SIP Techtorial: Intro to Advanced

Agenda

- SIP Basics
- SIP Request & Responses
- SIP Standards Efforts
- SIP Summary

SIP Basics

- The Session Initiation Protocol (SIP) is an application layer control (signaling) protocol for:
  - creating
  - modifying and
  - terminating

  multimedia sessions with one or more participants

SIP History

- SIP was originally a multicast session set-up protocol for the I2 (mid-late 90s)
  - then someone figured out it was good for unicast
- First Standardized in March 1999 in RFC 2543
- Revised Standard in May 2002 in RFC 3261, with
  - 34 Standards Track Extension RFCs
  - 15 Working Group Internet Drafts, and

    > 50 individual IDs that are not WG items (yet)

SIP – What is it?

SIP is a Session Set-up Protocol

SIP sessions include:

- **Internet multimedia conferences**
- **Internet telephone calls**
- **Internet Video sessions**
- **Multimedia distribution**
- **Subscriptions and Notifications**
- **Publications of State**

SIP Basics (cont'd)

SIP members can:

- **communicate via:**
  - unicast
  - multicast
  - via a mesh of unicast relations or
  - a combination of these
- **in IPv4 and IPv6 environments using:**
  - UDP
  - TCP
  - SCTP or
  - TLS over TCP

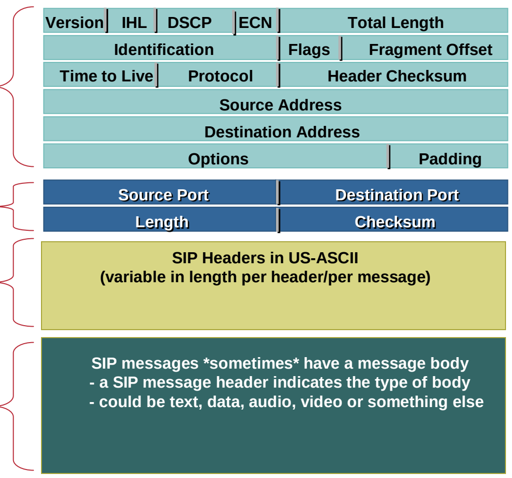
Generic SIP Packet format

IPv4 Header is 20 Bytes and Binary

UDP Header is 8 Bytes and Binary (Layer 4 here could also be TCP or SCTP)

SIP Header is Text-based and variable in length

SIP message body is also variable, but not always present (depending on the Message-type)



The Power of SIP

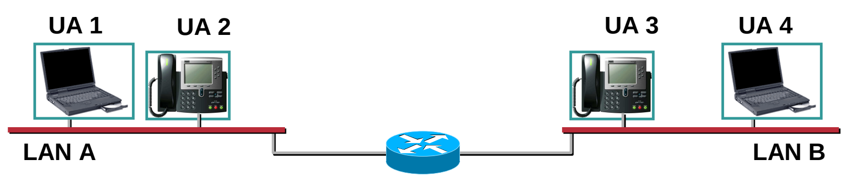
- SIP uses several existing IETF protocols to provide:
  - Message formatting (HTTP 1.1) RFC 2616
  - Media Description (SDP) RFC 2327
  - Media (RTP) RFC 3550 and (RTSP) RFC 2326
  - Addressing (URL) RFC 1738 and (URI) RFC 2396
  - Name resolution and mobility (DHCP) RFC 2131 and (DNS) RFCs 1034&1035
  - Application encoding (MIME) RFC 2045
  - Security (TLS) RFC 2246 and (IPsec) RFC 2401&2406

SIP Basics (cont'd)

SIP components include:

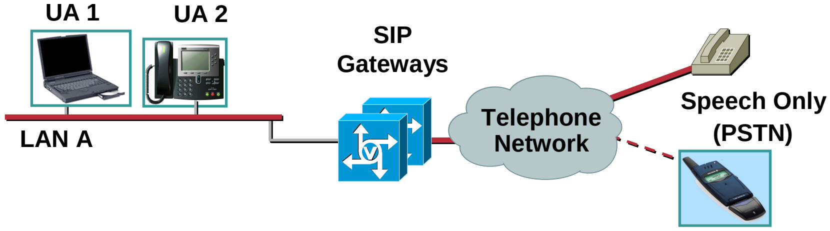
- User Agents (UAs)
- Gateways
- Registrar Servers
- Proxy Servers
- Redirect Servers

SIP User Agents



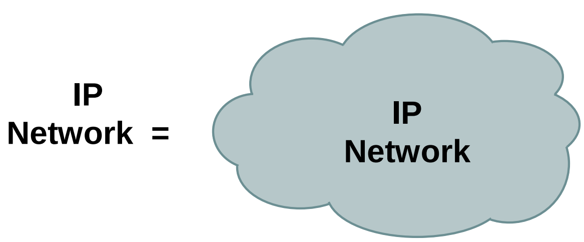
- Client—Server model
- User Agent Client (UAC)—Initiates sessions
- User Agent Server (UAS)—Responds to session requests
- User Agent = UAC + UAS

SIP Gateways



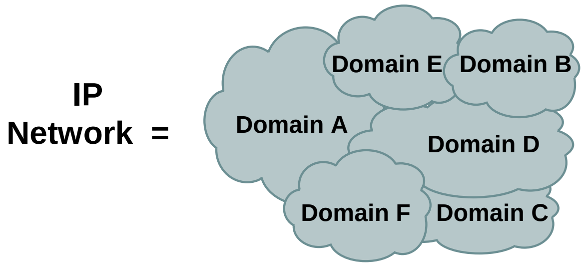
- **Translation between SIP protocol format to and from non-SIP protocol format**

A “cloud” is a “cloud”... or is it?



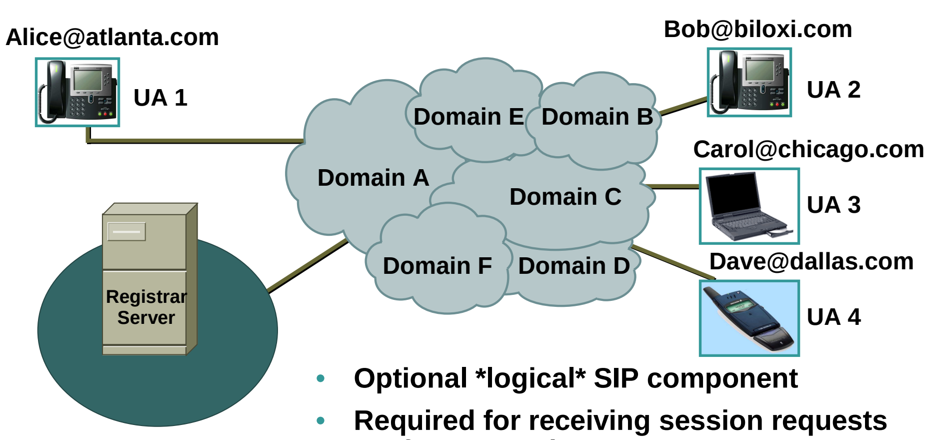
- **When referring to an “IP Network”**

A “cloud” is a “cloud”... or is it?



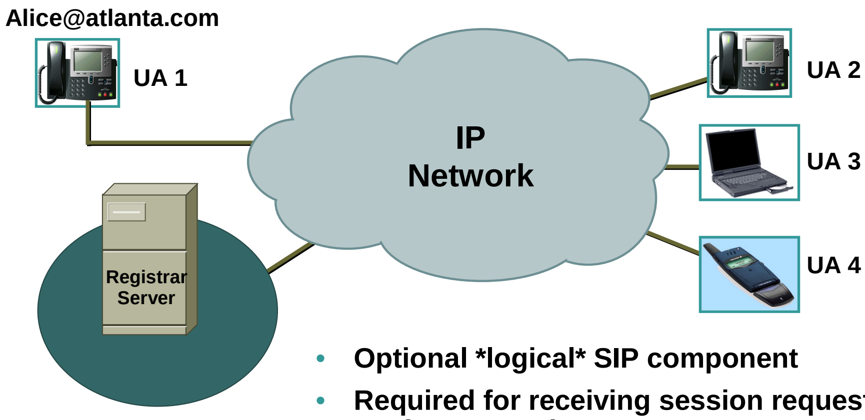
- When referring to an “IP Network”
- This is what it will look more like
- SLAs will determine packet paths

SIP Registrar Server



- Optional *logical* SIP component
- Required for receiving session requests from Domain Proxy Server
- Does not route SIP messages
- Also a mechanism for “Presence”

SIP Registrar Server

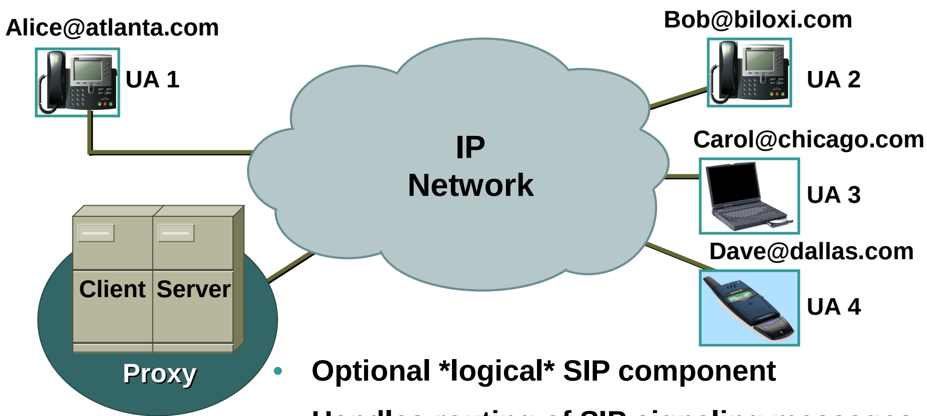


- Optional *logical* SIP component
- Required for receiving session requests from Domain Proxy Server
- Does not route SIP messages
- Also a mechanism for “Presence”

SIP Servers—Registrar Server

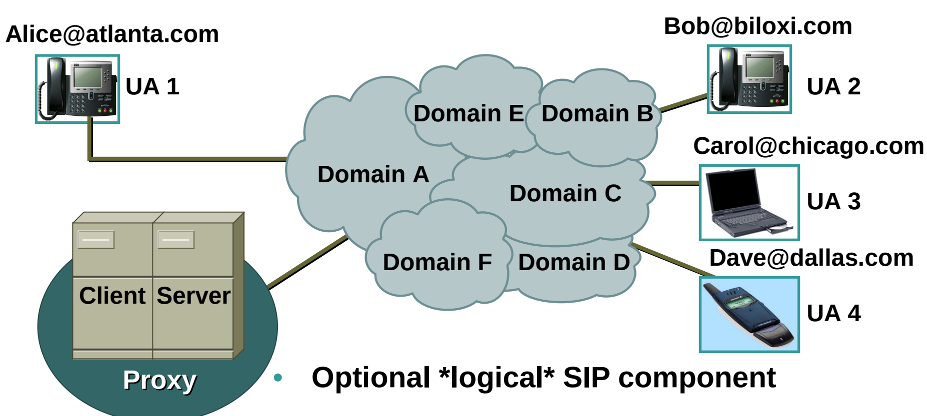
- An optional SIP *logical* component [can coexist with redirect and proxy server on same physical Computer]
- Binds the SIP URI of a user to the device known to that SIP domain
- Once a SIP UA is registered within a domain, the domain Proxy Server is able to route session requests to that user (agent) properly
- Not required to make a session request
- Registrar server is the device that handles SIP REGISTER messages from non-gateway SIP user agents
- Registrar server stores the values from a user agent REGISTER messages for location services

SIP Proxy Server



- Optional *logical* SIP component
- Handles routing of SIP signaling messages
- Can add/modify/remove Headers from SIP messages

SIP Proxy Server

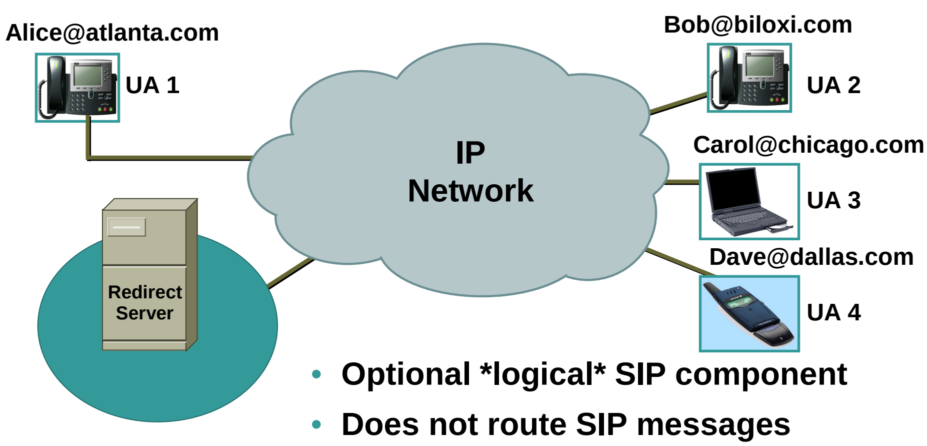


- Optional *logical* SIP component
- Handles routing of SIP signaling messages
- Can add/modify/remove Headers from SIP messages

SIP Servers—Proxy Server

- An optional SIP *logical* component
  – can coexist with registrar and proxy server on same physical computer
- Handles the routing of SIP messages
- SIP proxies can insert and/or remove one or more headers from SIP messages;
  “Record Route” and “Via” Headers, for example

SIP Redirect Server



- Optional *logical* SIP component
- Does not route SIP messages
- Returns a redirect to UAC for directed routing to the given “new” destination

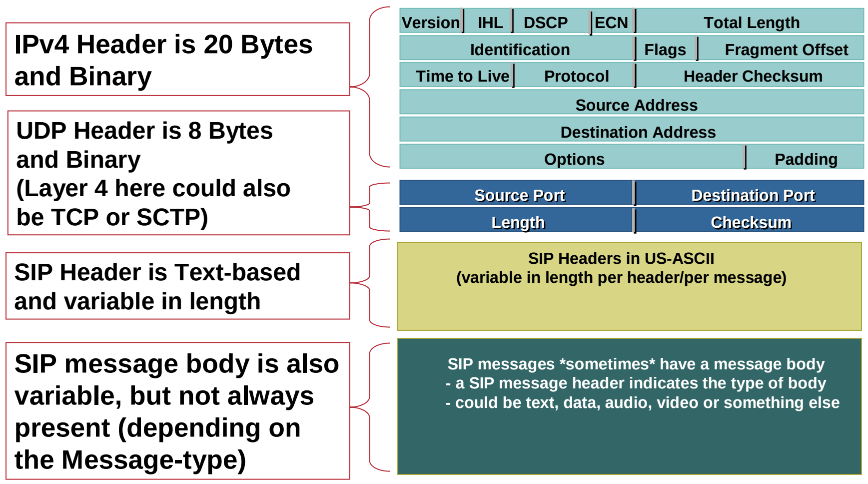
SIP Servers—Redirect Server

- An optional SIP component
- A redirect server does not route messages
- The redirect server determines next destination of the now moved UA and returns a 3xx redirect message for where that new location is with the translated addresses in the Contact: header
- The originating UA initiates a new session using the information supplied from the redirect server

SIP Addressing

- Fully-Qualified Domain Names
  sip:jdoe.cisco.com
- SMTP-style Domain Names [RFC 2368]
  sip:jdoe@cisco.com
- E.164 style addresses [RFC 2806]
  sip:14085551234@gateway.com; user=phone
    user=phone means this is a gateway
    (gateway.com is the FQDN of the egress IP gateway)
- Mixed addresses
  sip:14085551234@10.1.1.1; user=phone
  sip:jdoe@10.1.1.1

General SIP Packet format



SIP—Headers Explained

INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bK776asdhds
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

- Message body goes down here
- Content-Length Header indicates one is present

SIP—Headers Explained

Method = Command Request URI SIP Protocol Version

INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bK776asdhds
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

- Message body goes down here
- Content-Length Header indicates one is present

SIP—Headers Explained

List of all SIP devices to be included in the path

Mandatory header in all SIP Requests

'branch' is for identifying forking

Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bK776asdhds
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

- Message body goes down here
- Content-Length Header indicates one is present

SIP—Headers Explained

Mandatory header in all SIP Requests except INFO

Maximum Number of SIP Server hops permissible in signal path

Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

- Message body goes down here
- Content-Length Header indicates one is present

SIP—Headers Explained

Mandatory header in all SIP Requests

Destination for the SIP Message (but isn’t used for routing message)

To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

- Message body goes down here
- Content-Length Header indicates one is present

SIP—Headers Explained

Mandatory header in all SIP Requests

The Originator of the SIP Message

Pseudo Random String (used for id)

From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

- Message body goes down here
- Content-Length Header indicates one is present

SIP—Headers Explained

Mandatory header in all SIP Requests

Globally Unique Call Identifier within the Domain for session

Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

- Message body goes down here
- Content-Length Header indicates one is present

SIP—Headers Explained

Mandatory header in all SIP Requests

Incremental Sequence number of Method used from this UA

CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

- Message body goes down here
- Content-Length Header indicates one is present

SIP—Headers Explained

Direct route to Originator with Fully Qualified Domain Name or IP address

Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

- Message body goes down here
- Content-Length Header indicates one is present

SIP—Headers Explained

Description of the message body (if and when there is one)

Content-Type: application/sdp
Content-Length: 142

- Message body goes down here
- Content-Length Header indicates one is present

SIP—Headers Explained

Expected header in all SIP Requests

Content Length in Octets of Message Body

Content-Length: 142

- Message body goes down here
- Content-Length Header indicates one is present

SIP—Headers Explained

INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bK776asdhds
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

- Message body goes down here
- Content-Length Header indicates one is present

RFC 2327

Session Description Protocol (SDP)

- A session description protocol for multimedia connections
- Presents a set of parameters for a multimedia session
  - Similar to H.245 in functionality
- Developed by IETF MMUSIC WG
- Simple/Flexible
  - Text-based
  - Extensible
- SIP Offer/Answer Model is RFC 3264

"Lines" below are in order

- v = protocol version
- o = owner/creator and session identifier
- s = session name
- c = connection information – not required if included in all media
- k = encryption keys
- t = time the session is active
- m = media description and transport address
- a = (zero or more) media attributes lines

RFC 2327

SIP Message Body for multimedia

- An SDP message body for voice only

- Audio
- UDP port # 49172
- Codecs supported: G.711

```
v=0
o=alice 2890844526 2890844526 IN IP4 atlanta.com
c=IN IP4 10.1.3.33
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

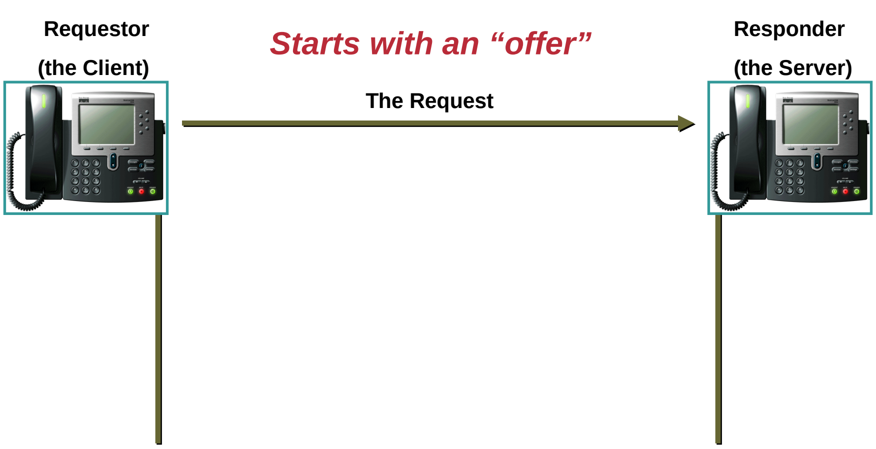
- An SDP message body for voice and video

- Audio
- UDP port # 49172
- Codecs supported: G.711

- Video
- UDP port # 51172
- Codecs supported: H.261, H.263

```
v=0
o=alice 2890844526 2890844526 IN IP4 atlanta.com
c=IN IP4 10.1.3.33
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
m=video 51172 RTP/AVP 31 34
a=rtpmap:31 H.261/90000
a=rtpmap:34 H.263/90000
```


SIP is a Request/Response Protocol



SIP Methods (which are Requests) from RFC 3261

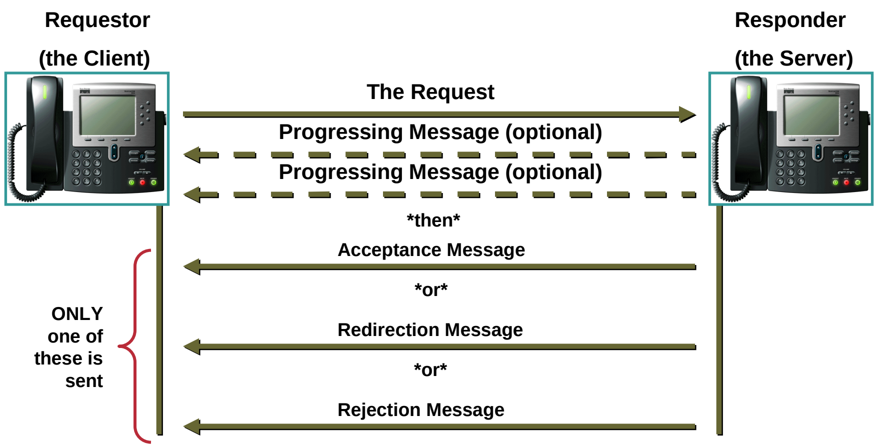
- **INVITE—A user or service is being invited to participate in a multimedia session**
- **ACK—Confirms that a client has received a final response to an INVITE request**
- **BYE—Terminates an existing session; can be sent by any user agent (in a multiparty session)**
- **CANCEL—Cancels pending requests; does not terminate sessions that have been accepted**
- **OPTIONS—Queries the capabilities of servers**
- **REGISTER—Registers the user agent with the registrar server of a domain**

Additional SIP Request Methods

- **INFO (RFC 2976)**
- **PRACK (RFC 3262)**
- **SUBSCRIBE and NOTIFY (RFC 3265)**
- **UPDATE (RFC 3311)**
- **MESSAGE (RFC 3428)**
- **REFER (RFC 3515)**
- **PUBLISH (RFC 3903)**

RFC 3261

SIP is a Request/Response Protocol



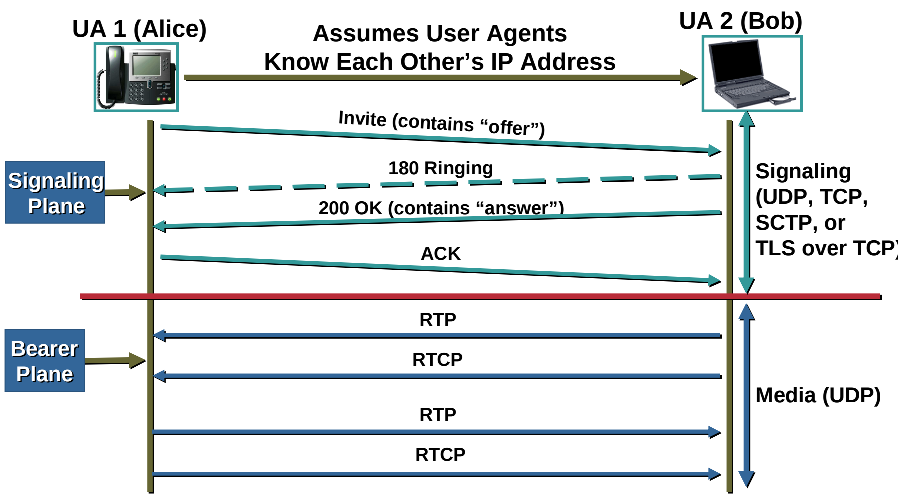
Expects an "answer"

RFC 3261

SIP Responses (which are Replies)

	Description	Examples
1xx	Informational – Request received, continuing to process request.	100 Trying 180 Ringing 181 Call is Being Forwarded 183 Session Progressing
2xx	Success – Action was successfully received, understood and accepted.	200 OK 202 Acceptable
3xx	Redirection – Another SIP Element needs to be contacted in order to complete the request.	300 Multiple Choices 301 Moved Permanently 302 Moved Temporarily
4xx	Client Error – Request contains bad syntax or cannot be fulfilled at this server.	401 Unauthorized 406 Not Acceptable 407 Proxy Authentication Required 486 Busy Here 487 Request Terminated 488 Not Acceptable Here
5xx	Server Error – Server failed to fulfill an apparently valid request.	502 Bad Gateway 503 Service Unavailable
6xx	Global Failure – Request is invalid at any server.	600 Busy Everywhere 603 Decline

SIP Endpoint-to-Endpoint Signaling without a Server



3 Mandatory Packets for Establishment Handshake INVITE/200 OK/ACK

TECVVT118 Session Agenda

- IETF Standardization Process
- SIP Basics
- SIP Request & Responses
- SIP Standards Efforts
- SIP Interworking with MGCP & H.323
- SIP Working Efforts
- SIP Summary

SIP Methods

- INVITE
- ACK
- BYE
- CANCEL
- OPTIONS
- REGISTER
- PRACK
- SUBSCRIBE
- NOTIFY
- MESSAGE
- INFO
- UPDATE
- REFER
- PUBLISH

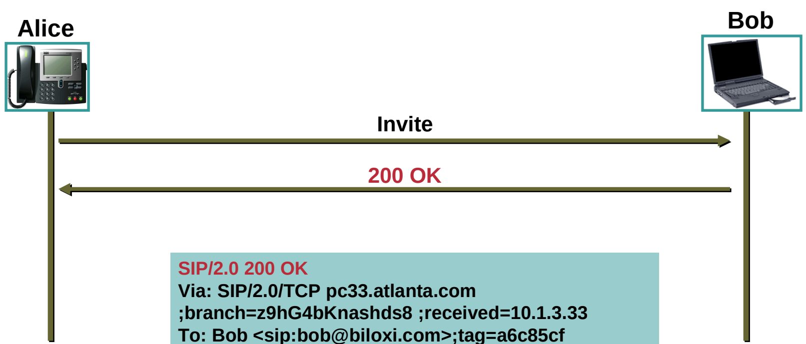
RFC 3261

SIP Methods: INVITE, ACK and BYE

INVITE sip:bob@192.168.10.20 SIP/2.0
Via: SIP/2.0/TCP pc33.atlanta.com
;branch=z9hG4bK776asdhds
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

(Alice's SDP not shown)

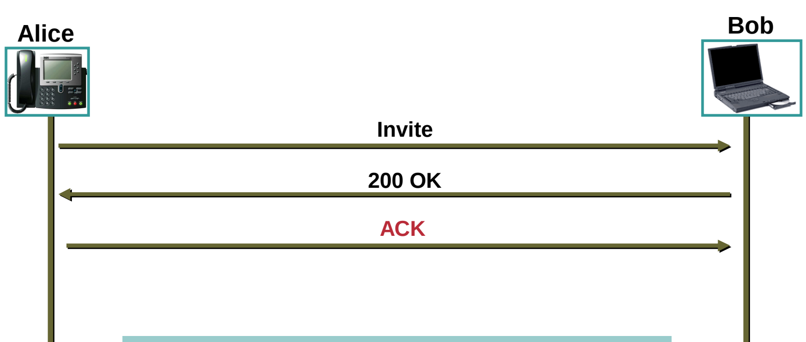
SIP Methods: INVITE, ACK and BYE



SIP/2.0 200 OK
Via: SIP/2.0/TCP pc33.atlanta.com
;branch=z9hG4bKnashds8 ;received=10.1.3.33
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:bob@192.168.10.20>
Content-Type: application/sdp
Content-Length: 131

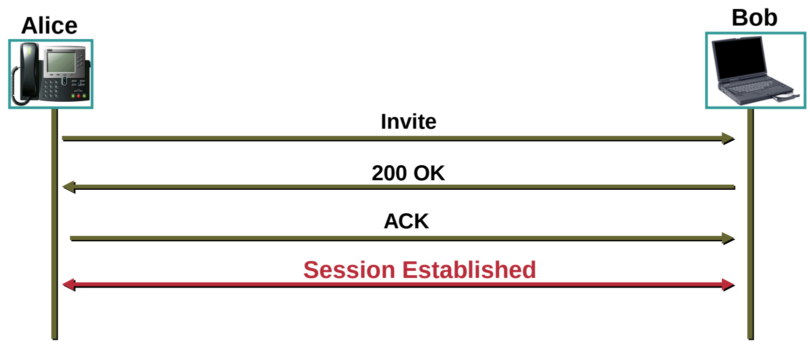
(Bob's SDP not shown)

SIP Methods: INVITE, ACK and BYE



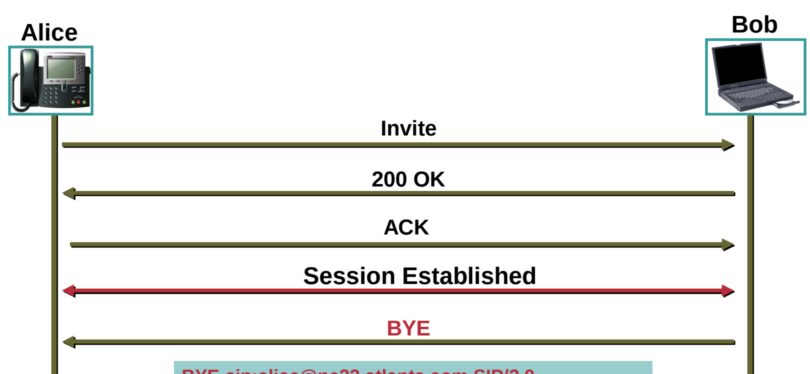
ACK sip:bob@192.168.10.20 SIP/2.0
Via: SIP/2.0/TCP pc33.atlanta.com;branch=z9hG4bKnashds8
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 ACK
Content-Length: 0

SIP Methods: INVITE, ACK and BYE



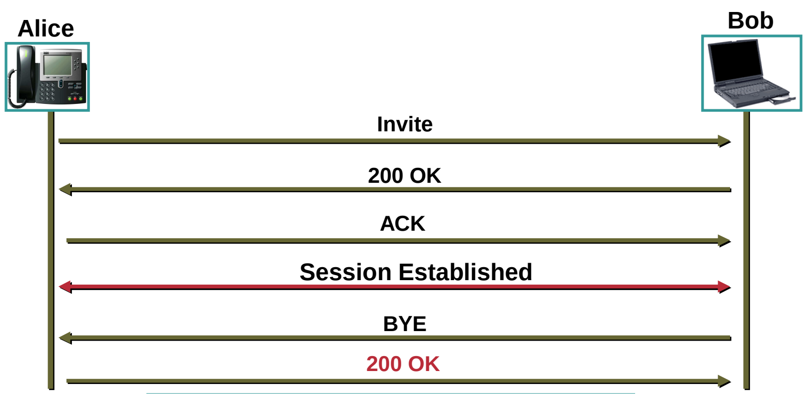
RFC 3261

SIP Methods: INVITE, ACK and BYE



BYE sip:alice@pc33.atlanta.com SIP/2.0
Via: SIP/2.0/TCP 10.1.3.33;branch=z9hG4bKnashds8
Max-Forwards: 70
From: Bob <sip:bob@biloxi.com>;tag=a6c85cf
To: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 231 BYE
Content-Length: 0

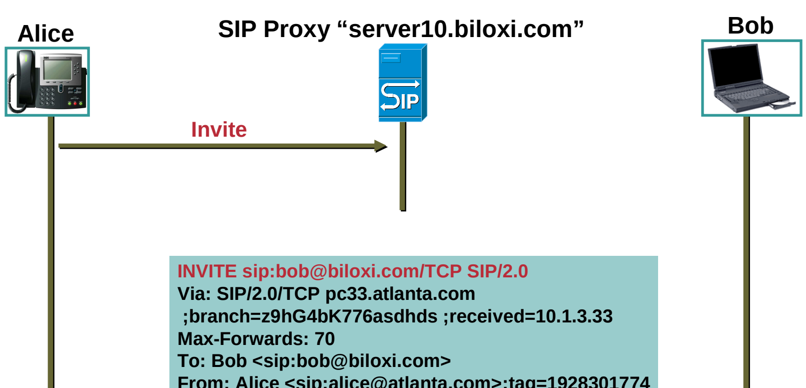
SIP Methods: INVITE, ACK and BYE



SIP/2.0 200 OK
Via: SIP/2.0/TCP 192.168.10.20
From: Alice <sip:alice@atlanta.com>;tag=1928301774
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 231 BYE
Content-Length: 0

RFC 3261

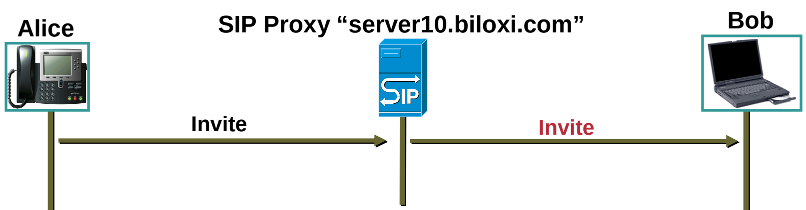
SIP Methods: INVITE, ACK and BYE w/Proxy



INVITE sip:bob@biloxi.com/TCP SIP/2.0
Via: SIP/2.0/TCP pc33.atlanta.com
;branch=z9hG4bK776asdhds ;received=10.1.3.33
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

(Alice's SDP not shown)

SIP Methods: INVITE, ACK and BYE w/Proxy



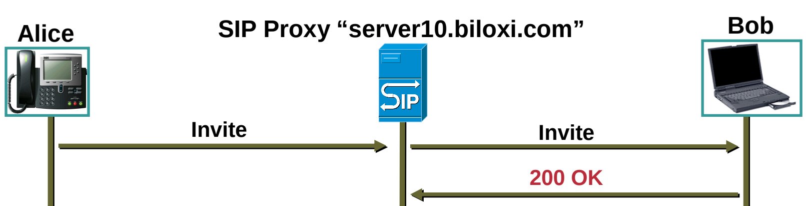
Via Header for Bigbox3 Server

Via Header for Alice’s phone

Max-Forwards decreases 1

```
INVITE sip:bob@192.168.10.20/TCP SIP/2.0
Via: SIP/2.0/TCP server10.biloxi.com
 ;branch=z9hG4bK4b43c2ff8.1 ;received=192.168.10.1
Via: SIP/2.0/TCP pc33.atlanta.com
 ;branch=z9hG4bK776asdhds ;received=10.1.3.33
Max-Forwards: 69
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

(Alice's SDP not shown)
```

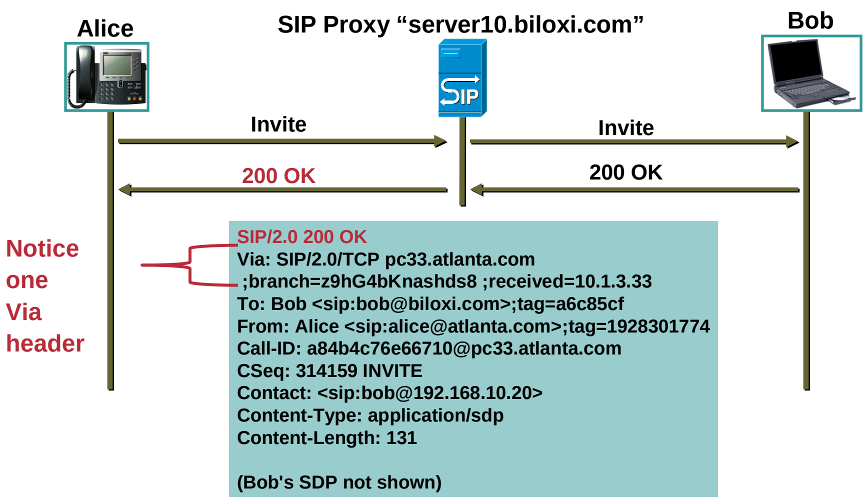

SIP Methods: INVITE, ACK and BYE w/Proxy



Notice both Via headers

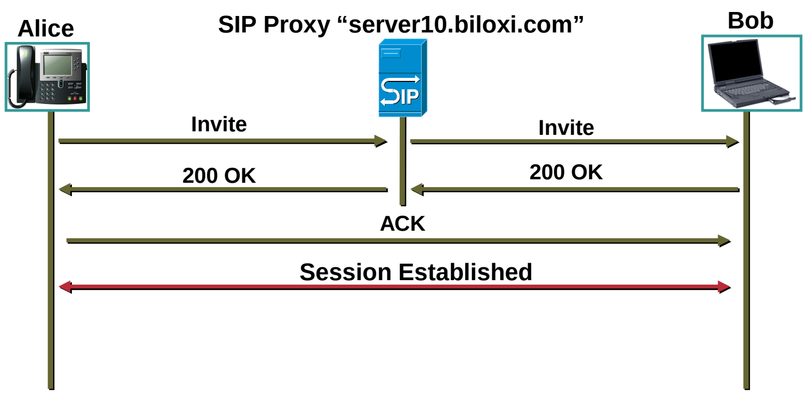
```
SIP/2.0 200 OK
Via: SIP/2.0/TCP server10.biloxi.com
 ;branch=z9hG4bK4b43c2ff8.1 ;received=192.168.10.1
Via: SIP/2.0/TCP pc33.atlanta.com
 ;branch=z9hG4bKnashds8 ;received=10.1.3.33
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:bob@192.168.10.20>
Content-Type: application/sdp
Content-Length: 131

(Bob's SDP not shown)
```

SIP Methods: INVITE, ACK and BYE w/Proxy



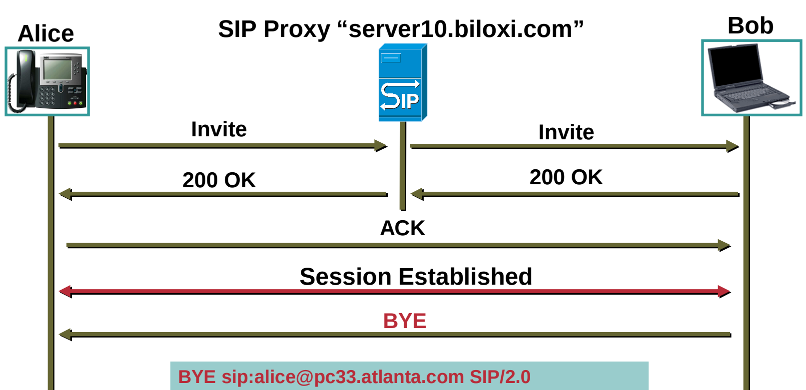
SIP Methods: INVITE, ACK and BYE w/Proxy

SIP Methods: INVITE, ACK and BYE w/Proxy



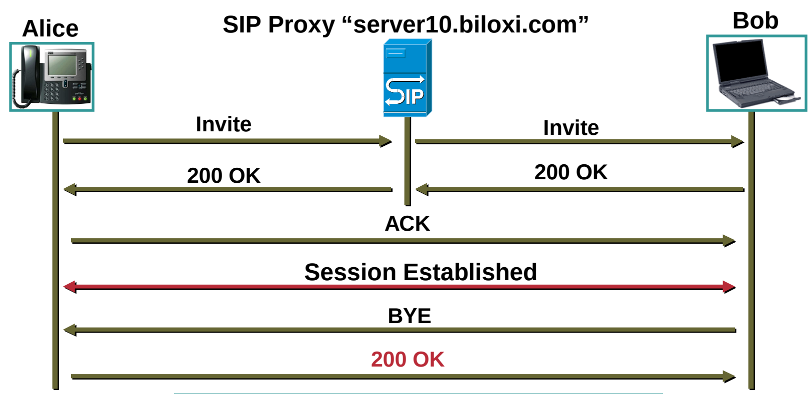
RFC 3261

SIP Methods: INVITE, ACK and BYE w/Proxy



BYE sip:alice@pc33.atlanta.com SIP/2.0
Via: SIP/2.0/TCP 10.1.3.33;branch=z9hG4bKnashds8
Max-Forwards: 70
From: Bob <sip:bob@biloxi.com>;tag=a6c85cf
To: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 231 BYE
Content-Length: 0

SIP Methods: INVITE, ACK and BYE w/Proxy



SIP/2.0 200 OK
Via: SIP/2.0/TCP 192.168.10.20
From: Alice <sip:alice@atlanta.com>;tag=1928301774
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 231 BYE
Content-Length: 0

Stateless, Stateful & “Really” Stateful Proxy

Transaction Stateless

The proxy server forwards all messages and responses without maintaining any state

Transaction Stateful

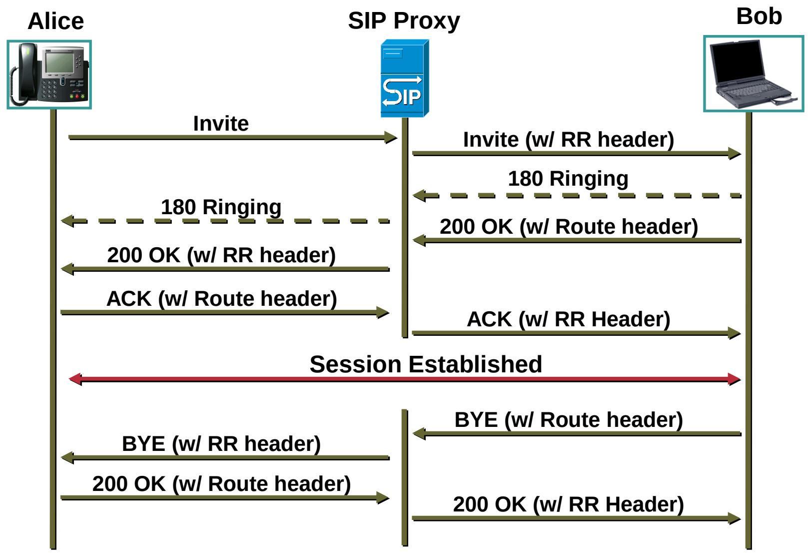
A Proxy Server that receives a SIP Request retains state of that transaction until that Server receives a Final Response (meaning a 2XX, 3XX, 4XX, 5XX or 6XX Response). Transaction Stateful has no knowledge of a session Update Request (UPDATE), a Transfer Request (REFER) or of a Termination Request (BYE)

Dialog Stateful

When Record Route Header is utilized by a Proxy during the first SIP Request to ensure all remaining messages traverse that Proxy; this applies to each proxy that is in the signaling path between UAs

RFC 3261

SIP Call Flow with Proxy and Record Route



Record-Route and Route Headers

Record-Route Header

Record-Route: server10.biloxi.com

- Optional SIP message header
- *Can* be inserted by any and all Proxies viewing a SIP message
- Routes all SIP requests and responses for that Call-ID through that proxy
  – a new Call-ID Request is treated independently

Route Header

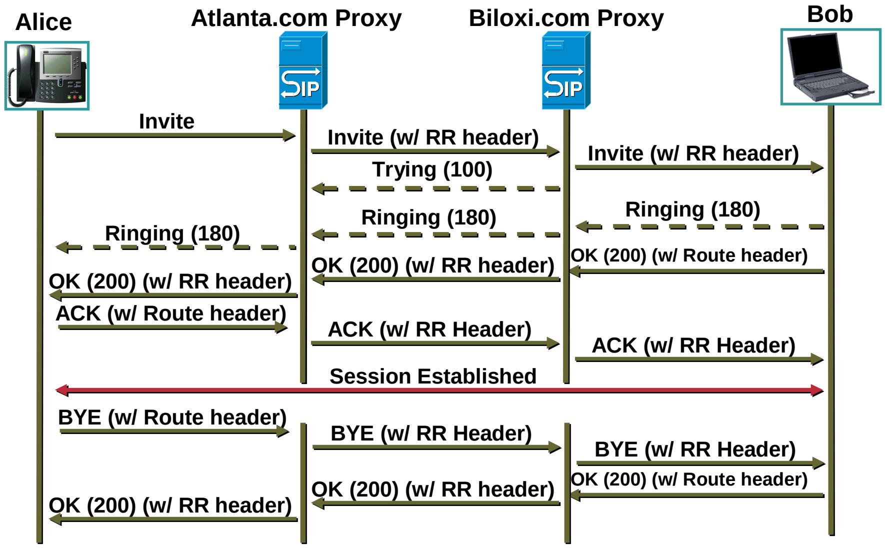
Route: server10.biloxi.com

- Built from Record-Route header field in same Call-ID Response
- Header deleted at SIP element identified by URI field

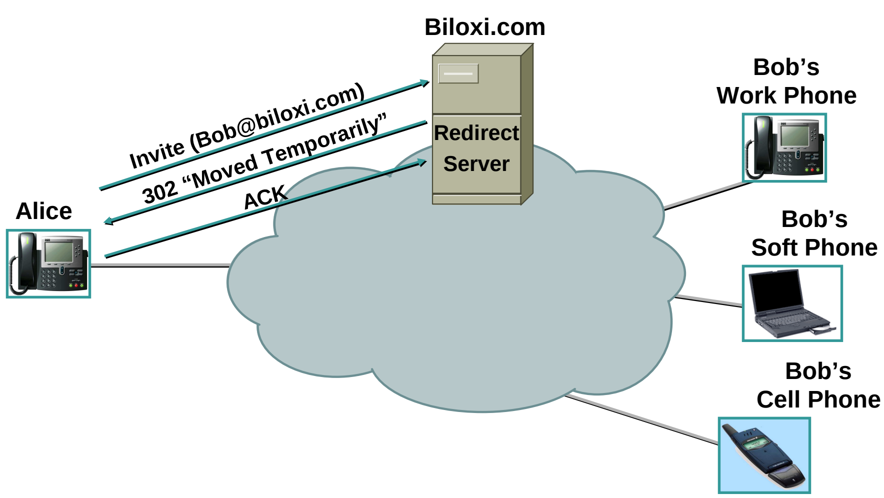
Useful for billing, call control, CALEA

RFC 3261

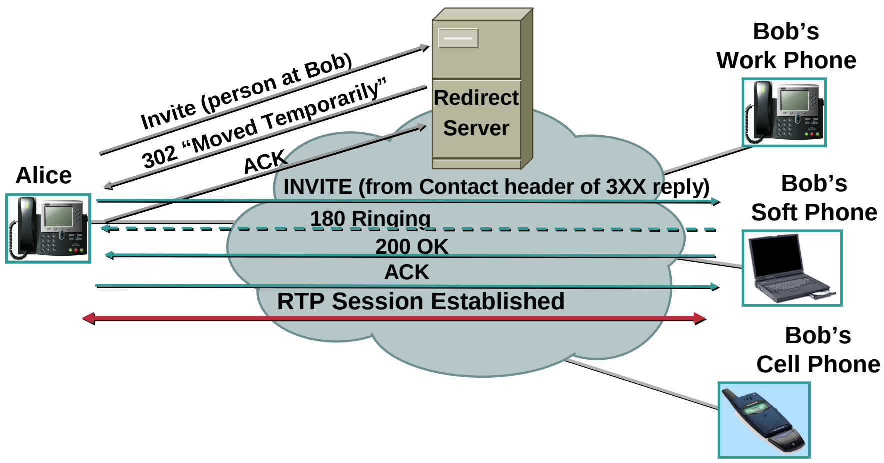
SIP Call Flow w/ 2 Proxies and Record Route



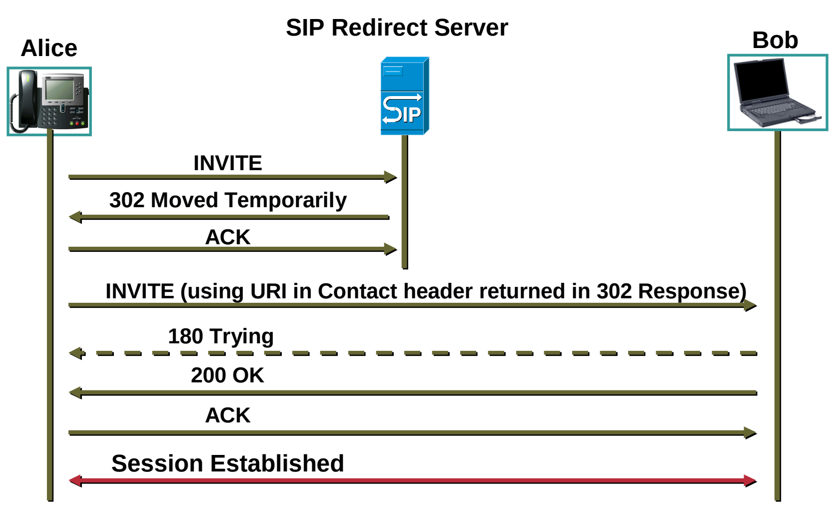
SIP Redirect Server



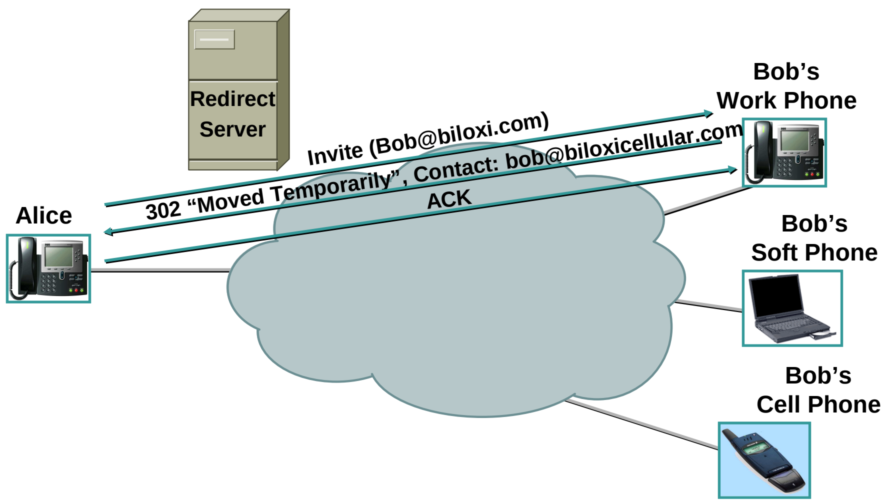
SIP Redirect Server



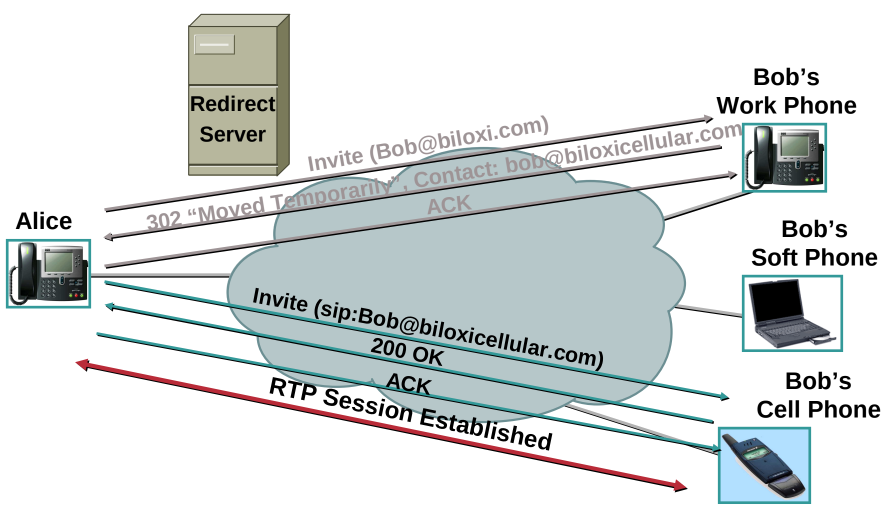
SIP Call Flow with Redirect



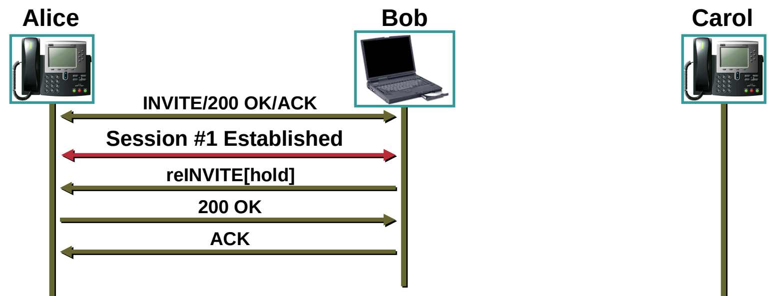
SIP Redirecting at the User Agent Server



SIP Redirecting at the User Agent Server

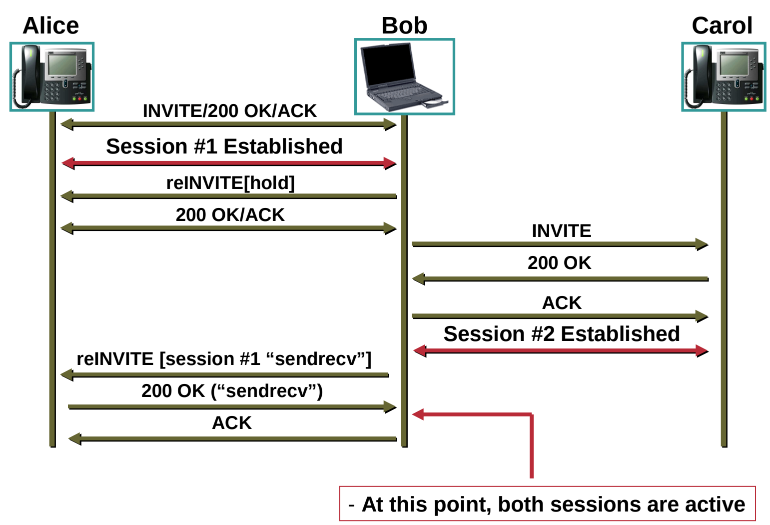


SIP Methods: reINVITE to Call Hold



- Placing an existing session into a “hold” state is done by sending an INVITE from the holder to the holdee with the same Call-ID sequence, but changing the state of the session from “sendrecv” to “sendonly”
- The 200 OK will have the reciprocal “recvonly”

SIP Methods: 3-Way Conference via Bob



Bob is the media mixer here, but he could have pointed to a DSP just as easy

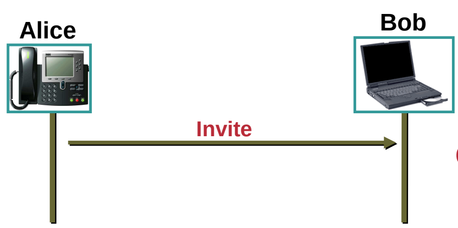
RFC 3261

SIP Methods: INVITE, ACK and BYE

- **INVITE—A user or service is being invited to participate in a multimedia session**
- **ACK—Confirms that a client has received a final response to an INVITE request**
- **BYE—Terminates an existing session; can be sent by any user agent in a dialog**

RFC 3261

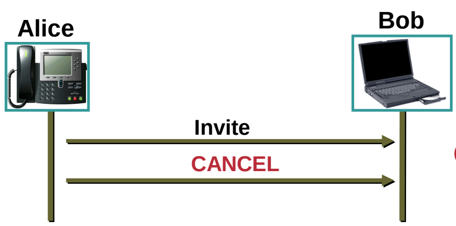
SIP Methods: CANCEL




```
INVITE sip:bob@192.168.10.20 SIP/2.0
Via: SIP/2.0/TCP 10.1.3.33
 ;branch=z9hG4bK776asdhds
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 314159 INVITE
Contact: <sip:alice@atlanta.com>
Content-Type: application/sdp
Content-Length: 142

(Alice's SDP not shown)
```

CANCEL— discontinues pending requests; does not terminate sessions that have been accepted

SIP Methods: CANCEL

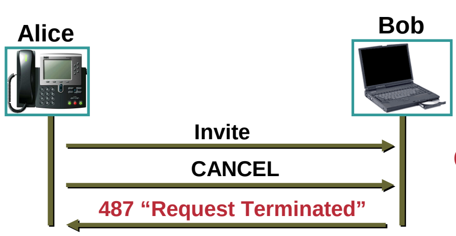



```
CANCEL sip:bob@192.168.10.20 SIP/2.0
Via: SIP/2.0/TCP 10.1.3.33
 ;branch=z9hG4bK776asdhds
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 10197 CANCEL
Contact: <sip:alice@atlanta.com>
Reason: SIP ;cause=486 ;text="Busy Here"
Content-Length: 0
```

CANCEL— discontinues pending requests; does not terminate sessions that have been accepted

- Reason Header will give the reason
- Here, the caller may have hung up to accept another call before the first was accepted

SIP Methods: CANCEL

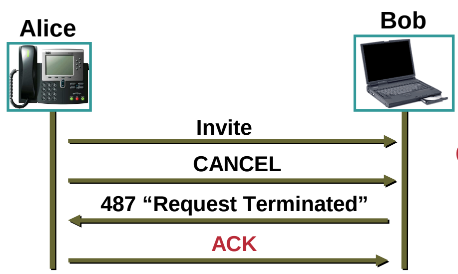



```
SIP/2.0 487 Request Terminated
Via: SIP/2.0/TCP 10.1.3.33
From: Alice <sip:alice@atlanta.com>;tag=1928301774
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 10197 CANCEL
Content-Length: 0
```

CANCEL— discontinues pending requests; does not terminate sessions that have been accepted

- 487 "Request Terminated" is the proper Response to an INVITE Request

SIP Methods: CANCEL



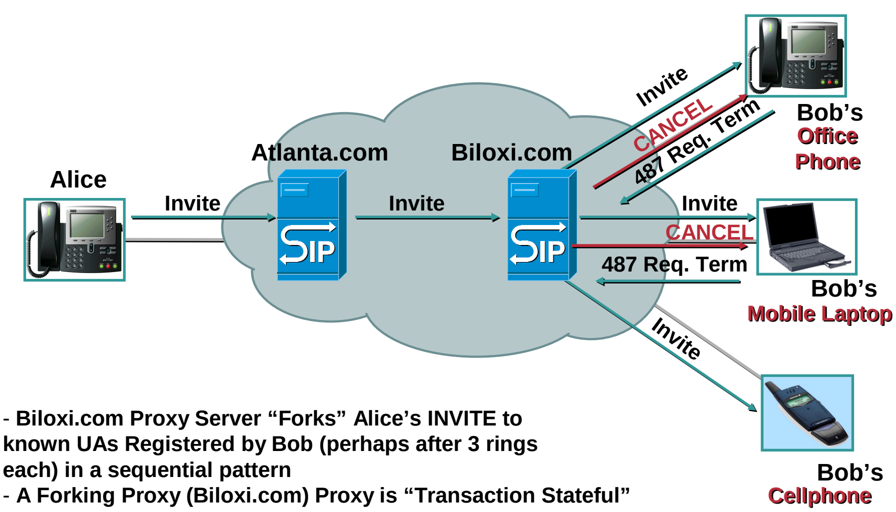

```
ACK sip:bob@192.168.10.20 SIP/2.0
Via: SIP/2.0/TCP 10.1.3.33
 ;branch=z9hG4bK776asdhds
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 10197 ACK
Content-Length: 0
```

CANCEL— discontinues pending requests; does not terminate sessions that have been accepted

- ACK always follows a 4XX Response

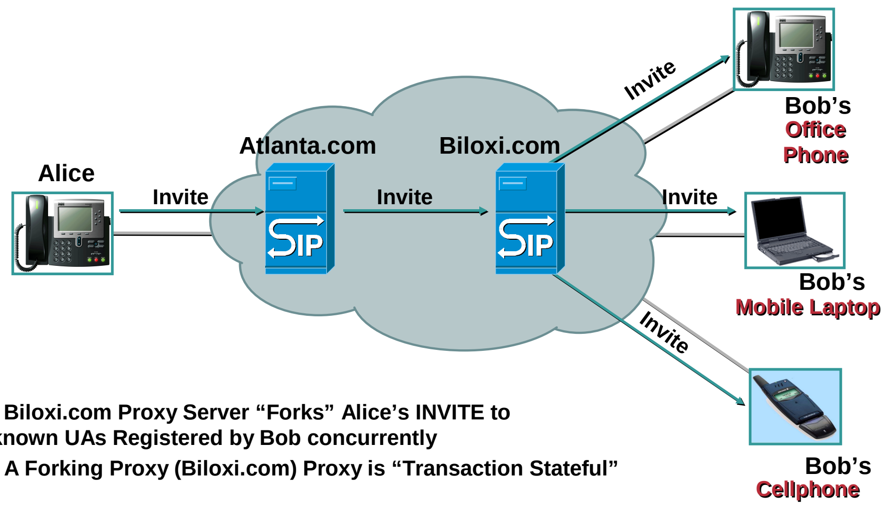
RFC 3261

Call Forking (Sequentially)



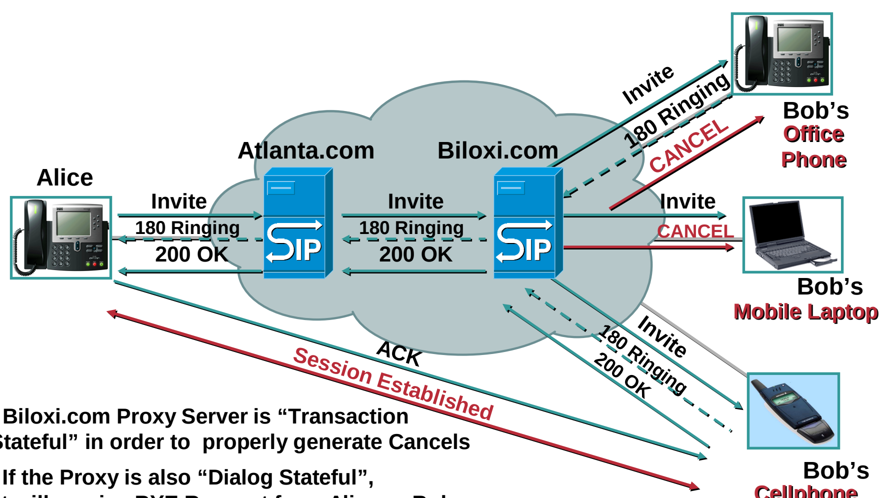
- Biloxi.com Proxy Server “Forks” Alice’s INVITE to known UAs Registered by Bob (perhaps after 3 rings each) in a sequential pattern
- A Forking Proxy (Biloxi.com) Proxy is “Transaction Stateful”
- “branch” values are different per forked INVITE

Call Forking (Concurrently)



- Biloxi.com Proxy Server "Forks" Alice's INVITE to known UAs Registered by Bob concurrently
- A Forking Proxy (Biloxi.com) Proxy is "Transaction Stateful"

Call Forking Flow



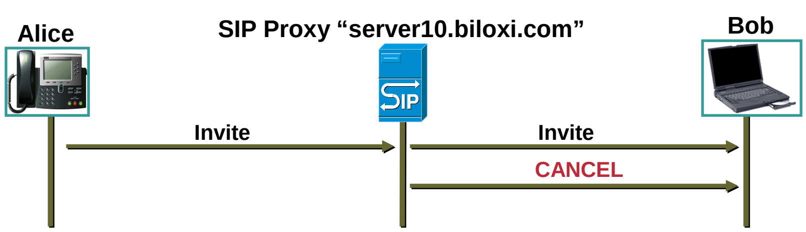
- Biloxi.com Proxy Server is "Transaction Stateful" in order to properly generate Cancels
- If the Proxy is also "Dialog Stateful", it will receive BYE Request from Alice or Bob

Call Forking

- Proxy forks original INVITE to multiple user agents
- Forks can be sequential, or concurrent
- “Branch” values within the Via header are different for each forked INVITE
- Session established to first user agent to respond with 200 OK
- CANCEL sent to non-respondent user agents within forking procedure
- Proxy MUST be at least “Transaction Stateful” for Forking
- Proxy could be controlled by Callee’s User Profile

RFC 3261

SIP Methods: CANCEL w/Proxy

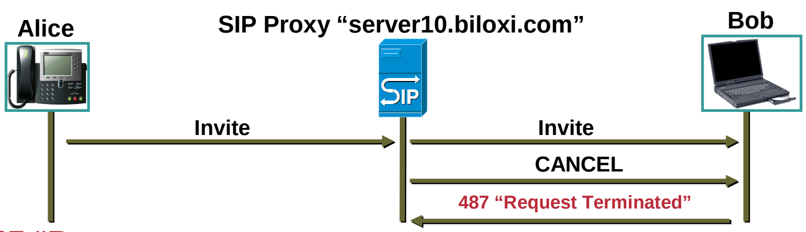


Why would a Proxy do a CANCEL by itself?

- A Sequential or concurrent forking cleanup, for example
- In this case, Proxy received 200 OK from another Forked INVITE

CANCEL sip:bob@192.168.10.20/TCP SIP/2.0
Via: SIP/2.0/TCP server10.biloxi.com
;branch=z9hG4bK4b43c2ff8.1 ;received=192.168.10.1
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Server10 <sip:server10.biloxi.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 6187 CANCEL
Contact: <sip:server10.biloxi.com>
Reason: SIP ;cause=200 ;text=”call completed elsewhere”
Content-Length: 0

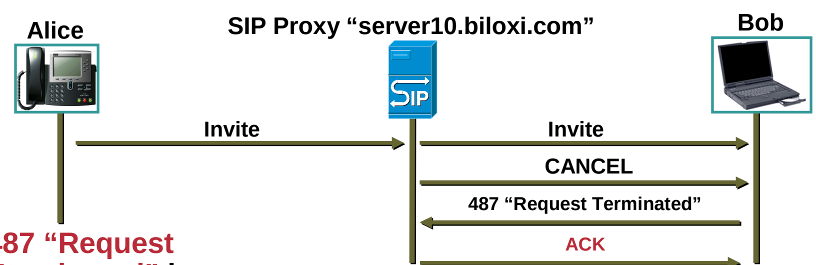
SIP Methods: CANCEL w/Proxy



- **487 “Request Terminated” is immediately sent by Bob’s UA to cancel this INVITE Request**
- **If Bob’s UA had already sent a 200 OK prior to receiving the CANCEL, the CANCEL would be ignored**

```
SIP/2.0 487 Request Terminated
Via: SIP/2.0/TCP 192.168.10.20
From: Alice <sip:alice@atlanta.com>;tag=1928301774
To: Bob <sip:bob@biloxi.com>;tag=a6c85cf
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 6187 CANCEL
Content-Length: 0
```


SIP Methods: CANCEL w/Proxy



- **487 “Request Terminated” is immediately sent by Bob’s UA to cancel this INVITE Request**
- **An ACK is always transmitted in this sequence as part of an INVITE**

```
ACK sip:bob@192.168.10.20/TCP SIP/2.0
Via: SIP/2.0/TCP server10.biloxi.com
   ;branch=z9hG4bK4b43c2ff8.1 ;received=192.168.10.1
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Server10 <sip:server10.biloxi.com>;tag=1928301774
Call-ID: a84b4c76e66710@pc33.atlanta.com
CSeq: 6187 CANCEL
Content-Length: 0
```

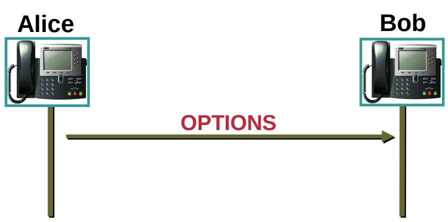
RFC 3261

SIP Methods: CANCEL

- CANCEL— Cancels pending requests; does not terminate sessions that have already been accepted
  – SHOULD only be for INVITE Requests
  – CANCEL Requests cannot be challenged by Servers
  – If a Request exceeds the time in the Expires header, a CANCEL Request should be sent immediately
  – 487 “Request Terminated” is the proper response to a CANCEL of a pending INVITE
  – If a UA has already sent a 200 OK [Final response] prior to receiving the CANCEL, the CANCEL would be ignored

RFC 3261

SIP Methods: OPTIONS



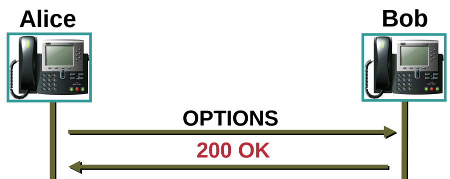

```
OPTIONS sip:bob@192.168.10.20 SIP/2.0
Via: SIP/2.0/TCP pc33.atlanta.com
 ;branch=z9hG4bK77i832k9 ;received=10.1.3.33
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e6Kr456@pc33.atlanta.com
CSeq: 22756 OPTIONS
Contact: <sip:alice@pc33.atlanta.com>
Allow: INVITE, ACK, OPTIONS, BYE, CANCEL, REFER,
  SUBSCRIBE, NOTIFY, MESSAGE, UPDATE
Accept: application/sdp, application/pidf-xml
Content-Length: 0
```

OPTIONS—enables queries of the capabilities of UASs or servers

– Allows a UAC to discover the supported:
  - methods,
  - content types,
  - extensions,
  - codecs,
  - etc.

  without "ringing" the other party

SIP Methods: OPTIONS



SIP/2.0 200 OK
Via: SIP/2.0/TCP sip:alice@atlanta.com
;branch=z9hG4bK77i832k9 ;received=10.1.3.33
To: Bob <sip:bob@biloxi.com>; tag=a6c85e3
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e6Kr456@pc33.atlanta.com
CSeq: 22756 OPTIONS
Contact: <sip:bob@biloxi.com>
Contact: <sip:bob_home@biloxi.com>
Allow: INVITE, ACK, OPTIONS, BYE, CANCEL, REFER, NOTIFY, MESSAGE
Accept: application/sdp, text/plain, image/jpeg
Accept-language: en, fr
Content-Type: application/sdp
Content-Length: 274

(Bob's SDP indicating codecs supported)

OPTIONS—enables queries of the capabilities of UASs or servers

- 200 OK provides all:
  - Contacts known
  - Methods supported
  - Language supported
  - Message Body type
- A 486 "Busy Here" is returned if the UA is not ready to accept a new Request

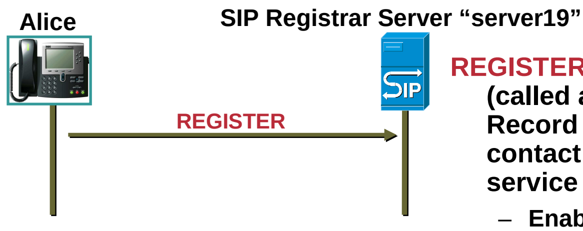
Notice the shorter list here

SIP Methods: OPTIONS

- **OPTIONS** - allows a UA to query another UA or a proxy server as to its capabilities
  - Allows a UAC to discover the supported methods, content types, extensions, codecs, etc. without "ringing" the other party
  - All UAs MUST support the OPTIONS method
  - A 200 OK provides all:
    - Contacts known to that UAS
    - Methods supported by that UAS
    - Language supported by that UAS
    - Message Body type accepted
      - assumed to be "application/sdp" if no Accept header
      - SDP port MUST be set to zero (0) to ensure this isn't considered an Offer by any SIP element
  - A 486 "Busy Here" is returned if the UA is not ready to accept a new Request

RFC 3261

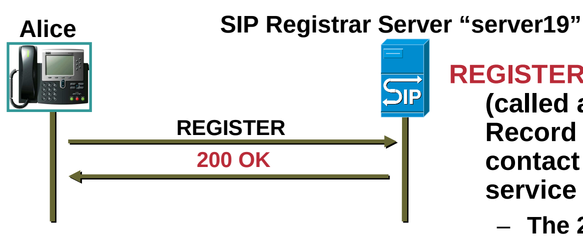
SIP Methods: REGISTER



REGISTER sip:server19.atlanta.com SIP/2.0
Via: SIP/2.0/TCP pc33.atlanta.com
;branch=z9hG4bk2l55n1
To: Alice <sip:alice@atlanta.com>
From: Alice <sip:alice@atlanta.com>;tag=283074
Call-ID: a84b4g96te10@pc33.atlanta.com
CSeq: 31862 REGISTER
Contact: <sip:alice@10.1.3.33>
Expires: 21600
Content-Length: 0

REGISTER - Binds a SIP URI (called an Address of Record (AOR)) with a contact name in a location service
  - Enables UAs to receive SIP messages
  - Registrations represent a dynamic piece of state maintained in a network
- UAs can use three ways to determine the address to send registrations:
  - Configuration
  - Address-of-Record
  - Multicast [224.0.1.75]

SIP Methods: REGISTER



SIP/2.0 200 OK
Via: SIP/2.0/TCP pc33.atlanta.com
;branch=z9hG4bk2l55n1; received=10.1.3.33
To: Alice <sip:alice@atlanta.com>; tag=a6c85e3
From: Alice <sip:alice@atlanta.com>;tag=283074
Call-ID: a84b4g96te10@pc33.atlanta.com
CSeq: 31862 REGISTER
Contact: <sip:alice@pc33.atlanta.com>
Contact: <sip:alice@cm9013.atlanta.com>
Expires: 3600
Contact-Length: 0

REGISTER often challenged

REGISTER - Binds a SIP URI (called an Address of Record (AOR)) with a contact name in a location service
  - The 200 (OK) response from the registrar contains a list of Contact fields enumerating all current bindings
  - Expires Header informs UA how long Registration lasts before a refresh is required
  - Because devices can be “always on”, a domain can request that a SIP device re-authenticate to the domain

RFC 3261

SIP Methods: REGISTER

- **REGISTER—Registers the user agent with the registrar server of a domain**
  - Binds a SIP URI (called an Address of Record (AOR)) and a contact name in a location service
  - enables UAs to receive SIP messages
  - Registrations represent a dynamic piece of state maintained in a network
  - because devices can be “always on”, a domain can request that a SIP device re-authenticate to the domain
  - The 200 (OK) response from the registrar contains a list of Contact fields enumerating all current bindings
  - UAs can use three ways to determine the address to which to send registrations:
    - •by configuration,
    - •using the address-of-record, and
    - •multicast