SIP
Session Initiation Protocol

Nicolas Montavont
nicolas.montavont@telecom-bretagne.eu

Henning Schulzrinne
Department of Computer Science
Columbia University, New York, USA

Jonathan Rosenberg
Cisco Systems

About the slides

- **These slides are part of the module on Voice over IP at Universidad de Los Andes - Venezuela**
- **Many thanks to (alphabetic order)**
  - Annie Gravey (Telecom Bretagne)
  - David Ros (Telecom Bretagne)
  - Emil Ivov (jitsi)
  - G6 - the french IPv6 task force
  - German Castignani (Telecom Bretagne)
  - Gilbert Martineau (Telecom Bretagne
  - Kurose and Ross - Computer Network, a top down approach
  - Laurent Toutain (Telecom Bretagne)
  - Xavier Lagrange (Telecom Bretagne)

Outline

- Introduction - History
- Basics of SIP
- Transactions
  - INVITE client and server transactions
  - non-INVITE client and server transactions
- SIP headers
- NAT traversal

Short History

- **Developments of SIP fall under MMUSIC within IETF**
  - Multiparty Multimedia Session Control (MMUSIC)
- **February 1996**
  - Session Invitation Protocol (SIPv1) Internet Draft
    - Mark Handley & Eve Schooler
    - Purpose was to invite registered users to conference sessions
    - Specified SDP and UDP
  - Simple Conferencing Invitation Protocol (SCIP) Internet Draft
    - Henning Schulzrinne
    - Purpose was to invite users to point to point and multicast sessions
    - Used email identifiers, TCP, but defined its own format for session description
- **December 1996**
  - Session Initiation Protocol (SIPv2) Internet Draft
    - Handley, Schooler & Schulzrinne
    - HTTP based, could use UDP or TCP, and SDP for session description
    - Jonathan Rosenberg became co-author in 1998
- **February 1999**
  - SIP became a proposed standard, published as RFC 2543

Short History

- **March 2001: SIP Working group split:**
  - **SIP** Fundamental specification and its extensions
  - **SIPPING** Applications that use SIP
- **Notification services added later:**
  - **SIMPLE** IETF WG for Instant Messaging and Presence using SIP
- **June 2002 new version published: RFC3261 obsoletes RFC 2543**
- **Today, there are many WGs, including:**
  - mmusic - Multiparty Multimedia Session Control
  - p2psip - Peer-to-Peer Session Initiation Protocol
  - simple - SIP for Instant Messaging and Presence Leveraging Extensions
  - sipclf - SIP Common Log Format
  - sipcore - Session Initiation Protocol Core

See: http://www.ietf.org/html.charters/sip-charter.html
http://www.ietf.org/html.charters/sipping-charter.html
http://www.ietf.org/html.charters/simple-charter.html

Standardisation

- **SIP @ IETF (ietf.org) - several RFC**
  - RFC 3261 : SIP: Session Initiation Protocol
  - RFC 3262: Reliability of Provisional Responses in Session Initiation Protocol (SIP)
  - RFC 3263: Session Initiation Protocol (SIP): Locating SIP Servers
  - RFC 3264: An Offer/Answer Model with Session Description Protocol (SDP)
  - RFC 3265: Session Initiation Protocol (SIP)-Specific Event Notification
  - RFC 3266: Support for IPv6 in Session Description Protocol (SDP)
  - RFC 3428: SIP Message Extension
  - RFC 5411: A Hitchhiker's Guide to the Session Initiation Protocol (SIP)
  - RFC 3665 - BCP 75: Session Initiation Protocol (SIP) Basic Call Flow Examples
  - RFC 5359 - BCP 144: Session Initiation Protocol Service Examples

For what the signalling is used for

- **Location**
  - Users can connect anywhere, we need a location mechanism
  - Separation between location and identification
- **Codec negotiation**
  - We must agree on how we'll talk
- **Port numbers**
  - On which port number I will send you my RTP?
- **Call control**
  - How is calling, billing, etc

The Session Initiation Protocol

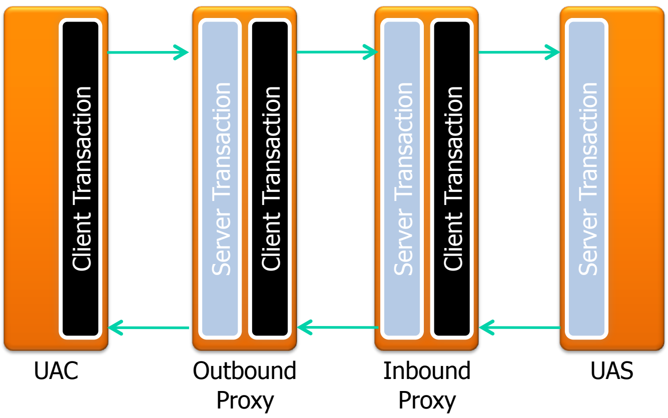
- **The Session Initiation Protocol (SIP) is an application-layer control protocol that can establish, modify, and terminate different kinds of sessions such as Internet telephony calls**
  - Request/response protocol (like HTTP)
  - Uses a <header:value> format (like SMTP)
  - Simple and extensible
  - Designed for mobility (proxy/redirect servers)
  - Authentication
  - Capability negotiation
  - User status / availability
  - Works on any transport: UDP, TCP, SCTP, ATM
- **SIP relies on**
  - RTP / RTCP to transport the media
  - SDP for describing multimedia session
  - MEGACO / MGCP for controlling gateways to the PSTN
- **SIP is used for signaling:**
  - Instant Messaging sessions
  - Phone calls over the Internet
  - Gaming servers

Key Elements

- **Client /server model**
  - Determined by the initiator of the requests
- **Client / server Exchange**
  - Transaction: request - response
  - Dialog: SIP relation SIP between 2 UA which lasts, indicate the context on how to interpret SIP messages
- **User Agent : endpoint**
  - Commands issued by user (human or gateway) and act as an agent to set up and clear down sessions
- **URI**
  - Identification of the users
  - sip:user@host
- **Proxy**
  - May play the role of the client or the server
  - Rendez-vous point

SIP User Agent

- **User Agent**
  - Commands issued by user (human or gateway) and act as an agent to set up and clear down sessions
  - Act as a UA Client
    - Generate requests
  - Act as a UA server
    - Respond to request



SIP servers

- **Servers - Applications that accept SIP request and respond to them.**
  - **Proxy server -** receive request from UA or another proxy and act on behalf of the UA in forwarding or responding to the request
    - Help in routing SIP messages
    - Can be used to enforce policies
    - Transaction stateless / stateful
    - Call stateless / stateful
    - Preserve the end-to-end transparency
  - Forking proxy: routes call requests
    - Duplicates (“forks”) requests
    - Forward only one final answer back to the UAC
  - B2B UA / ALG: Application Layer Gateway
    - Reformulate requests
    - (See next slide)
- **Redirect server**
  - respond to but do not forward request
  - Return new locations for servers
- **Registration server -** aka Registrar, bind a SIP URI to an address of Record (IP address)

Application Layer Gateway

- A back-to-back user agent (B2BUA) is a logical entity that receives a request and processes it as a **user agent server** (UAS). In order to determine how the request should be answered, it acts as a **user agent client** (UAC) and generates requests. Unlike a proxy server, it **maintains dialog state** and must participate in all requests sent on the dialogs it has established. Since it is a concatenation of a UAC and UAS, no explicit definitions are needed for its behavior.

- **Can be used as anonymizer**
- **(Break the end-to-end nature of SIP, constitutes a single point of failure)**
- **Difference between a proxy server and an ALG**
  - A proxy server does not issue Requests; it only responds to requests from a user agent (except for CANCEL)
  - A proxy server has no media capabilities
  - A proxy server does not parse message bodies; it relies exclusively on header fields

SIP URIs

Identifier - locator split

- **Location**
  - Dynamic
  - IP address: depend on the point of attachement
- **Identifier**
  - URI - Unique Resource Identifier
  - ex: nicolas@jitsi.org

Which user in this domain

Domain name, that allows to find a proxy / a server

- **2 URI categories**
  - For user: known as **Address of Record**
  - For a device or end-point: temporarily allocated to a user, indicated in the contact field

Examples of URI

- SIP URI with username:
  - `sip:andrea@sip-communicator.org`
  - `sip:andrea@starsip.tilab.com`
  - `sip:andrea@163.162.3.19`
- SIP URI without a username:
  - `sip:example.com`
  - `sip:x.example.com`
  - `sip:163.162.3.19`
- SIP URI with parameters:
  - `sip:abc@example.com;transport=tcp;user=phone`
- IPv6 SIP URI:
  - `sip:andrea@[fe80::5445:5245:444f]:5560`

TELECOM Bretagne

Messages

Message Structure: First Line

- The first line, determines the semantical type of the message:
  - Request
  - Response
- Request line contains:
  - method: determines the type of the request
  - SIP URI: determines the destination of the request
  - SIP protocol version

```
<METHOD> <Request-URI> SIP/2.0
```

- Response line contains
  - SIP protocol version
  - status-code: digital response code
  - reason phrase

```
SIP/2.0 <Status-Code> <Reason-Phrase>
```

*1xx informational responses are not retransmitted if lost **
2xx success responses are delivered to the end with reliability
3xx - 6xx non-successful responses delivered hop-by-hop

The Session Initiation Protocol

- Methods are the "verbs" of the protocol
- Original six methods in version 2.0 of SIP
  INVITE
  REGISTER
  BYE
  ACK
  CANCEL
  OPTIONS

Request Format

Request line
Several Headers
Empty Line
Message Body

- Both Requests and Responses can carry SIP bodies
  - usually SDP, but could be a JPEG or JAVA script
- SIP Responses carry a status code and a reason phrase - human readable

Response Format

Status line
Several Headers
Empty Line
Message Body

SIP Response Codes

CODE RANGE	RESPONSE CLASS		EXAMPLES
1XX	Informational	Provisional	-Queued, Ringing, Being Forwarded
2XX	Success	Final	-OK, Accepted
3XX	Redirection	Final	-Moved Temporarily, Moved Permanently
4XX	Client error	Final	-Payment Required, Method Not Allowed
5XX	Server error	Final	-Not Implemented, Service Unavailable
6XX	Global failure	Final	-Busy Everywhere, Decline

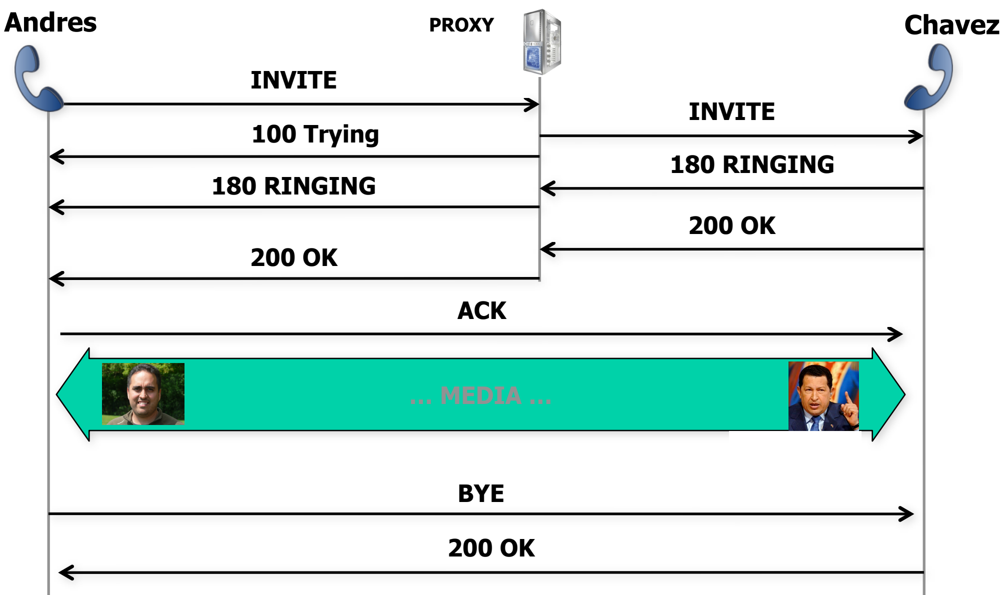
SIP Header

```
Request message :

  <METHOD> <Request-URI> SIP/2.0 CRLF
  <Header1>: <Value1> CRLF
  <Header2>: <Value2> CRLF
  <HeaderN>: <ValueN> CRLF
   CRLF
  <Message Body>

Response message:
  SIP/2.0 <Status-Code> <Reason-Phrase> CRLF
  <Header1>: <Value1> CRLF
  <Header2>: <Value2> CRLF
  <HeaderN>: <ValueN> CRLF
  CRLF
  <Message Body>
```


SIP call



Requests

- **INVITE:** Start / modify sessions
- **ACK :** Acknowledge the reception of a final response to an INVITE
- **CANCEL:** cancel a pending INVITE
- **UPDATE**: Update the parameters of a pending session
- **BYE:** ends a session
- **OPTIONS:** Request the supported features
- **REGISTER:** Attach an IP address to a SIP URI
- **REFER:** request a UA to access a URI or URL
- **SUBSCRIBE:** establish a subscription to receive a notification about an event
- **NOTIFY:** Convey information about the occurence of a particular event
- **PRACK:** acknowledge receipt of reliably transported provisional response
- **MESSAGE:** Transport instant message using SIP
- **INFO:** Send call signalling information to another user agent with which it has an established media session

TELECOM Bretagne

Responses

- **Responses**
  - 1xx : information
  - 200 : succès
  - 3xx : redirection
  - 4xx : erreur client
  - 5xx : erreur serveur
  - 6xx : échec

TELECOM Bretagne

Error Codes

Note:
Many codes are same as HTTP
SIP specific codes start x80

Informational

100	Trying
180	Ringing
181	Call Is Being Forwarded
182	Queued
183	Session Progress

Success

200	OK
202	Accepted

Redirection

300	Multiple Choices
301	Moved Permanently
302	Moved Temporarily
303	See Other
305	Use Proxy
380	Alternative Service

Client error

400	Bad Request
401	Unauthorized
402	Payment Required
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Request Timeout
409	Conflict
410	Gone
411	Length Required
413	Request Entity Too Large
414	Request-URI Too Large
415	Unsupported Media Type
420	Bad Extension
480	Temporarily not available
481	Call Leg/Transaction Does Not Exist
482	Loop Detected
483	Too Many Hops
484	Address Incomplete
485	Ambiguous
486	Busy Here
487	Request Cancelled
488	Not Acceptable Here

Server error

500	Internal Server Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Time-out
505	SIP Version not supported

Global failure

600	Busy Everywhere
603	Decline
604	Does not exist anywhere
606	Not Acceptable

TELECOM Bretagne

SIP Headers

Examples of Headers Used in Requests and Responses

HEADER	FUNCTION
Call-ID	-Used to uniquely identify a call between two user agents
Contact	-Used to convey URL of original resource requested or request originator
CSeq	-Command Sequence identifies out of sequence requests & retransmissions
From	-Identifies originator of request
To	-Indicates recipient of request
Subject	-Optional header indicating subject of media session
Content-Length	-Number of octets in the message body
Content-Type	-Indicates Internet media type. If not present application/SDP is assumed
User Agent	-Provides additional information about the user agent e.g. manufacturer
Server	-Provides additional information about the User Agent Server
Via	-Records the route taken by a request and used to route response
Record-Route	-Used to force all requests between UAs to be routed through a Proxy
Route	-Forces routing through a path extracted from a Record-Route header
Max-forwards	-limit the number of hops a request can make on the way to its destination (70)
Authorization	-Carries credentials of user agent to a server
Encryption	-Used to specify the portion of a SIP message that has been encrypted
Hide	-Requests next hop proxy to encrypt the Via headers
Priority	-Allow the user agent to set the priority of a request: e.g. urgent, emergency
Supported	-List one more options implemented in a user agent or server
Unsupported	-Indicates features that are not supported by the server

Dialog identification: call-id, local tag (after the from), remote tag (after the to)

Transaction identification: branch parameter in the via header

Minimum required fields

Additional required field for an INVITE

TELECOM Bretagne

SIP Request

```
INVITE sip:barbara@b.com SIP/2.0
Via: SIP/2.0/UDP 10.43.122.3;branch=1
From: sip:alice@a.com;tag=4ad340f
To: sip:barbara@b.com
Contact: <sip:alice@10.43.122.3>
Call-ID: 1874630@10.43.122.3
Cseq: 12442 INVITE


v=0
o=user 14341433 14341433 IP4 10.43.122.3
s=.
t=0 0
c=IN IP4 10.43.122.3
m=audio 13222 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```


TELECOM Bretagne

SIP Request

Request line	`INVITE sip:barbara@b.com SIP/2.0`	
Headers	`Via: SIP/2.0/UDP 10.43.122.3;branch=1`	
	`From: sip:alice@a.com;tag=4ad340f`	→ User URI
	`To: sip:barbara@b.com`	
	`Contact: <sip:alice@10.43.122.3>`	→ Device URI
	`Call-ID: 1874630@10.43.122.3`	
	`Cseq: 12442 INVITE`	
Empty line		
Body	`v=0`	
	`o=user 14341433 14341433 IP4 10.43.122.3`	
	`s=.`	
	`t=0 0`	
	`c=IN IP4 10.43.122.3`	
	`m=audio 13222 RTP/AVP 0`	
	`a=rtpmap:0 PCMU/8000`	

TELECOM Bretagne

SIP Response

```
SIP/2.0 404 Not Found
Via: SIP/2.0/UDP 10.43.122.3; branch=1
From: sip:alice@a.com;tag=4ad340f
To: sip:barbara@b.com;tag=4435211
Call-ID: 1874630@10.43.122.3
Cseq: 12442 INVITE
```


SIP Response

Response line	SIP/2.0 404 Not Found
Headers	Via: SIP/2.0/UDP 10.43.122.3; branch=1 From: sip:alice@a.com;tag=4ad340f To: sip:barbara@b.com;tag=4435211 Call-ID: 1874630@10.43.122.3 Cseq: 12442 INVITE
Empty line	

Methods

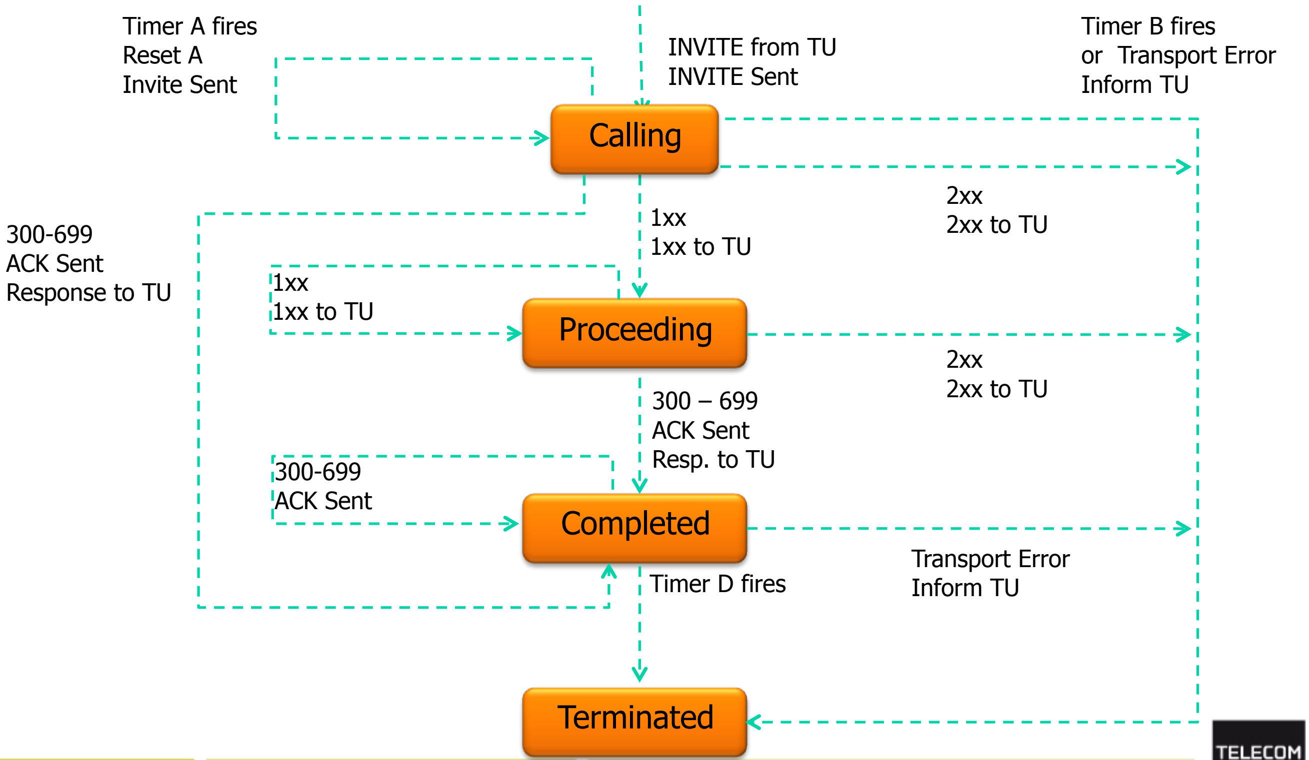
➡ *INVITE*

Methods

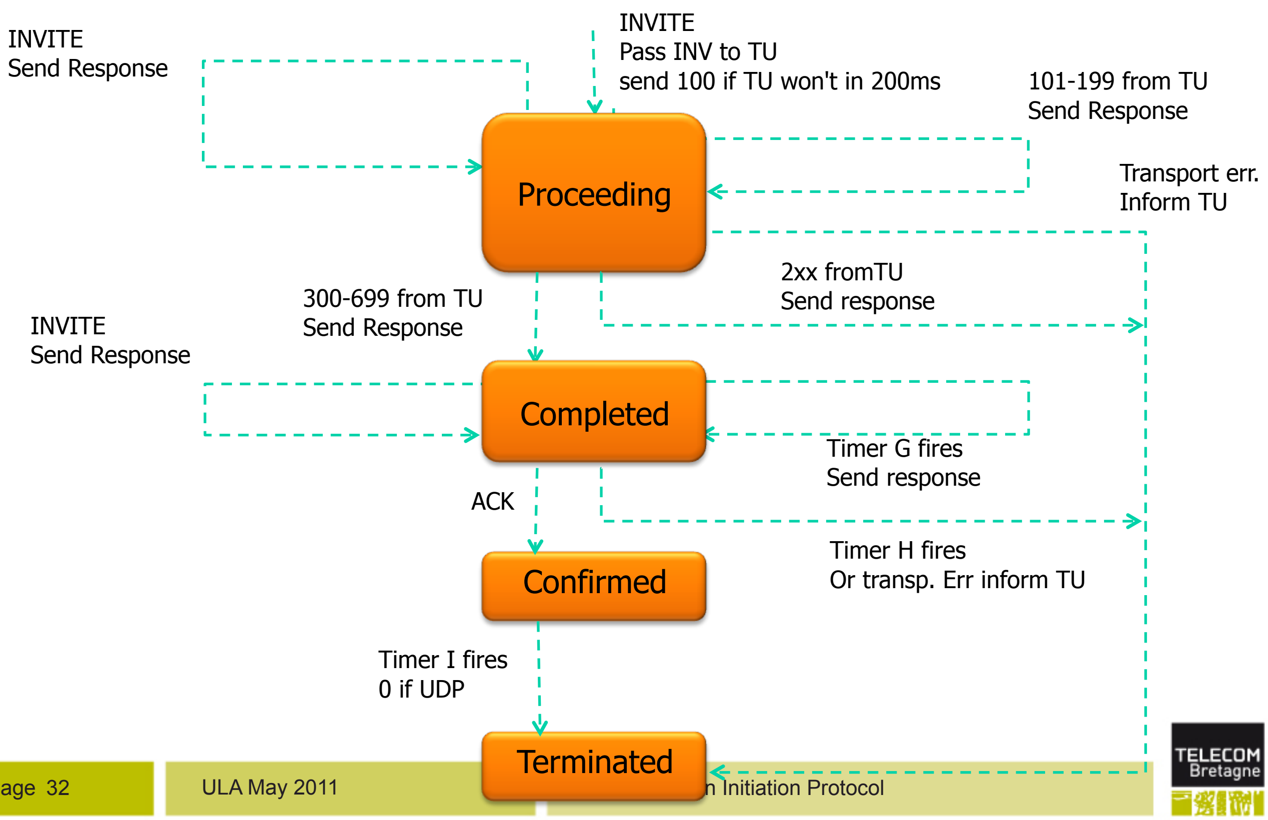
- **INVITE - establish media session between user agents**
  - Equivalent to Set Up message of Q.931
  - Always acknowledged with an ACK method
  - Usually contain a message body with the session description
  - A session is established only when the INVITE, 200 OK and ACK have been exchanged
  - A session (opened with an INVITE) is closed with a BYE method
  - An INVITE establishes a **dialog** identified with Call-ID, to and from tags

- **BYE - Terminate an established media session**
  - Equivalent to the Release message of Q.931
  - Only sent by user agents participating in the session (never by a proxy)

INVITE Client transaction



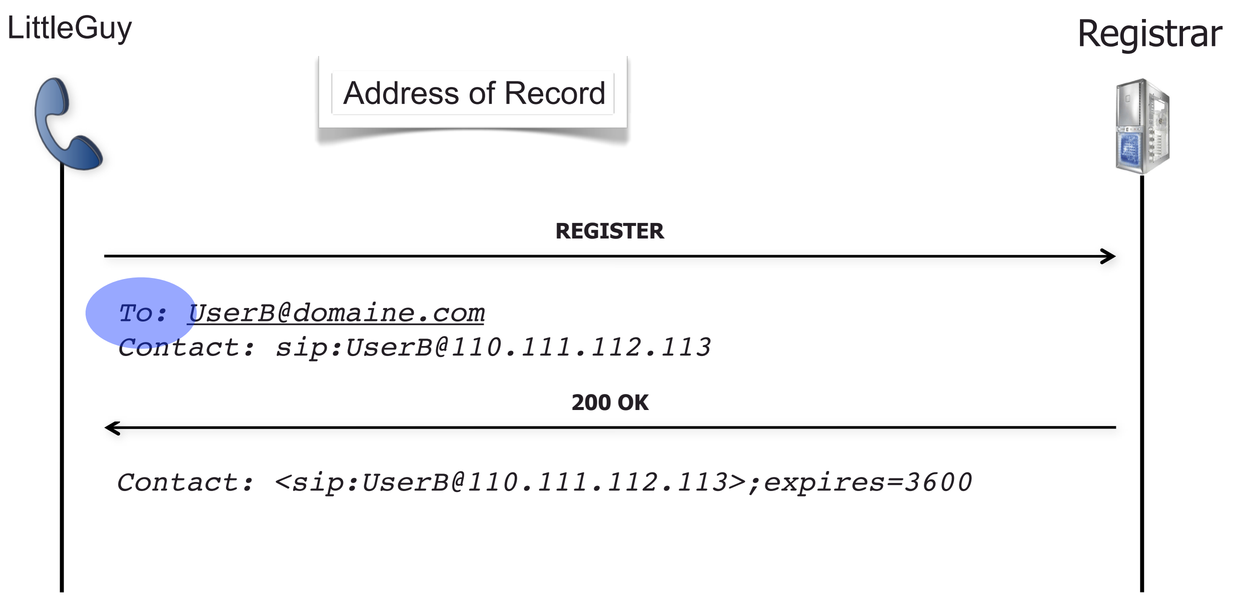
INVITE server transaction



Methods

➡ *Non-INVITE*

Registration



Authenticated registration

- **REGISTER - Notify a SIP network about the current contact URI (IP address) of the user agent**

An example REGISTER request

```
REGISTER sip:b.com SIP/2.0
Via: SIP/2.0/UDP 192.168.15.2
From: sip:barbara@b.com;tag=199257
To: sip:barbara@b.com
Contact: <sip:b@192.168.15.2>
Expires: 45
Call-ID: 950398549@192.168.15.2
CSeq: 1 REGISTER
```

An example REGISTER request

Request-URI registration domain	`REGISTER sip:b.com SIP/2.0`
	`Via: SIP/2.0/UDP 192.168.15.2`
Who's registering	`From: sip:barbara@b.com;tag=199257`
AOR	`To: sip:barbara@b.com`
Contact	`Contact: <sip:barbara@192.168.15.2>`
Duration in minutes	`Expires: 45`
	`Call-ID: 950398549@192.168.15.2` `CSeq: 1 REGISTER`
Empty line	

An example REGISTER response

```
SIP/2.0 200 Ok
Via: SIP/2.0/UDP 192.168.15.2
From: sip:barbara@b.com;tag=199257
To: sip:barbara@b.com;tag=jjf223
Contact:<sip:barbara@192.168.15.2>;expires=
2700
Contact:<sip:10.0.0.1>;expires=345
Contact:<sip:10.0.0.2>;expires=1000
Call-ID: 950398549@192.168.15.2
CSeq: 1 REGISTER
```

An example REGISTER response

	`SIP/2.0 200 Ok`
	`Via: SIP/2.0/UDP 192.168.15.2`
Who's regisering	`From: sip:barbara@b.com;tag=199257`
AOR	`To: sip:barbara@b.com;tag=jjf223`

List of all Contact headers for know AORs	`Contact:<sip:barbara@192.168.15.2>;expires=2700`
	`Contact:<sip:10.0.0.1>;expires=345`
	`Contact:<sip:10.0.0.2>;expires=1000`

	`Call-ID: 950398549@192.168.15.2` `CSeq: 1 REGISTER`
Empty line	

Register: lifetime of the registration

- **Either use**
  - ***expires*** parameter

    Contact:<sip:barbara@192.168.15.2>;expires=2700

    - In seconds
    - only concerns that contact
  - ***Expires*** header

    Contact:<sip:barbara@[2001:660:200::1]>
    Contact:<sip:barbara@192.168.15.2>
    Expires:45

    - In minutes
    - Concerns all contacts that do not have an *expires* parameter
  - **No indication**
    - Default is 1 hour

REGISTER: refresh, cancel, query

- **It is up to the user agent to refresh registrations of Contact addresses. In order to do so, a UA has to resend its initial REGISTER request.**
- **In order to cancel a Contact registration, a user agent has to set its "Expires" time to zero**

```
To: sip:barbara@b.com
Contact: <sip:barbara@192.168.15.2>
Expires: 0
```

- **In order to cancel all contact address of records, a UA could use an asterisk (*)**

```
To: sip:barbara@b.com
Contact: *
Expires: 0
```

- **Omitting the `Contact` header would not modify any AOR and the corresponding response would contain all existing registrations.**

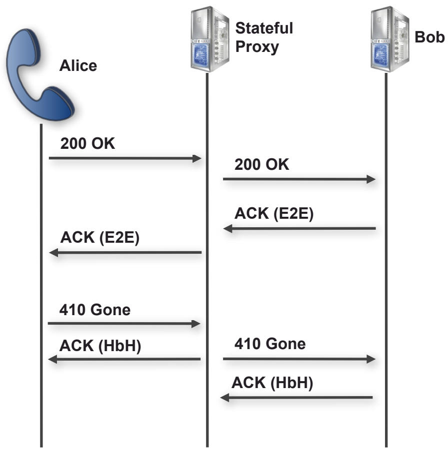
Register: multiple contacts registration

- **It is possible to associate more than one device URI to an AoR**
  - Use multiple `contact` headers
  - Use the parameter `q` to give preferences on the contact addresses
    - q varies between 0 and 1, the highest the preferred

```
Contact:<sip:barbara@[2001:660:200::1]>;q=0.4
Contact:<sip:barbara@192.168.15.2>;q=0.1
Expires:45
```

Methods

- **ACK - Acknowledge final responses to INVITE requests**
  - Final responses are 2xx, 3xx, 4xx, 5xx, 6xx
  - An Ack is end-to-end for 2xx responses, otherwise hop-by-hop (for stateful proxies)
  - CSeq is not incremented (same as the request), but the CSeq request method is Ack
  - Branch ID
    - Hop-by-hop ACK reuses the same branch ID as the INVITE
    - End-to-end ACK uses a different branch ID



Ack method

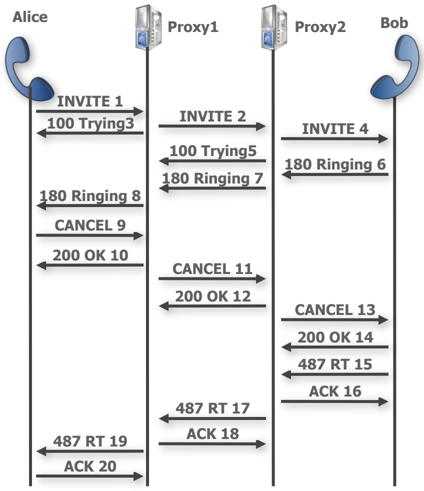
```
INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@atlanta.example.com
CSeq: 1 INVITE
Contact: <sip:alice@client.atlanta.example.com>
Route: <sip:alg1.atlanta.example.com;lr>
Content-Type: application/sdp
Content-Length: 151
```

...

```
ACK sip:bob@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060;branch=z9hG4bK74bhh
Max-Forwards: 70
Route: <sip:alg1.atlanta.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@atlanta.example.com
CSeq: 1 ACK
Content-Length: 0
```

Methods

- **CANCEL - terminate pending searches or call attempt**
  - Can be generated by user agents or proxy (provided that a 1xx response was received, and no final response)
  - Hop-by-hop request - receive a response by the next stateful element
  - Cseq and Branch ID are the same as the INVITE
  - A proxy receiving a CANCEL forwards the CANCEL and generates a response (200 OK). The INVITE is answered with a 487 Request Terminated



CANCEL method

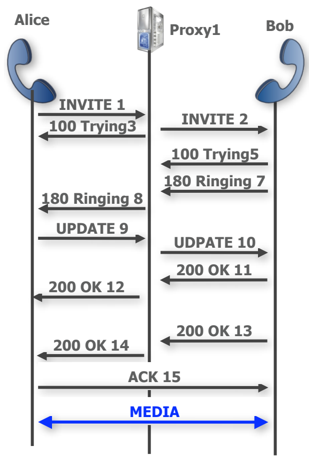
```
INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Route: <sip:ss1.atlanta.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@atlanta.example.com
CSeq: 1 INVITE
Contact: <sip:alice@client.atlanta.example.com>
Content-Type: application/sdp
Content-Length: 151
```

...

```
CANCEL sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Route: <sip:ss1.atlanta.example.com;lr>
Call-ID: 2xTb9vxSit55XU7p8@atlanta.example.com
CSeq: 1 CANCEL
Content-Length: 0
```

Methods

- **UPDATE - modify the state of a session without changing the state of the dialog**
  - Used instead of a re-INVITE in a pending session



Methods

- **OPTIONS - query a user agent or server about its capabilities and discover its current availability**
  - Only generated by server or user agent
  - Responses are 4xx, 6xx for negative answers and 2xx for positive answers with
    - Allow: specify the requests it accepts
    - Accept: type of accepted Internet media types (e.g., Application/SDP)
    - Accept-Encoding: used to specify acceptable message body encoding schemes (e.g., Accept-Encoding: text/plain)
    - Accept-Language: preferences for language (such as the one used for reason phrase, or subject)

OPTIONS: request and response

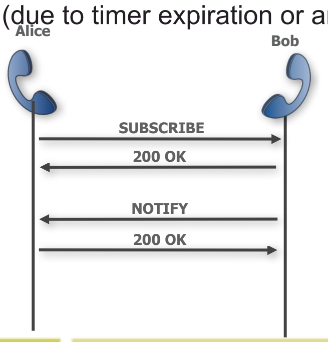
```
OPTIONS sip:norton@savons.com SIP/2.0
via: SIP/2.0/UDP client1.telecom-bretagne.eu;branch=z9hG4bK1834
Max-Forwards: 70
To: <sip:edouard.norton@savons.com>
From: B. Pitt <sip:brad.pitt@savons.com>;tag=34
Call-ID: 9352812@client1.telecom-bretagne.eu
CSeq : 1 OPTIONS
Content-Length: 0


SIP/2.0 200 OK
via: SIP/2.0/UDP client1.telecom-bretagne.eu;branch=z9hG4bK1834;
received=192.168.0.2
Max-Forwards: 70
To: <sip:norton@savons.com>;tag=68
From: B. Pitt <sip:brad.pitt@savons.com>;tag=34
Call-ID: 9352812@client1.telecom-bretagne.eu
CSeq : 1 OPTIONS
Allow: INVITE, OPTIONS, ACK, BYE, CANCEL, REFER
Accept-Language: en, de, fr
Content-Length: ...
Content-Type: application/sdp

v=0
...
```

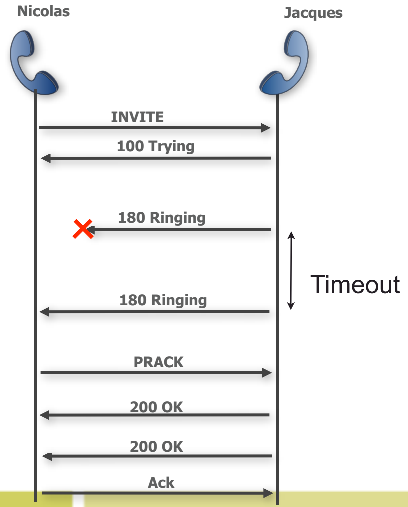

Methods

- **SUBSCRIBE - establish a subscription for the purpose of receiving notifications about a particular event**
  - Request the state and state updates from a remote node
  - Establish a dialog during the time indicated in the `Expires` header. A server accepting the request responds with a 200 OK
  - Identification of events is provided by three pieces of information:
    - Request URI
    - Event Type (*Event* header)
    - and (optionally) message body.
- **NOTIFY - Convey information about the occurrence of a particular event**
  - Always sent within a dialog
  - Is also used to notify the unsubscribe (due to timer expiration or an explicit unsuscribe (with `Expires: 0`)

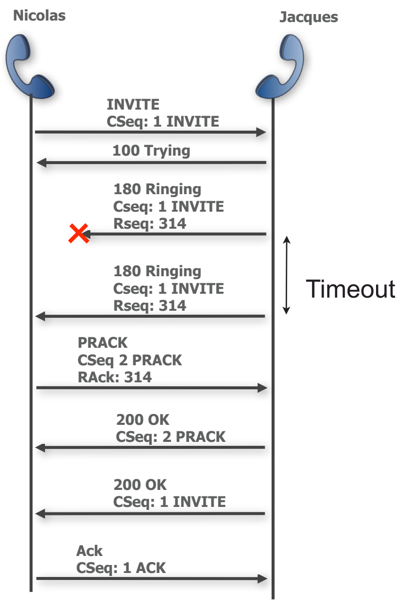


Methods

- PRACK (RFC 3262)
  - Used to acknowledge receipt of reliability transported provisional responses (1xx - exept the 100 which is never reliably transported)
  - Reliably transported provisional responses contain a *RSeq* header (with a sequence number) and a *Supported: 100rel* header
  - A timer on the UAS triggers the retransmission of the provisional response
  - A `RAck` header field is used within a response to a PRACK request to reliably acknowledge a provisional response that contained a `RSeq` header field. The RAck echoes the `CSeq` and `RSeq` from the provisional response

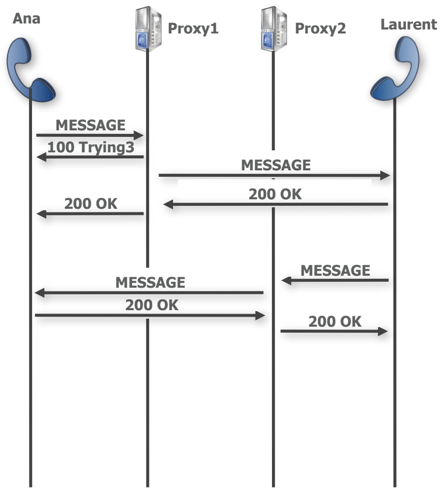


PRACK - usage of CSeq and RSeq



Methods

- **MESSAGE - Transport instant message using SIP**
  - Can be exchanged within a dialog or outside a dialog
  - Acknowledged by a 200 OK message



Methods

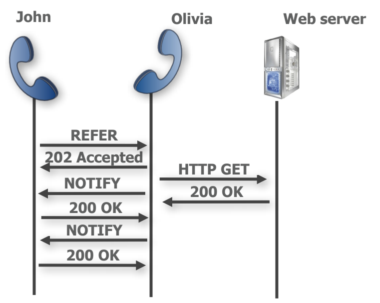
- **INFO - Send call signaling information to another user agent with which it has an established media session**
  - Different from a re-INVITE as it does not change the characteristics of the call
  - Can be used to transport midcall signaling information

```
INFO sip:poynting@mason.edu.uk SIP/2.0
Via: SIP/2.0/UDP cavendish.kings/cambridge.edu.uk;
banch=z9hG4bK24555
Max-Forwards: 70
To: sip:poynting@mason.edu.uk SIP/2.0; tag=12390
From: sip:quelqun@kings.cambridge.edu.uk; tag=5289
Call-ID: 18379@cavendish.kings.cambridge.edu.uk
CSeq: 6 INFO
Content-Type message/ISUP
Content-Length: 16

51a6324134527
```

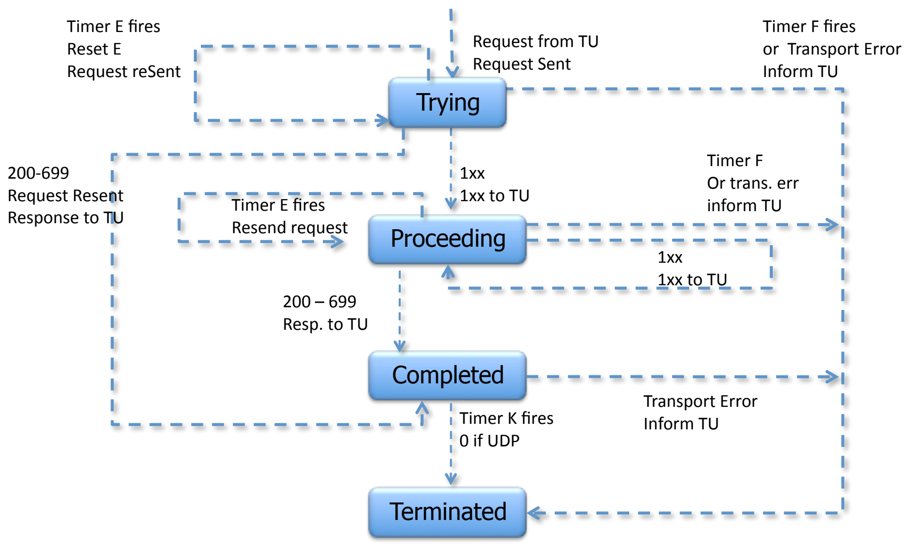
Methods

- **REFER - used by a user agent to request another user agent to access a URI or URL ressource**
  - May be used to a call transfer service



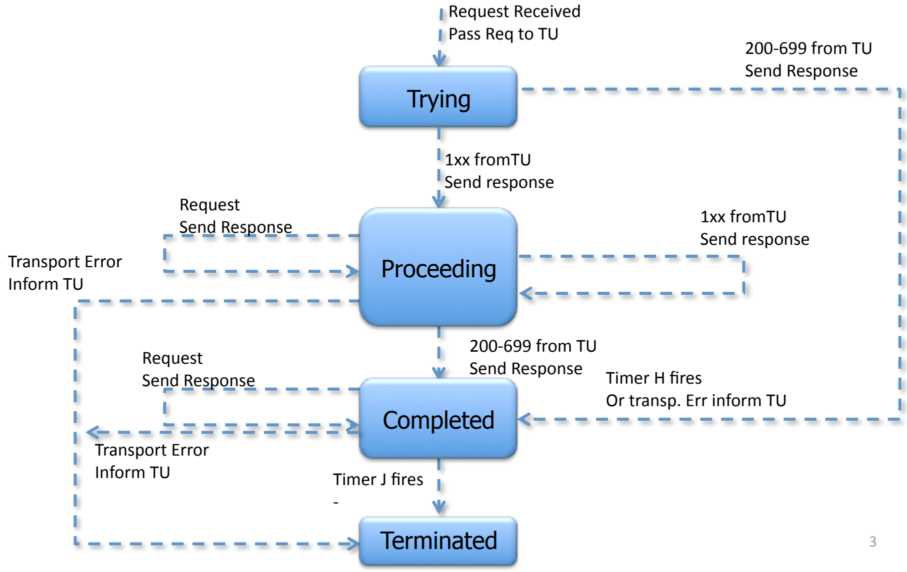
TELECOM Bretagne

Non-INVITE client transactions



TELECOM Bretagne

Non-INVITE server transaction



SIP models

- **Client / Server**
  - Determined by the initiator of the message
    - Request sender: client
    - Request receiver: server
- **Transaction : Messages exchange from a request to the final response (between 200 and 699)**
- **Dialog: SIP relation between two user agents that last for a while.**
  - Can be understood as a session
  - Created by a reponse to an INVITE (like a *200 Ok*) or to SUBSCRIBE
  - Ease messages sequencing
  - Allow routing SIP messaging that belong to the same dialog

Transaction and Dialog

Transaction
- Identified by the *branch* field of the *via* header
- Determined and unique on each traversed proxy / UA

```
REGISTER sip:registrar.biloxi.com SIP/2.0
Via: SIP/2.0/UDP bobspc.biloxi.com:5060;branch=z9hG4bKnashds7
```

Dialog
- Identified by the *call-id*, the *to tag* and the *from tag*

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP bobspc.biloxi.com:5060;branch=z9hG4bKnashds7;received=192.0.2.4
To: Bob <sip:bob@biloxi.com>;tag=2493k59kd
From: Bob <sip:bob@biloxi.com>;tag=456248
Call-ID: 843817637684230@998sdasdh09
```


Headers

Which headers are needed?

Header field	where	proxy	ACK	BYE	CAN	INV	OPT	REG
Accept	R		-	o	-	o	m*	o
Accept	2xx		-	-	-	o	m*	o
Accept	415		-	c	-	c	c	c
Accept-Encoding	R		-	o	-	o	o	o
Accept-Encoding	2xx		-	-	-	o	m*	o
Accept-Encoding	415		-	c	-	c	c	c
Accept-Language	R		-	o	-	o	o	o
Accept-Language	2xx		-	-	-	o	m*	o
Accept-Language	415		-	c	-	c	c	c
Alert-Info	R	ar	-	-	-	o	-	-
Alert-Info	180	ar	-	-	-	o	-	-
Allow	R		-	o	-	o	o	o
Allow	2xx		-	o	-	m*	m*	o
Allow	r		-	o	-	o	o	o
Allow	405		-	m	-	m	m	m
Authentication-Info	2xx		-	o	-	o	o	o
Authorization	R		o	o	o	o	o	o
Call-ID	c	r	m	m	m	m	m	m
Call-Info		ar	-	-	-	o	o	o
Contact	R		o	-	-	m	o	o
Contact	1xx		-	-	-	o	-	-
Contact	2xx		-	-	-	m	o	o
Contact	3xx	d	-	o	-	o	o	o
Contact	485		-	o	-	o	o	o
Content-Disposition			o	o	-	o	o	o
Content-Encoding			o	o	-	o	o	o
Content-Language			o	o	-	o	o	o
Content-Length		ar	t	t	t	t	t	t
Content-Type			*	*	-	*	*	*
CSeq	c	r	m	m	m	m	m	m
Date		a	o	o	o	o	o	o
Error-Info	300-699	a	-	o	o	o	o	o
Expires			-	-	-	o	-	o
From	c	r	m	m	m	m	m	m
In-Reply-To	R		-	-	-	o	-	-
Max-Forwards	R	amr	m	m	m	m	m	m
Min-Expires	423		-	-	-	-	-	m
MIME-Version			o	o	-	o	o	o
Organization		ar	-	-	-	o	o	o

- **The "where" column describes the request and response types in which the header field can be used. Values in this column are:**
  - R: header field may only appear in requests;
  - r: header field may only appear in responses;
  - 2xx, 4xx, etc.: response codes with which the header field can be used;
  - c: header field is copied from the request to the response.
  - Empty entry: may be present in all requests and responses.
- **The "proxy" column describes the operations a proxy may perform on a header field:**
  - a: A proxy can add or concatenate the header field if not present.
  - m: A proxy can modify an existing header field value.
  - d: A proxy can delete a header field value.
  - r: A proxy must be able to read the header field, and thus this header field cannot be encrypted.
- **The next six columns relate to the presence of a header field in a method:**
  - c: Conditional; requirements on the header field depend on the context of the message.
  - **m: The header field is mandatory.**
  - *: The header field is required if the message body is not empty.
  - -: The header field is not applicable.
  - **m*: The header field SHOULD be sent, but clients/servers need to be prepared to receive messages without that header field.**
  - o: The header field is optional.
  - t: The header field SHOULD be sent, but clients/servers need to be prepared to receive messages without that header field.

CSeq header

- **Required field in all requests**
- **Contain a decimal number increased for each new request**
  - Exept for CANCEL and ACK which use the CSeq from the INVITE they refer to
- **Differentiate a retransmission from a new request**

```
CSeq 34 INVITE
```

Routing and headers

- ***From***, ***Contact***, ***Record-route/Route*** **and** ***Via*** **headers determine how requests and responses are routed in a network of SIP proxy servers.**
- **From**
  - Used for subsequent requests if there is no Contact or Record-Route header.
- **Contact**
  - Determines the destination placed in the Request-URI for subsequent requests and can be used to bypass proxies not enumerated in a Record-Route header. Also used in responses by redirect servers and in REGISTER requests and responses.
- **Record-Route/Route:**
  - The Record-Route header is inserted into requests by proxies that want to be in the path of subsequent requests for the same call-id. It is then used by the user agent to route subsequent requests.
- **Via**
  - Via headers are inserted by servers into requests to detect loops and to allow responses to find their way back to the client. They have no influence on the routing of future requests (or responses).

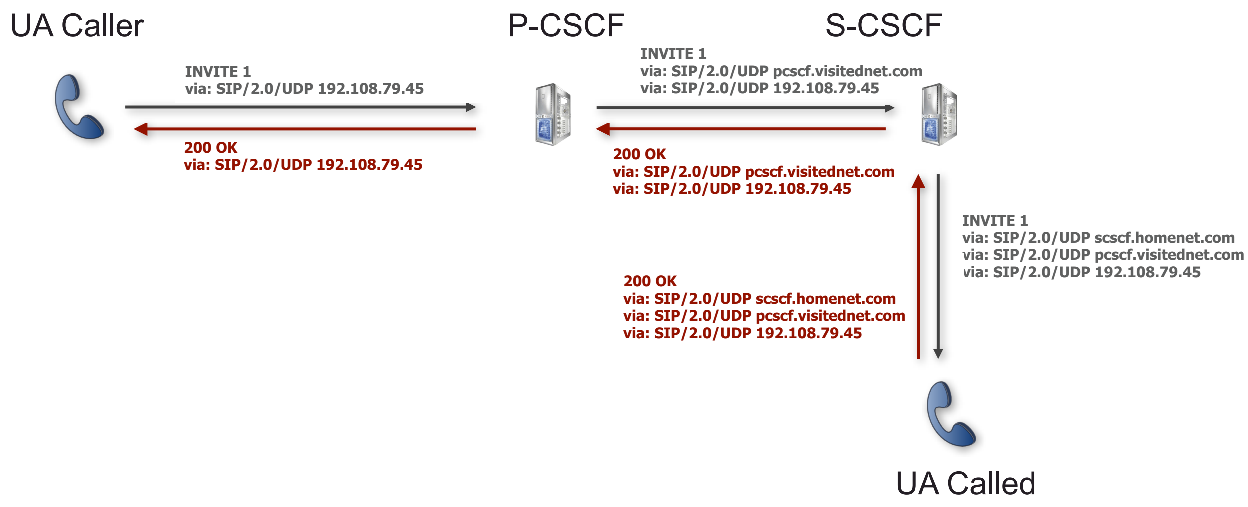
➡ **Generally, in short, requests should be sent to Route if present, Contact if there is no Route, From if there is no Contact.**

Routes for the message

- **Request are routed via URL**
- **Responses traces back request route without proxy server state**
- **Forward to host,port in next *via***

```
via: SIP/20.0/UDP proxy.sip.org:5060; received=130.79.3.126
```

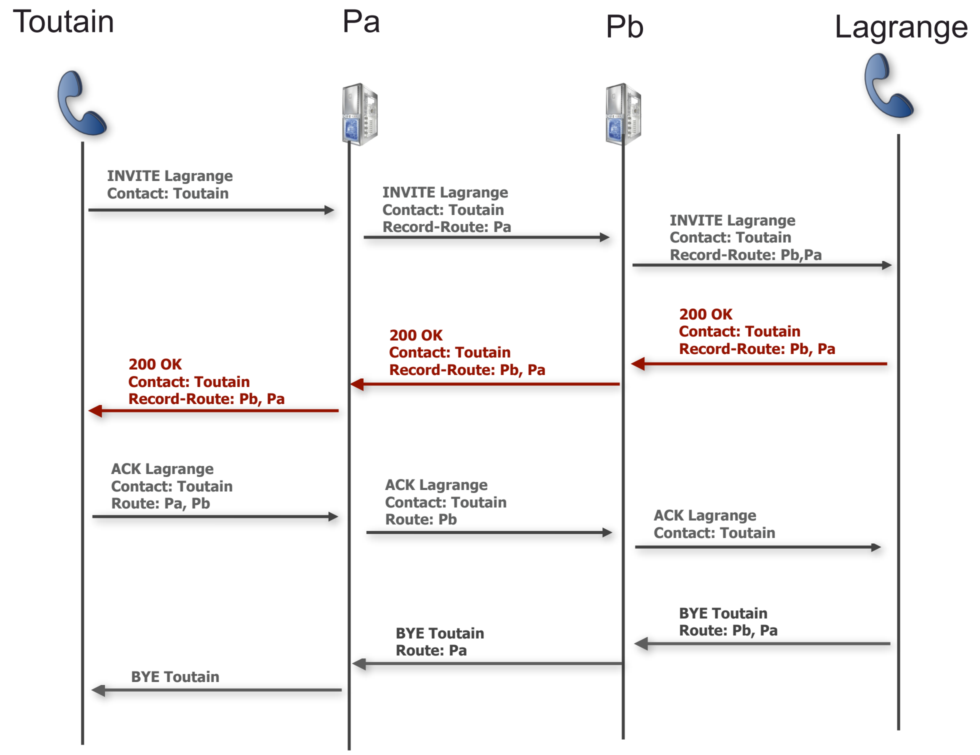
SIP response routing



Forcing request path

- **Usually by-pass proxy in subsequent requests**
- **A proxy may want to stay in the path**
  - Anonymizer poxies
  - Firewalls
  - proxies controlling PSTN gateways
- **Use of *record-route* and *route* headers**

Force request path



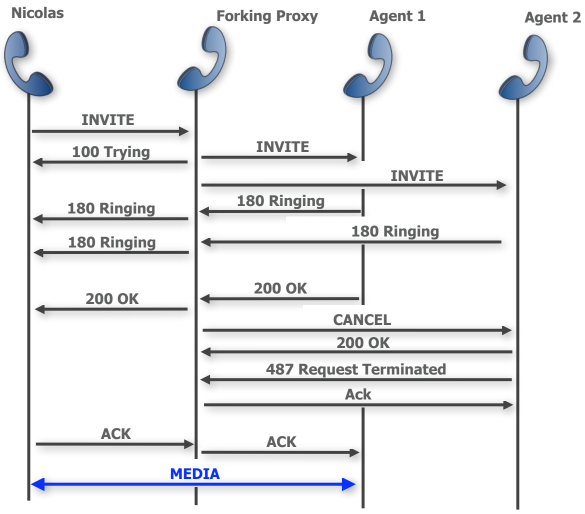
TELECOM Bretagne

Scenarios

SIP request forking

- **A proxy may fork a request**
  - ➡ *if the location service returns multiple possible locations for the called party*
  - Parallel
  - Sequential
  - Combination of both

Forking Proxy Server



Exercice: UAC parallel search

- **The caller receives multiple locations for the called party from a redirect server. Instead of trying the locations one at a time, the user agent implements a parallel search for the called party by simultaneously sending the INVITE to three different locations.**
  - The 1st location indicates that the user has not been found
  - The 2nd and 3rd locations send a Ringing
  - The 2nd location picks up
- **Question**
  - Write the response sent from the Redirect server
  - Write the three INVITE requests sent to the three locations (in particular indicate the request line, To, Call-ID, CSeq headers and the branch parameters for each message
  - How the three INVITE are terminated?
  - Draw the call diagramm for this scenario

TELECOM Bretagne

Error Codes

Note:
Many codes are same as HTTP
SIP specific codes start x80

Informational

100	Trying
180	Ringing
181	Call Is Being Forwarded
182	Queued
183	Session Progress

Success

200	OK
202	Accepted

Redirection

300	Multiple Choices
301	Moved Permanently
302	Moved Temporarily
303	See Other
305	Use Proxy
380	Alternative Service

Client error

400	Bad Request
401	Unauthorized
402	Payment Required
403	Forbidden
404	Not Found
405	Method Not Allowed
406	Not Acceptable
407	Proxy Authentication Required
408	Request Timeout
409	Conflict
410	Gone
411	Length Required
413	Request Entity Too Large
414	Request-URI Too Large
415	Unsupported Media Type
420	Bad Extension
480	Temporarily not available
481	Call Leg/Transaction Does Not Exist
482	Loop Detected
483	Too Many Hops
484	Address Incomplete
485	Ambiguous
486	Busy Here
487	Request Cancelled
488	Not Acceptable Here

Server error

500	Internal Server Error
501	Not Implemented
502	Bad Gateway
503	Service Unavailable
504	Gateway Time-out
505	SIP Version not supported

Global failure

600	Busy Everywhere
603	Decline
604	Does not exist anywhere
606	Not Acceptable

TELECOM Bretagne

Response from the redirect server

```
SIP/2.0 300 Multiple locations
sip:poynting@mason.edu.uk SIP/2.0
Via: SIP/2.0/UDP 1.2.3.4:6000; banch=z9hG4bK3
To: sip:alice@wonderland.com; tag=12390
From: sip:nicolas@telecom-bretagne.eu; tag=4
Call-ID: 100@1.2.3.4
CSeq: 1 INVITE
Contact: sip:alice@location1.fr
Contact: sip:+33.299436589@sip-phone.org
Contact: sip:alice@location3.fr
```


The three invite sent to the three locations

First INVITE

```
INVITE sip:alice@location1.fr SIP/2.0
Via: SIP/2.0/UDP 1.2.3.4;branch=z9hG4bK3
From: sip:nicolas@telecom-bretagne.eu;tag=5
To: sip:alice@wonderland.com
Contact: <sip:nicolas@1.2.3.4>
Call-ID: 100@1.2.3.4
Cseq: 2 INVITE
```

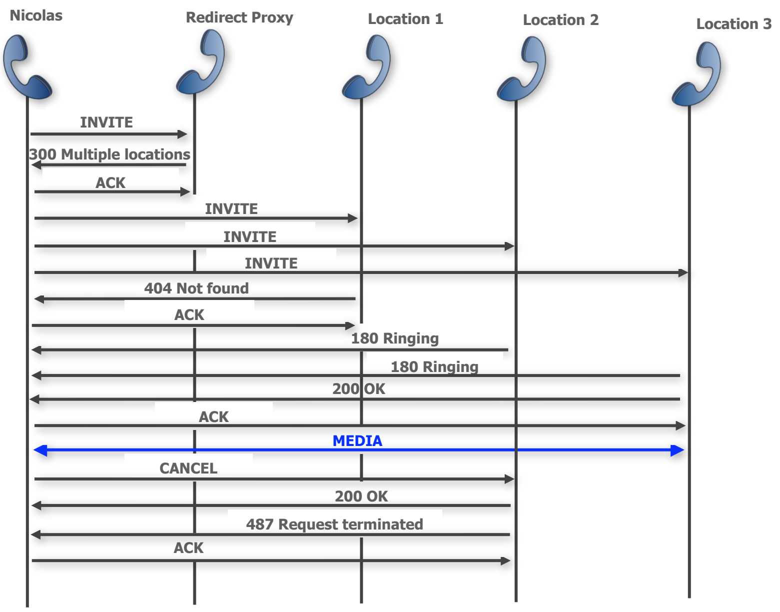
Second INVITE

```
INVITE sip: +33.299436589@sip-phone.org SIP/2.0
Via: SIP/2.0/UDP 1.2.3.4;branch=z9hG4bK4
From: sip:nicolas@telecom-bretagne.eu;tag=5
To: sip:alice@wonderland.com
Contact: <sip:nicolas@1.2.3.4>
Call-ID: 100@1.2.3.4
Cseq: 2 INVITE
```

Third INVITE

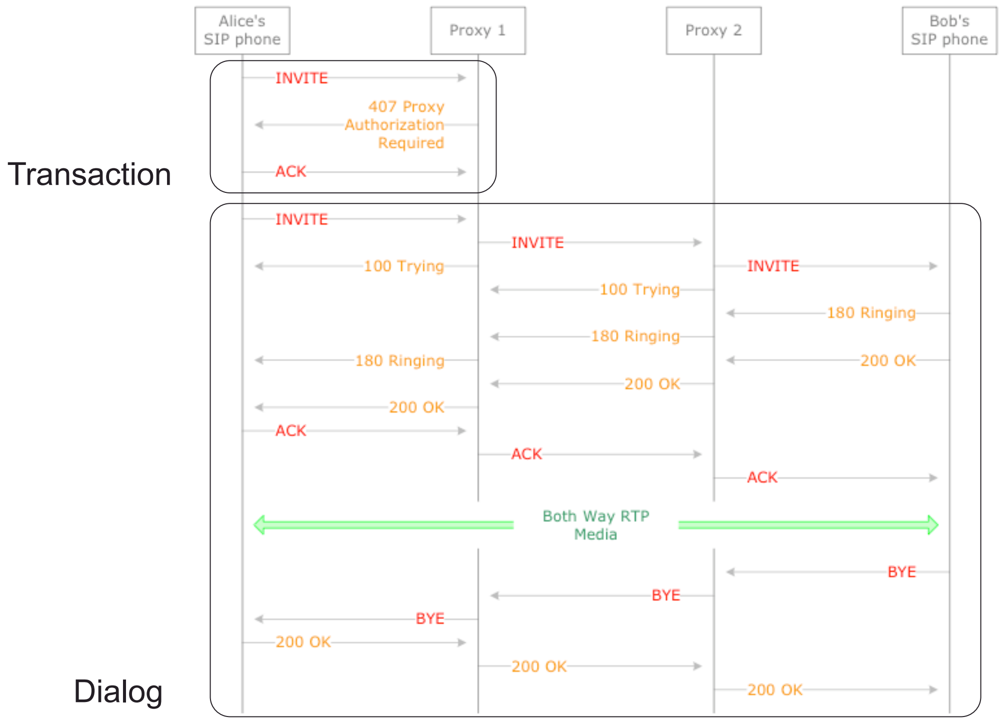
```
INVITE sip:alice@location1.fr SIP/2.0
Via: SIP/2.0/UDP 1.2.3.4;branch=z9hG4bK5
From: sip:nicolas@telecom-bretagne.eu;tag=5
To: sip:alice@wonderland.com
Contact: <sip:nicolas@1.2.3.4>
Call-ID: 100@1.2.3.4
Cseq: 2 INVITE
```

UAC forking



TELECOM Bretagne

SIP call flows - session establishment through two proxies (RFC 3665)



TELECOM Bretagne


```
INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060
;branch=z9hG4bk74b43
Max-Forwards: 70
Route: <sip:ss1.altanta.example.com;lr>
From: Alice <sip:alice@a.com;tag=9fxced76sl>
To: Bob <sip:bob@biloxi.example.com>
Contact: <sip:alice@10.43.122.3>
Call-ID: 3838276298220188511@atlanta.example.com
Cseq: 1 INVITE
```

Proxy 1

F3

```
ACK sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/TCP client.atlanta.example.com:5060
 ;branch=z9hG4bK74b43
Max-Forwards: 70
From: Alice <sip:alice@atlanta.example.com>
 ;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
 ;tag=3flal12sf
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 1 ACK
Content-Length: 0
```

```
INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060
;branch=z9hG4bk74b43
Max-Forwards: 70
Route: <sip:ss1.altanta.example.com;lr>
From: Alice <sip:alice@a.com;tag=9fxced76sl>
To: Bob <sip:bob@biloxi.example.com>
Contact: <sip:alice@10.43.122.3>
Call-ID: 3838276298220188511@atlanta.example.com
Cseq: 1 INVITE
```

Alice's SIP phone

Proxy 1

F1

```
INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/TCP client.atlanta.example.com:5060
 ;branch=z9hG4bK74b43
Max-Forwards: 70
Route: <sip:ss1.atlanta.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>
 ;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 1 INVITE
Contact: <sip:alice@client.atlanta.example.com
 ;transport=tcp>
Content-Type: application/sdp
Content-Length: 151

v=0
o=alice 2890844526 2890844526 IN IP4 client.atlanta.example.com
s=-
c=IN IP4 192.0.2.101
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

Alice's SIP phone

Proxy 1

F4

```
INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/TCP client.atlanta.example.com:5060
 ;branch=z9hG4bK74bf9
Max-Forwards: 70
Route: <sip:ss1.atlanta.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>
 ;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 2 INVITE
Contact: <sip:alice@client.atlanta.example.com
 ;transport=tcp>
Proxy-Authorization: Digest username="alice",
 realm="atlanta.example.com",
 nonce="wf84f1ceczx41ae6cbe5aea9c8e88d359", opaque="",
 uri="sip:bob@biloxi.example.com",
 response="42ce3cef44b22f50c6a6071bc8"
Content-Type: application/sdp
Content-Length: 151

v=0
o=alice 2890844526 2890844526 IN IP4 client.atlanta.example.com
s=-
c=IN IP4 192.0.2.101
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

Alice's SIP phone

Proxy 1

F5

INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/TCP ss1.atlanta.example.com:5060
;branch=z9hG4bK2d4790.1
Via: SIP/2.0/TCP client.atlanta.example.com:5060
;branch=z9hG4bK74bf9
;received=192.0.2.101
Max-Forwards: 69
Record-Route: <sip:ss1.atlanta.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>
;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 2 INVITE
Contact: <sip:alice@client.atlanta.example.com
;transport=tcp>
Content-Type: application/sdp
Content-Length: 151

v=0
o=alice 2890844526 2890844526 IN IP4 client.atlanta.example.com
s=-
c=IN IP4 192.0.2.101
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

Proxy 2

F6

SIP/2.0 100 Trying
Via: SIP/2.0/TCP client.atlanta.example.com:5060
;branch=z9hG4bK74bf9
;received=192.0.2.101
From: Alice <sip:alice@atlanta.example.com>
;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 2 INVITE
Content-Length: 0

Alice's SIP phone

```
INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060
;branch=z9hG4bk74bf9
Max-Forwards: 70
Route: <sip:ss1.altanta.example.com;lr>
From: Alice <sip:alice@a.com;tag=9fxced76sl>
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
Cseq: 2 INVITE
Contact: <sip:alice@client.atlanta.example.com ;transport:tcp>
Proxy-Authorization: Digest username="alice",
realm="atkanta.example.com",
nonce="wf84f1ceczx41ae6cbe5aea9c8e88d359", opaque="",
uri="sip:bob@biloxi.example.com",
response="42ce3cef44b22f50c6a6071bc8"
Content-Type: application/sdp
Content-Length: 151
```


Proxy 1

F8

SIP/2.0 100 Trying
Via: SIP/2.0/TCP ss1.atlanta.example.com:5060
;branch=z9hG4bK2d4790.1
;received=192.0.2.111
Via: SIP/2.0/TCP client.atlanta.example.com:5060
;branch=z9hG4bK74bf9
;received=192.0.2.101
From: Alice <sip:alice@atlanta.example.com>
;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 2 INVITE
Content-Length: 0

Proxy 2

F7

INVITE sip:bob@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/TCP ss2.biloxi.example.com:5060
;branch=z9hG4bK721e4.1
Via: SIP/2.0/TCP ss1.atlanta.example.com:5060
;branch=z9hG4bK2d4790.1
;received=192.0.2.111
Via: SIP/2.0/TCP client.atlanta.example.com:5060
;branch=z9hG4bK74bf9
;received=192.0.2.101
Max-Forwards: 68
Record-Route: <sip:ss2.biloxi.example.com;lr>,
<sip:ss1.atlanta.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>
;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 2 INVITE
Contact: <sip:alice@client.atlanta.example.com
;transport=tcp>
Content-Type: application/sdp
Content-Length: 151

v=0
o=alice 2890844526 2890844526 IN IP4 client.atlanta.example.com
s=-
c=IN IP4 192.0.2.101
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

```
INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060
;branch=z9hG4bk74bf9
Max-Forwards: 70
Route: <sip:ss1.altanta.example.com;lr>
From: Alice <sip:alice@a.com;tag=9fxced76sl>
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
Cseq: 2 INVITE
Contact: <sip:alice@client.atlanta.example.com ;transport:tcp>
Proxy-Authorization: Digest username="alice",
realm="atkanta.example.com",
nonce="wf84f1ceczx41ae6cbe5aea9c8e88d359", opaque="",
uri="sip:bob@biloxi.example.com",
response="42ce3cef44b22f50c6a6071bc8"
Content-Type: application/sdp
Content-Length: 151
```

Bob's SIP phone

Proxy 1

Proxy 2

F10

```
SIP/2.0 180 Ringing
Via: SIP/2.0/TCP ss1.atlanta.example.com:5060
 ;branch=z9hG4bK2d4790.1
 ;received=192.0.2.111
Via: SIP/2.0/TCP client.atlanta.example.com:5060
 ;branch=z9hG4bK74bf9
 ;received=192.0.2.101
Record-Route: <sip:ss2.biloxi.example.com;lr>,
 <sip:ss1.atlanta.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>
 ;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
 ;tag=314159
Call-ID: 3848276298220188511@atlanta.example.com
Contact: <sip:bob@client.biloxi.example.com;transport=tcp>
CSeq: 2 INVITE
Content-Length: 0
```

F9

```
SIP/2.0 180 Ringing
Via: SIP/2.0/TCP ss2.biloxi.example.com:5060
 ;branch=z9hG4bK721e4.1
 ;received=192.0.2.222
Via: SIP/2.0/TCP ss1.atlanta.example.com:5060
 ;branch=z9hG4bK2d4790.1
```

F11

```
SIP/2.0 180 Ringing
Via: SIP/2.0/TCP client.atlanta.example.com:5
 ;branch=z9hG4bK74bf9
 ;received=192.0.2.101
Record-Route: <sip:ss2.biloxi.example.com;lr
 <sip:ss1.atlanta.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>
 ;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
 ;tag=314159
Call-ID: 3848276298220188511@atlanta.exan
Contact: <sip:bob@client.biloxi.example.com;
CSeq: 2 INVITE
Content-Length: 0
```

Alice's SIP phone

```
INVITE sip:bob@biloxi.example.com SIP/2.0

Via: SIP/2.0/UDP client.atlanta.example.com:5060
;branch=z9hG4bk74bf9
Max-Forwards: 70
Route: <sip:ss1.altanta.example.com;lr>
From: Alice <sip:alice@a.com;tag=9fxced76sl>
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
Cseq: 2 INVITE
Contact: <sip:alice@client.atlanta.example.com ;transport:tcp>
Proxy-Authorization: Digest username="alice",
realm="atkanta.example.com",
nonce="wf84f1ceczx41ae6cbe5aea9c8e88d359", opaque="",
uri="sip:bob@biloxi.example.com",
response="42ce3cef44b22f50c6a6071bc8"
Content-Type: application/sdp
Content-Length: 151
```


Proxy 2

```
INVITE sip:bob@biloxi.example.com SIP/2.0

Via: SIP/2.0/UDP client.atlanta.example.com:5060
;branch=z9hG4bk74bf9
Max-Forwards: 70
Route: <sip:ss1.altanta.example.com;lr>
From: Alice <sip:alice@a.com;tag=9fxced76sl>
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
Cseq: 2 INVITE
Contact: <sip:alice@client.atlanta.example.com ;transport:tcp>
Proxy-Authorization: Digest username="alice",
realm="atkanta.example.com",
nonce="wf84f1ceczx41ae6cbe5aea9c8e88d359", opaque="",
uri="sip:bob@biloxi.example.com",
response="42ce3cef44b22f50c6a6071bc8"
Content-Type: application/sdp
Content-Length: 151
```

F12

```
SIP/2.0 200 OK
Via: SIP/2.0/TCP ss2.biloxi.example.com:5060
 ;branch=z9hG4bK721e4.1
 ;received=192.0.2.222
Via: SIP/2.0/TCP ss1.atlanta.example.com:5060
 ;branch=z9hG4bK2d4790.1
 ;received=192.0.2.111
Via: SIP/2.0/TCP client.atlanta.example.com:5060
 ;branch=z9hG4bK74bf9
 ;received=192.0.2.101
Record-Route: <sip:ss2.biloxi.example.com;lr>,
 <sip:ss1.atlanta.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>
 ;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
 ;tag=314159
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 2 INVITE
Contact: <sip:bob@client.biloxi.example.com;transport=tcp>
 Content-Type: application/sdp
 Content-Length: 147

v=0
o=bob 2890844527 2890844527 IN IP4 client.biloxi.example.com
s=-
c=IN IP4 192.0.2.201
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

Bob's SIP phone

Proxy 1

Proxy 2

F13

```
SIP/2.0 200 OK
Via: SIP/2.0/TCP ss1.atlanta.example.com:5060
 ;branch=z9hG4bK2d4790.1
 ;received=192.0.2.111
Via: SIP/2.0/TCP client.atlanta.example.com:5060
 ;branch=z9hG4bK74bf9
 ;received=192.0.2.101
Record-Route: <sip:ss2.biloxi.example.com;lr>,
 <sip:ss1.atlanta.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 2 INVITE
Contact: <sip:bob@client.biloxi.example.com;transport=tcp>
Content-Type: application/sdp
Content-Length: 147

v=0
o=bob 2890844527 2890844527 IN IP4 client.biloxi.example.com
s=-
c=IN IP4 192.0.2.201
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```


Proxy 1

F14

```
SIP/2.0 200 OK
Via: SIP/2.0/TCP client.atlanta.example.com:5060
 ;branch=z9hG4bK74bf9;received=192.0.2.101
Record-Route: <sip:ss2.biloxi.example.com;lr>,
 <sip:ss1.atlanta.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 2 INVITE
Contact: <sip:bob@client.biloxi.example.com;transport=tcp>
Content-Type: application/sdp
Content-Length: 147

v=0
o=bob 2890844527 2890844527 IN IP4 client.biloxi.example.com
s=-
c=IN IP4 192.0.2.201
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

Alice's SIP phone

```
INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060
;branch=z9hG4bk74bf9
Max-Forwards: 70
Route: <sip:ss1.altanta.example.com;lr>
From: Alice <sip:alice@a.com;tag=9fxced76sl>
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
Cseq: 2 INVITE
Contact: <sip:alice@client.atlanta.example.com ;transport:tcp>
Proxy-Authorization: Digest username="alice",
realm="atkanta.example.com",
nonce="wf84f1ceczx41ae6cbe5aea9c8e88d359", opaque="",
uri="sip:bob@biloxi.example.com",
response="42ce3cef44b22f50c6a6071bc8"
Content-Type: application/sdp
Content-Length: 151
```

Proxy 1

F15

```
ACK sip:bob@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/TCP client.atlanta.example.com:5060
 ;branch=z9hG4bK74b76
Max-Forwards: 70
Route: <sip:ss1.atlanta.example.com;lr>,
 <sip:ss2.biloxi.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 2 ACK
Content-Length: 0
```

Alice's SIP phone

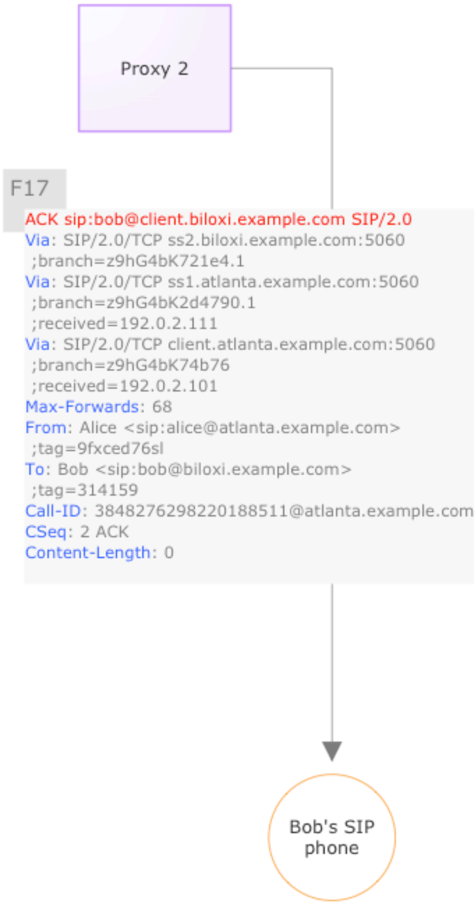
SIP call flows - session establishment through two proxies (RFC 3665)

Proxy 1

F16

```
ACK sip:bob@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/TCP ss1.atlanta.example.com:5060
 ;branch=z9hG4bK2d4790.1
Via: SIP/2.0/TCP client.atlanta.example.com:5060
 ;branch=z9hG4bK74b76
 ;received=192.0.2.101
Max-Forwards: 69
Route: <sip:ss2.biloxi.example.com;lr>
From: Alice <sip:alice@atlanta.example.com>
 ;tag=9fxced76sl
To: Bob <sip:bob@biloxi.example.com>
 ;tag=314159
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 2 ACK
Content-Length: 0
```

Proxy 2




```
INVITE sip:bob@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.atlanta.example.com:5060
;branch=z9hG4bk74bf9
Max-Forwards: 70
Route: <sip:ss1.altanta.example.com;lr>
From: Alice <sip:alice@a.com;tag=9fxced76sl>
To: Bob <sip:bob@biloxi.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
Cseq: 2 INVITE
Contact: <sip:alice@client.atlanta.example.com ;transport:tcp>
Proxy-Authorization: Digest username="alice",
realm="atkanta.example.com",
nonce="wf84f1ceczx41ae6cbe5aea9c8e88d359", opaque="",
uri="sip:bob@biloxi.example.com",
response="42ce3cef44b22f50c6a6071bc8"
Content-Type: application/sdp
Content-Length: 151
```

SIP call flows - session establishment through two proxies (RFC 3665)

Proxy 1

Proxy 2

F19

```
BYE sip:alice@client.atlanta.example.com SIP/2.0
Via: SIP/2.0/TCP ss2.biloxi.example.com:5060
 ;branch=z9hG4bK721e4.1
Via: SIP/2.0/TCP client.biloxi.example.com:5060
 ;branch=z9hG4bKnashds7
 ;received=192.0.2.201
Max-Forwards: 69
Route: <sip:ss1.atlanta.example.com;lr>
From: Bob <sip:bob@biloxi.example.com>
 ;tag=314159
To: Alice <sip:alice@atlanta.example.com>
 ;tag=9fxced76sl
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 1 BYE
Content-Length: 0
```


TELECOM Bretagne

Proxy 1

F20

```
BYE sip:alice@client.atlanta.example.com SIP/2.0
Via: SIP/2.0/TCP ss1.atlanta.example.com:5060
 ;branch=z9hG4bK2d4790.1
Via: SIP/2.0/TCP ss2.biloxi.example.com:5060
 ;branch=z9hG4bK721e4.1
 ;received=192.0.2.222
Via: SIP/2.0/TCP client.biloxi.example.com:5060
 ;branch=z9hG4bKnashds7
 ;received=192.0.2.201
Max-Forwards: 68
From: Bob <sip:bob@biloxi.example.com>
 ;tag=314159
To: Alice <sip:alice@atlanta.example.com>
 ;tag=9fxced76sl
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 1 BYE
Content-Length: 0
```

Alice's SIP phone

TELECOM Bretagne


```
SIP/2.0 200 OK
Via: SIP/2.0/TCP ss1.atlanta.example.com:5060
 ;branch=z9hG4bK2d4790.1
 ;received=192.0.2.111
Via: SIP/2.0/TCP ss2.biloxi.example.com:5060
 ;branch=z9hG4bK721e4.1
 ;received=192.0.2.222
Via: SIP/2.0/TCP client.biloxi.example.com:5060
 ;branch=z9hG4bKnashds7
 ;received=192.0.2.201
From: Bob <sip:bob@biloxi.example.com>;tag=314159
To: Alice <sip:alice@atlanta.example.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 1 BYE
Content-Length: 0
```


SIP call flows - session establishment through two proxies (RFC 3665)


```
SIP/2.0 200 OK
Via: SIP/2.0/TCP ss2.biloxi.example.com:5060
 ;branch=z9hG4bK721e4.1
 ;received=192.0.2.222
Via: SIP/2.0/TCP client.biloxi.example.com:5060
 ;branch=z9hG4bKnashds7
 ;received=192.0.2.101
From: Bob <sip:bob@biloxi.example.com>
 ;tag=314159
To: Alice <sip:alice@atlanta.example.com>
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 1 BYE
Content-Length: 0
```


Proxy 2
F23
SIP/2.0 200 OK
Via: SIP/2.0/TCP client.biloxi.example.com:5060
;branch=z9hG4bKnashds7
;received=192.0.2.201
From: Bob <sip:bob@biloxi.example.com>
;tag=314159
To: Alice <sip:alice@atlanta.example.com>
;tag=9fxced76sl
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 1 BYE
Content-Length: 0
Bob's SIP phone