect 3-way SDP negotiation, or other information available only through receipt of the PRACK.

If the INVITE request or PRACK message contained mandatory preconditions, or requested a confirmation of preconditions, the UAS MUST NOT alert the called user.

The UAS now attempts to reserve the qos resources and establish the security associations. The UAS MAY set a local timer to limit the time waiting for preconditions to complete.

If the UAS is unable to perform any mandatory precondition, and neither the UAC nor UAS requested a confirmation, the UAS MUST send a 580-Precondition-Failure response to the UAC. If the UAS is unable to perform any optional precondition, it MAY consult with the customer to obtain guidance regarding completion of the session.

When processing of all preconditions is complete, if a precondition in the initial INVITE specified a confirmation request, the UAS MUST send a COMET message to the UAC containing SDP, along with the qos/security result of success/failure for each precondition. If the direction tag of the precondition was "sendrecv" but the UAS was only able to ensure "send" or "recv," the direction tag in the COMET

MUST only indicate what the UAS ensures. The Request-URI, call-leg identification, and other headers of this COMET message are to be constructed identically to a BYE.

If the UAS had requested confirmation of a precondition in the response SDP, it SHOULD wait for the COMET message from the originator containing the success/failure indication of each precondition from the originator's point of view. The success/failure indications in the COMET message from the UAC SHOULD be combined with the local status to determine the overall success/failure of the precondition. For example, if the COMET message indicated success in the "send" direction, and the UAS was also able to meet the precondition in the "send" direction, they combine to meet the precondition in the "sendrecv" direction. If that combination indicates a failure for a mandatory precondition, the UAS MUST send a 580-Precondition-Failure response to the UAC.

Once the preconditions are met, the UAS alerts the user, and the SIP transaction proceeds normally.

The UAS MAY send additional 18x provisional responses with Content-Disposition of "precondition," and the procedures described above are repeated sequentially for each.

12. Examples

12.1 Single Media Call Flow

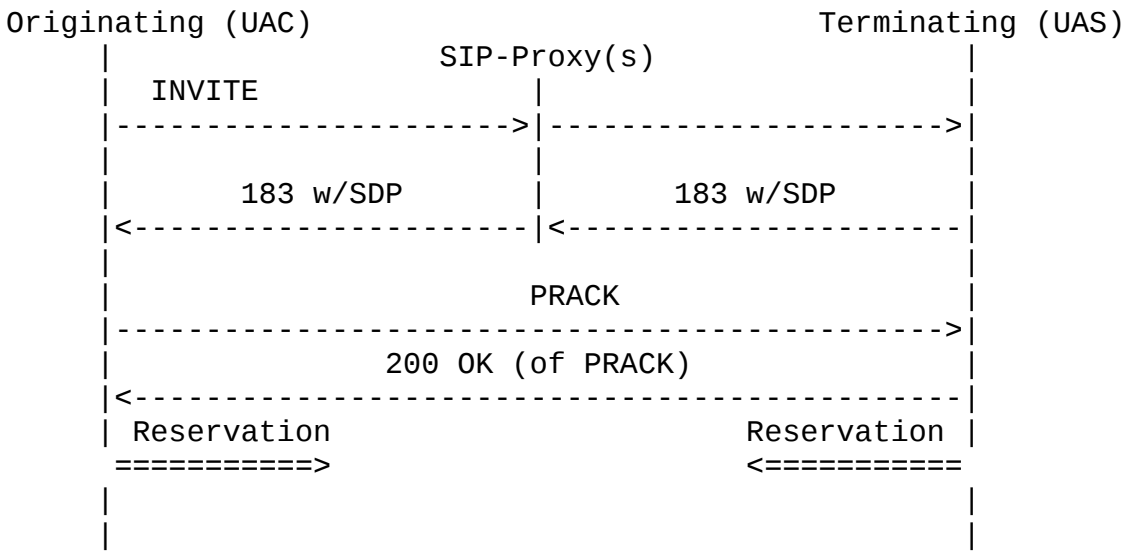
Figure 1 presents a high-level overview of a basic end-point to end-point (UAC-UAS) call flow. This example is appropriate for a single-media session with a mandatory quality-of-service "sendrecv" precondition, where both the UAC and UAS can only perform a single-direction ("send") resource reservation.

The session originator (UAC) prepares an SDP message body for the INVITE describing the desired QoS and security preconditions for each media flow, and the desired direction "sendrecv." This SDP is included in the INVITE message sent through the proxies, and includes an entry "a=qos:mandatory sendrecv."

The recipient of the INVITE (UAS), being willing to perform the requested precondition, returns a 183-Session-Progress provisional response containing SDP, along with the qos/security attribute for each stream having a precondition. Since the "sendrecv" direction tag required a cooperative effort of the UAC and UAS, the UAS requests a confirmation when the preconditions are met, with the SDP entry "a=qos:mandatory sendrecv confirm." The UAS now attempts to reserve the qos resources and establish the security associations.

The 183-Session-Progress provisional response is sent using the reliability mechanism of [11]. UAC sends the appropriate PRACK and UAS responds with a 200-OK to the PRACK.

The 183-Session-Progress is received by the UAC, and the UAC requests the resources needed in its "send" direction, and establishes the security associations. Once the preconditions are met, the UAC sends a COMET message as requested by the confirmation token. This COMET message contains an entry "a=qos:success send"



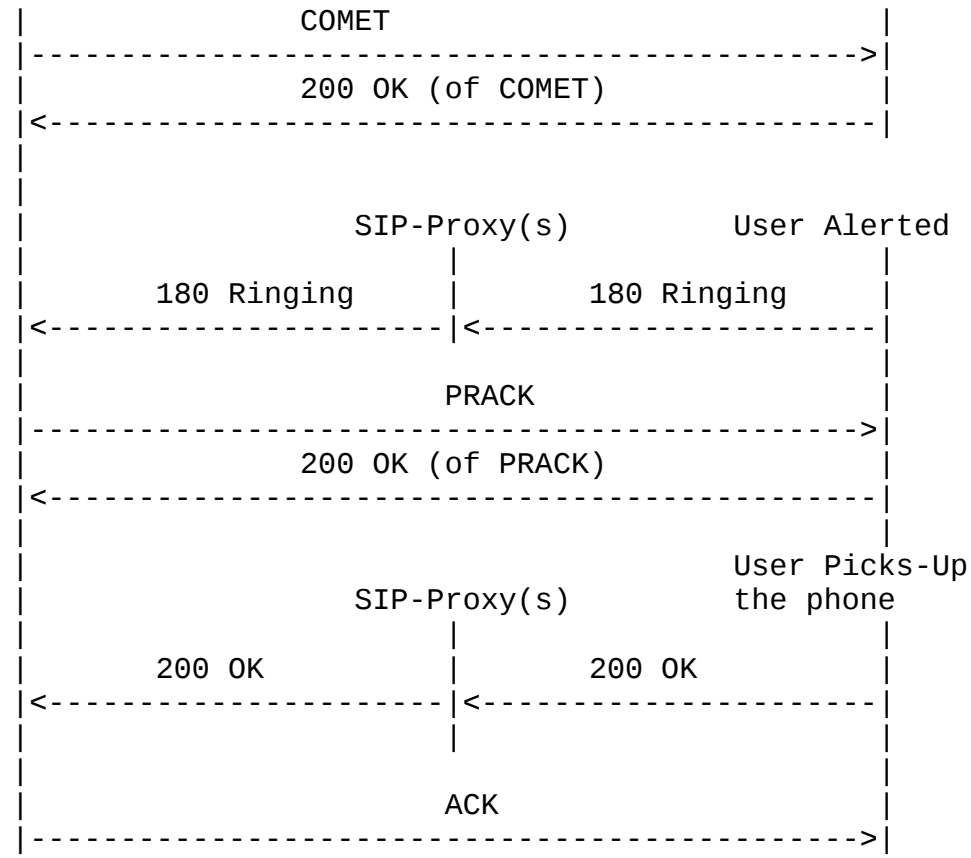


Figure 1: Basic Call Flow

The UAS successfully performs its resource reservation, in its "send" direction, and waits for the COMET message from the UAC.

On receipt of the COMET message, the UAS processes the "send" confirmation contained in the COMET SDP. The "send" confirmation from the UAC coupled with its own "send" success, allows the UAS to determine that all preconditions have been met. The UAS then continues with session establishment. At this point it alerts the user, and sends a 180-Ringing provisional response. This

provisional response is also sent using the reliability mechanism of [11], resulting in a PRACK message and 200-OK of the PRACK.

When the destination party answers, the normal SIP 200-OK final response is sent through the proxies to the originator, and the originator responds with an ACK message.

Either party can terminate the call. An endpoint that detects an "on-hook" (request to terminate the call) releases the QoS resources held for the connection, and sends a SIP BYE message to the remote endpoint, which is acknowledged with a 200-OK.

12.2 Multiple Media Call Flow

Figure 2 presents a high-level overview of an advanced end-point to end-point (UAC-UAS) call flow. This example is appropriate for a multiple-media session with some combination of mandatory and optional quality-of-service precondition. For example, the originator may suggest five media streams, and be willing to establish the session if any three of them are satisfied.

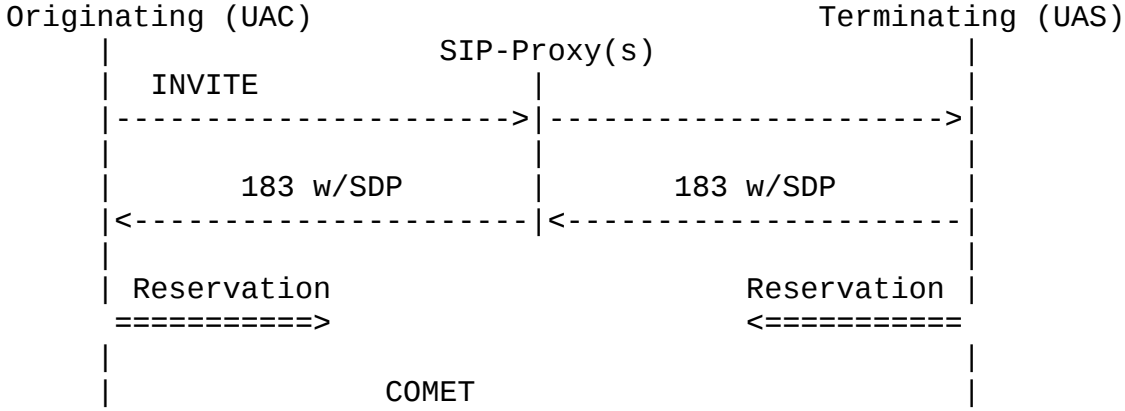
The use of reliable provisional responses is assumed, but not shown in this figure.

The session originator (UAC) prepares an SDP message body for the INVITE describing the desired QoS and security preconditions for each media flow, and the desired directions. UAC also requests confirmation of the preconditions. The UAS receiving the INVITE message responds with 183-Session-Progress, as in the previous example.

When the UAS has completed the resource reservations and security session establishment, it sends a confirmation to the UAC in the form of a COMET message, with each precondition marked in the SDP as either success or failure. Note that if UAS was not satisfied with the combination of successful preconditions, it could instead have responded with 580-Precondition-Failure, and ended the INVITE transaction.

If the UAC has satisfied its preconditions, and is satisfied with the preconditions achieved by the UAS, it responds with the COMET message. The COMET message contains the SDP with the success/failure results of each precondition attempted by UAC. If UAC is not satisfied with the combination of successful preconditions, it could instead have sent a CANCEL message.

On receipt of the COMET message, UAS examines the combination of satisfied preconditions reported by UAC, and makes a final decision whether to proceed with the session. If sufficient preconditions are not satisfied, the UAS responds with 580-Precondition-Failure. Otherwise, the session proceeds as in the previous example.



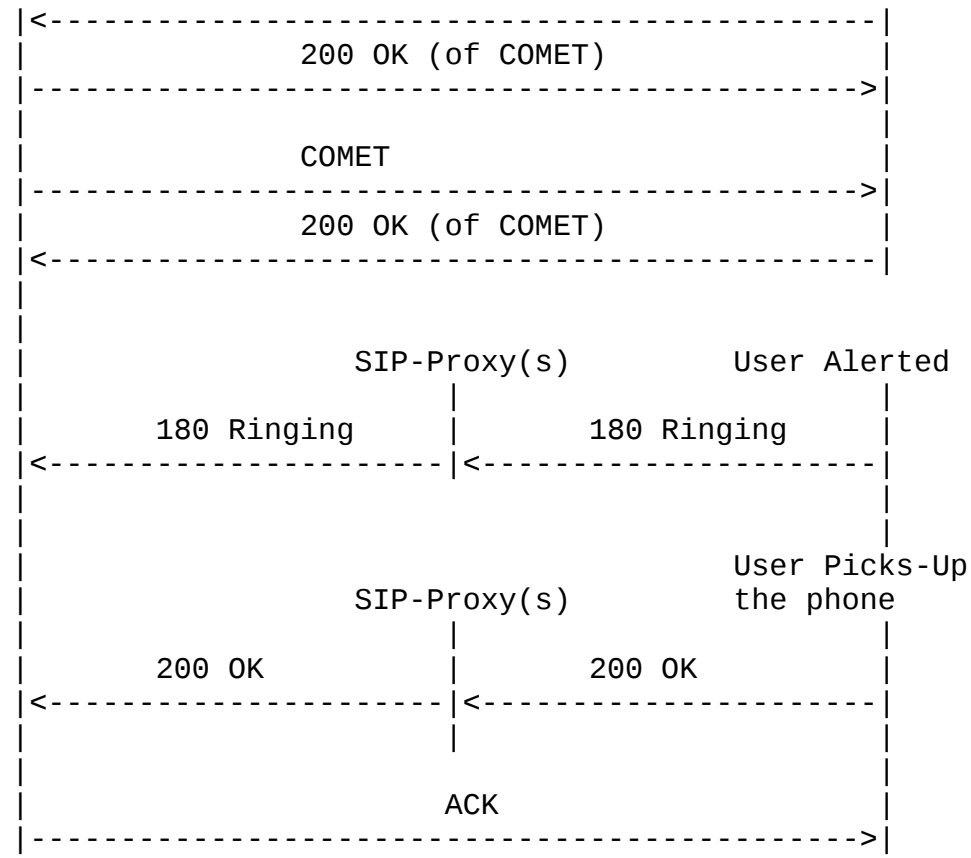


Figure 2: Call Flow with negotiation of optional preconditions

13. Special considerations with Forking Proxies

If a proxy along the signaling path between the UAC and UAS forks the INVITE request, it results in two or more UASs simultaneously sending provisional responses with preconditions. The procedures above result in the UAC handling each independently, reserving resources and responding with COMET messages as required.

This results in multiple resource reservations from the UAC, only one of which will be utilized for the final session. While functionally correct, this has the unfortunate side-effect of increasing the call blocking probability.

Customized resource allocation protocols may be used by the UAC to reduce this duplicate allocation and prevent excess blocking of calls. For one such example, see [8].

14. Advantages of the Proposed Approach

The use of two-phase call signaling makes it possible for SIP to meet user expectations that come from the telephony services.

The reservation of resources before the user is alerted makes sure

that the network resources are assured before the destination end-point is informed about the call.

The number of messages and the total delay introduced by these messages is kept to a minimum without sacrificing compatibility with SIP servers that do not implement preconditions.

15. Security Considerations

If the contents of the SDP contained in the 183-Session-Progress are private then end-to-end encryption of the message body can be used to prevent unauthorized access to the content.

The security considerations given in the SIP specification apply to the COMET method. No additional security considerations specific to the COMET method are necessary.

16. Notice Regarding Intellectual Property Rights

The IETF has been notified of intellectual property rights claimed in regard to some or all of the specification contained in this document. For more information consult the online list of claimed rights.

17. References

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

2. M. Handley, H. Schulzrinne, E. Schooler, and J. Rosenberg, "SIP: Session Initiation Protocol," RFC 2543, March 1999.

3. M. Handley and V. Jacobson, "SDP: Session Description Protocol," RFC 2327, April 1998.

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

5. R. Braden, Ed., L. Zhang, S. Berson, S. Herzog, and S. Jamin, "Resource ReSerVation protocol (RSVP) -- version 1 functional specification," RFC 2205, September, 1997.

6. P. P. Pan and H. Schulzrinne, "YESSIR: A simple reservation mechanism for the Internet," in Proc. International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV), (Cambridge, England), July 1998. Also IBM Research Technical Report TC20967. Available at http://www.cs.columbia.edu/~hgs/papers/Pan98_YESSIR.ps.gz.

7. PacketCable, Dynamic Quality of Service Specification, pkt-sp-

dqos-i01-991201, December 1, 1999. Available at http://www.packetcable.com/specs/pkt-sp-dqos-i01-991202.pdf.

8. S. Kent and R. Atkinson, "Security architecture for the internet protocol," RFC 2401, November 1998.

9. H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RTP: a Transport Protocol for Real-Time Applications," RFC 1889, January 1996.

10. M. Handley, C. Perkins, and E. Whelan, "Session Announcement Protocol," RFC2974, October, 2000.

11. "Reliability of Provisional Responses in SIP," RFC pending.

12. "The SIP Supported Header," RFC pending.

18. 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, Jay Strater, Jeff Ollis, Clive Holborow, General Instruments; Doug Newlin, Guido Schuster, Ikhlaq Sidhu, 3Com; Jiri Matousek, Bay Networks; Farzi Khazai, Nortel; John Chapman, Bill Guckel, Cisco; Chuck Kalmanek, Doug Nortz, John Lawser, James Cheng, Tung-Hai Hsiao, Partho Mishra, AT&T; Telcordia Technologies; and Lucent Cable Communications.

19. 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
Terayon
Louisville, CO 80027
Email: E.Miller@terayon.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

Michael Ramalho
Cisco
Wall Township, NJ
Email: mramalho@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
Copper Mountain Networks
San Diego, CA 92121
Email: jfellows@coppermountain.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

Adam Roach
Ericsson
Richardson, TX 75081
Email: adam.roach@ericsson.com

Jonathan Rosenberg
dynamicsoft
West Orange, NJ 07052
Email: jdrosen@dynamicsoft.com

Dean Willis
dynamicsoft
West Orange, NJ 07052
Email: dwillis@dynamicsoft.com

Steve Donovan
dynamicsoft
West Orange, NJ 07052
Email: sdonovan@dynamicsoft.com

Henning Schulzrinne
Columbia University
New York, NY
Email: schulzrinne@cs.columbia.edu

Expiration Date This memo is filed as <draft-ietf-sip-manyfolks-resource-02.txt>, and expires February 28, 2002.