Session Initiation Protocol

SIP protocol and its extensions
SIP – Need for signaling compression
Main Sources

IETF:
RFC 3261: SIP: Session Initiation Protocol
RFC 3262: Reliability of Provisional Responses in SIP (PRACK)
RFC 3265: SIP Specific Event Notification
RFC 3311: SIP UPDATE method
RFC 3428: SIP Extension for Instant Messaging
RFC 2327: SDP: Session Description Protocol
RFC 3264: An Offer/Answer Model with Session Description Protocol (SDP)

3G Releases 5 … 7:
3GPP TS 24.228 v5.15.0 (2006-10) Signaling flows for the IP MM call control based on SIP and SDP; stage 3 (Release 5)
3GPP TS 24.229 v7.11.0 (2008-03) IP multimedia call control protocol based on SIP (rel 7) and SDP, Stage 3 (Release 7)
3GPP TS 29.228 v7.4.0 (2006-12) IMS Cx and Dx interfaces, Signaling flows and (for Rel 7) message contents; (Release 7)

Etc…
RFC 3204 MIME media types for ISUP and QSIG Objects (PS) Upd by RFC3459, RFC5621
RFC 3261 SIP: Session Initiation Protocol (PS) Upd by RFC3265, RFC3853, RFC4320, RFC4916
RFC5393, RFC5621, RFC5626, RFC5630, RFC5922, RFC5954, RFC6026, RFC6141, RFC6665, RFC6878
RFC 3262 Reliability of Provisional Responses in SIP (PS)
RFC 3263 SIP: Locating SIP Servers (PS)
RFC 3265 SIP-Specific Event Notification (PS) Obsd by RFC6665, Upd by RFC5367, RFC5727, RFC6446
RFC 3310 HTTP Digest Authentication Using Authentication and Key Agreement (AKA) (Inf)
RFC 3311 SIP UPDATE Method (PS)
RFC 3312 Integration of Resource Management and SIP (PS) Upd by RFC4032, RFC5027
RFC 3313 Private SIP Extensions for Media Authorization (Inf)
RFC 3319 Dynamic Host Configuration Protocol (DHCPv6) Options for SIP Servers (PS)
RFC 3323 A Privacy Mechanism for the SIP (PS)
RFC 3325 Private Extensions to SIP for Asserted Identity within Trusted Networks (Inf) Upd by RFC5876
RFC 3326 The Reason Header Field for SIP (PS)
RFC 3327 SIP Extension Header Field for Registering Non-Adjacent Contacts (PS) Upd by RFC5626
RFC 3329 Security Mechanism Agreement for SIP (PS)
RFC 3361 Dynamic Host Configuration Protocol (DHCP-for-IPv4) Option for SIP Servers (PS)
RFC 3420 Internet Media Type message/sipfrag (PS)
RFC 3428 SIP Extension for Instant Messaging (PS)
RFC 3486 Compressing SIP (PS) Upd by RFC5049
SIP WG documents (2) from IETF 6.1.2014

RFC 3515 SIP Refer Method (PS)
RFC 3581 An Extension to SIP for Symmetric Response Routing (PS)
RFC 3608 SIP Extension Header Field for Service Route Discovery Dng Registration (PS) Upd by RFC5630
RFC 3840 Indicating User Agent Capabilities in SIP (PS)
RFC 3841 Caller Preferences for SIP (PS)
RFC 3853 S/MIME Advanced Encryption Standard (AES) Requirement for SIP (PS)
RFC 3891 SIP "Replaces" Header (PS)
RFC 3892 SIP Referred-By Mechanism (PS)
RFC 3893 SIP Authenticated Identity Body (AIB) Format (PS)
RFC 3903 SIP Extension for Event State Publication (PS)
RFC 3911 SIP "Join" Header (PS)
RFC 3968 The Internet Assigned Number Authority (IANA) Header Field Parameter Registry for SIP (BCP)
RFC 3969 IANA Uniform Resource Identifier (URI) Parameter Registry for SIP (BCP) Upd by RFC5727
RFC 4028 Session Timers in SIP (PS)
RFC 4032 Update to SIP Preconditions Framework (PS)
RFC 4168 The Stream Control Transmission Protocol (SCTP) as a Transport for SIP (PS)
RFC 4244 An Extension to SIP for Request History Information (PS)
RFC 4320 Actions Addressing Identified Issues with SIP Non-INVITE Transaction (PS)
RFC 4321 Problems Identified Associated with SIP Non-INVITE Transaction (Inf)
RFC 4412 Communications Resource Priority for SIP (PS)
SIP WG documents (3) from IETF 6.1.2014

RFC 4474 Enhancements for Authenticated Identity Management in SIP (PS)
RFC 4483 A Mechanism for Content Indirection in SIP Messages (PS)
RFC 4485 Guidelines for Authors of Extensions to SIP (Informational)
RFC 4488 Suppression of SIP REFER Method Implicit Subscription (PS)
RFC 4508 Conveying Feature Tags with SIP REFER Method (PS)
RFC 4538 Request Authorization through Dialog Identification in SIP (PS)
RFC 4780 Management Information Base for SIP (PS)
RFC 4916 Connected Identity in SIP (PS)
RFC 5079 Rejecting Anonymous Requests in SIP (PS)
RFC 5360 A Framework for Consent-Based Communications in SIP (PS)
RFC 5365 Multiple-Recipient MESSAGE Requests in SIP (PS)
RFC 5366 Conference Establishment Using Request-Contained Lists in SIP (PS)
RFC 5367 Subscriptions to Request-Contained Resource Lists in SIP (PS)
RFC 5368 Referring to Multiple Resources in SIP (PS)
RFC 5373 Requesting Answering Modes for SIP (PS)
RFC 5393 Addressing an Amplification Vulnerability in SIP Forking Proxies (PS)
RFC 5411 A Hitchhiker's Guide to SIP (Informational)
RFC 5478 IANA Registration of New SIP Resource-Priority Namespaces (PS)
RFC 5479 Requirements and Analysis of Media Security Management Protocols (Inf)
RFC 5621 Message Body Handling in SIP (PS)
RFC 5626 Managing Client-Initiated Connections in SIP (PS)
SIP WG documents (4) from IETF 6.1.2014

RFC 5627 Obtaining and Using Globally Routable User Agent URIs (GRUUs) in SIP (PS)
RFC 5630 The Use of the SIPS URI Scheme in SIP (PS)
RFC 5658 Addressing Record-Route Issues in SIP (PS)
   Using Datagram Transport Layer Security (DTLS) (PS)
RFC 5767 User-Agent-Driven Privacy Mechanism for SIP (Inf)
RFC 5768 Indicating Support for Interactive Connectivity Establishment (ICE) in SIP (PS)
   Event Package (PS)
RFC 5922 Domain Certificates in SIP (PS)
RFC 5923 Connection Reuse in SIP (PS)
RFC 5924 Extended Key Usage (EUK) for SIP X.509 Certificates (Experimental)
RFC 5954 Essential Correction for IPv6 ABNF and URI Comparison in RFC 3261 (PS)
RFC 6072 Certificate Management Service for SIP (PS)
RFC 6794 A Framework for SIP Session Policies 2012-12 (Proposed Standard)

== 71 RFCs over 11 years

First on the list from the year 2001 and last from 2012!

http://datatracker.ietf.org/wg/sip/
SIPCORE WG (9.1.2014)

draft-ietf-sipcore-rfc4244bis-12 An Extension to SIP for Request History Info
draft-ietf-sipcore-rfc4244bis-callflows-08 SIP History-Info Header Call Flow Examples(subm for Pub)
draft-ietf-sipcore-sip-websocket-10 The WebSocket Protocol as a Transport for SIP
draft-rosen-rph-reg-policy-01 Resource Priority Header (RPH) Registry Management Policy to IETF Review (Subm for Pub)

RFC 5839 An Extension to SIP Events for Conditional Event Notification (PS)
RFC 6026 Correct Transaction Handling for 2xx Responses to SIP INVITE Requests (PS)
RFC 6086 SIP INFO Method and Package Framework (PS)
RFC 6141 Re-INVITE and Target-Refresh Request Handling in SIP (PS)
RFC 6216 Example Call Flows Using SIP Security Mechanisms (Inf)
RFC 6223 Indication of Support for Keep-Alive (PS)
RFC 6228 SIP Response Code for Indication of Terminated Dialog (PS)
RFC 6442 Location Conveyance for SIP (PS)
RFC 6446 SIP Event Notification Extension for Notification Rate Control (PS)
RFC 6665 SIP-Specific Event Notification (PS)
RFC 6809 Mechanism to Indicate Support of Features and Capabilities in SIP (PS)
RFC 6878 IANA Registry for SIP "Priority" Header Field (PS)
SIP documents by SIMPLE WG (1) (6.1.2014)
draft-ietf-simple-chat-18 Multi-party Chat Using the Message Session Relay Protocol (MSRP)
RFC 3856 A Presence Event Package for SIP (PS) [year 2004]
RFC 3857 A Watcher Information Event Template-Package for SIP (PS)
RFC 3858 An Extensible Markup Language (XML) Based Format for Watcher Info (PS)
RFC 3994 Indication of Message Composition for Instant Messaging (PS)
RFC 4479 A Data Model for Presence (PS)
RFC 4480 RPID: Rich Presence Extensions to the Presence Info Data Format (PIDF) (PS)
RFC 4481 Timed Presence Extensions to the Presence Info Data Format (PIDF) to Indicate Status Information for Past and Future Time Intervals (PS)
RFC 4482 CIPID: Contact Info for the Presence Info Data Format (PS)
RFC 4660 Functional Description of Event Notification Filtering (PS) Upd by RFC6665
RFC 4661 An XML-Based Format for Event Notification Filtering (PS)
RFC 4662 SIP Event Notification Extension for Resource Lists (PS)
RFC 4825 XML Configuration Access Protocol (XCAP) (PS)
RFC 4826 XML Formats for Representing Resource Lists (PS)
RFC 4975 The Message Session Relay Protocol (MSRP) (PS)
RFC 4976 Relay Extensions for the Message Sessions Relay Protocol (MSRP) (PS)
SIP documents by SIMPLE WG (2) (6.1.2014)

RFC 5025 Presence Authorization Rules (PS)
RFC 5196 SIP User Agent Capability Extension to Presence Info Data Format (PIDF) (PS)
RFC 5261 XML Patch Operations Framework Utilizing XML Path Language (XPath) Selectors (PS)
RFC 5262 Presence Info Data Format (PIDF) Extension for Partial Presence (PS)
RFC 5263 SIP Extension for Partial Notification of Presence Information (PS)
RFC 5264 Publication of Partial Presence Information (PS)
RFC 5438 Instant Message Disposition Notification (IMDN) (PS)
RFC 5874 XML Document Format for Indicating a Change in XML Configuration Access Protocol (XCAP) Resources (PS)
RFC 6135 An Alternative Connection Model for the Message Session Relay Protocol (MSRP) (PS)
RFC 6714 Connection Establishment for Media Anchoring (CEMA) for the Message Session Relay Protocol (MSRP) (PS)
RFC 6914 SIMPLE Made Simple: An Overview of the IETF Specifications for Instant Messaging and Presence Using SIP (Inf)

Results (27 RFCs) between 2004 and 2013!

http://datatracker.ietf.org/wg/simple/
XCON – Centralized Conf WG (concluded) – documents 6.1.2014

RFC 4376 Requirements for Floor Control Protocols 2006-02 (Inf)
RFC 4582 The Binary Floor Control Protocol (BFCP) (PS)
RFC 4597 Conferencing Scenarios (Inf)
RFC 5018 Connection Establishment in BFCP (PS)
RFC 5239 A Framework for Centralized Conferencing (PS)
RFC 6501 Conference Information Data Model for Centralized Conferencing (XCON) (PS)
RFC 6502 Conference Event Package Data Format Extension for XCON (PS)
RFC 6503 Centralized Conferencing Manipulation Protocol (PS)
RFC 6504 Centralized Conferencing Manipulation Protocol (CCMP) Call Flow Examples 2012-03 (Inf)

Results (9 RFCs) between 2006 and 2012!

Sipping documents (1) – Concluded WG (9.01.2014)

RFC 3324 Short Term Requirements for Network Asserted Identity (Inf)
RFC 3351 User Requirements for SIP in Support of Deaf, Hard of Hearing and Speech-impaired (Inf)
RFC 3372 SIP for Telephones (SIP-T): Context and Architectures (BCP)
RFC 3398 ISDN User Part (ISUP) to SIP Mapping (PS)
RFC 3485 SIP and SDP Static Dictionary for SigComp (PS) Upd by RFC4896
RFC 3578 Mapping of ISUP Overlap Signalling to SIP (PS)
RFC 3665 SIP Basic Call Flow Examples (BCP)
RFC 3666 SIP Public Switched Telephone Network (PSTN) Call Flows (BCP)
RFC 3680 SIP Event Package for Registrations (PS) Upd by RFC6140
RFC 3702 Authentication, Authorization, and Accounting Requirements for SIP (Inf)
RFC 3725 BCP for Third Party Call Control (3pcc) in SIP (BCP)
RFC 3824 Using E.164 numbers with SIP (Inf)
RFC 3842 A Message Summary and Message Waiting Indication Event Package for SIP (PS)
RFC 3959 The Early Session Disposition Type for SIP (PS)
RFC 3960 Early Media and Ringing Tone Generation in SIP (Inf)
RFC 4083 Input 3GPP Release 5 Requirements on SIP (Inf)
RFC 4117 Transcoding Services Invocation in SIP Using Third Party Call Control (Inf)
RFC 4189 Requirements for End-to-Middle Security for SIP (Inf)
RFC 4235 An INVITE-Initiated Dialog Event Package for SIP (PS)
RFC 4245 High-Level Requirements for Tightly Coupled SIP Conferencing (Inf)
RFC 4353 A Framework for Conferencing with SIP (Inf)
Sipping documents(2) – Concluded WG (9.01.2014)

RFC 4411 Extending SIP Reason Header for Preemption Events (PS)
RFC 4453 Requirements for Consent-Based Communications in SIP (Inf)
RFC 4475 SIP Torture Test Messages (Inf)
RFC 4484 Trait-Based Authorization Requirements for SIP (Inf)
RFC 4497 Interworking between SIP and QSIG (BCP)
RFC 4569 IANA Registration of the Message Media Feature Tag (Inf)
RFC 4575 SIP Event Package for Conference State (PS)
RFC 4579 SIP Call Control - Conferencing for User Agents (BCP)
RFC 4596 Guidelines for Usage of SIP Caller Preferences Extension (Inf)
RFC 4730 SIP Event Package for Key Press Stimulus (KPML) (PS)
RFC 5039 SIP and Spam (Inf)
RFC 5057 Multiple Dialog Usages in SIP (Inf)
RFC 5118 SIP Torture Test Messages for Internet Protocol Version 6 (IPv6) (Inf)
RFC 5194 Framework for Real-Time Text over IP Using SIP (Inf)
RFC 5359 SIP Service Examples (BCP)
RFC 5361 A Document Format for Requesting Consent (PS)
RFC 5362 SIP Pending Additions Event Package (PS)
RFC 5363 Framework and Security Considerations for SIP URI-List Services (PS)
RFC 5364 XML Format Extension for Representing Copy Control Attributes in Resource Lists (PS)
RFC 5369 Framework for Transcoding with SIP (Inf)
RFC 5370 SIP Conference Bridge Transcoding Model (PS)
Sipping documents(3) – Concluded WG (9.01.2014)

RFC 5390 Requirements for Management of Overload in SIP (Inf)
RFC 5407 Example Call Flows of Race Conditions in SIP (BCP)
RFC 5589 SIP Call Control - Transfer (BCP)
RFC 5628 Registration Event Package Extension for SIP Globally Routable User Agent URIs (GRUUs) (PS)
RFC 5629 A Framework for Application Interaction in SIP (PS)
RFC 5850 A Call Control and Multi-Party Usage Framework for SIP (Inf)
RFC 5853 Requirements from SIP Session Border Control (SBC) Deployments (Inf)
RFC 5876 Updates to Asserted Identity in SIP (Inf)
RFC 5897 Identification of Communications Services in SIP (Inf)
RFC 6035 SIP Event Package for Voice Quality Reporting (PS)
RFC 6080 A Framework for SIP User Agent Profile Delivery (PS)
RFC 6157 IPv6 Transition in SIP (PS)
RFC 6795 SIP Event Package for Session-Specific Policies (PS)

[link to draft-drage-sipping-rfc3455bis-12] Private Header (P-Header) Extensions to SIP for 3GPP 2014-01-08

55 RFCs published between 2002….2012!
P2PSIP status (9.01.2014)

Active Internet-Drafts

- draft-ietf-p2psip-base-26 REsource LOcation And Discovery (RELOAD) Base Protocol (Subm for Pub)
- draft-ietf-p2psip-concepts-05 Concepts and Terminology for Peer to Peer SIP
- draft-ietf-p2psip-diagnostics-13 P2P Overlay Diagnostics
- draft-ietf-p2psip-disco-02 A RELOAD Usage for Distributed Conference Control (DisCo)
- draft-ietf-p2psip-drr-11 An Extension to REsource LOcation And Discovery (RELOAD) Protocol to Support Direct Response Routing
- draft-ietf-p2psip-rpr-11 An Extension to REsource LOcation And Discovery (RELOAD) Protocol to Support Relay Peer Routing
- draft-ietf-p2psip-self-tuning-09 Self-tuning Distributed Hash Table (DHT) for RELOAD
- draft-ietf-p2psip-service-discovery-09 Service Discovery Usage for RELOAD
- draft-ietf-p2psip-share-02 A Usage for Shared Resources in RELOAD (ShaRe)
- draft-ietf-p2psip-sip-11 A SIP Usage for RELOAD

P2PSIP tries to repeat what SKYPE has done.
Interest at the moment is low…!
## Summary of IETF documents on SIP and related WGs

<table>
<thead>
<tr>
<th>WG</th>
<th>RFCs</th>
<th>Years</th>
<th>Ends</th>
</tr>
</thead>
<tbody>
<tr>
<td>SIP</td>
<td>71</td>
<td>11</td>
<td>2012</td>
</tr>
<tr>
<td>SIPCore</td>
<td>12</td>
<td></td>
<td>ongoing</td>
</tr>
<tr>
<td>SIMPLE</td>
<td>27</td>
<td>9</td>
<td>2013</td>
</tr>
<tr>
<td>XCON</td>
<td>9</td>
<td>6</td>
<td>2012</td>
</tr>
<tr>
<td>SIPPING</td>
<td>55</td>
<td>10</td>
<td>2012</td>
</tr>
<tr>
<td>P2PSIP</td>
<td>--</td>
<td></td>
<td>ongoing</td>
</tr>
<tr>
<td>SUM</td>
<td>174</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Recent additions in IMS

- **VCC** – voice call continuity allows users to roam between IP-CAN/IMS (e.g. WLAN access) and Circuit switched domain without losing their call (24.206 v 7.4.0)
  - IP-CAN can be any IP access network based on cellular, WLAN, ADSL or other technology
- **Better NAT traversal** (SIP outbound and ICE)
- **IBCF** – Interconnection Border Control Function procedures. IBCF hides networks from each other, is an application level gateway (another term: session border controller)
- **Emergency Service**
- **PSTN/ISDN simulation services for NGN**
Summary of course scope

- SIP
- IP
- CAS, R2
- ISDN
- V5
- AN
- PABX
- Control Part of an Exchange Or Call Processing Server
- SIP or ISUP
- IP
- Diameter
- HLR/HSS
- MAP
- CCS7
- ISUP
- INAP
- SCP
- Media Gateway or Switching Fabric
- circuit
- packets
- Megaco/MGCP/…
SIP Requirements and fundamentals

- Part of IETF toolkit
  - Reusing other protocols & mechanisms: HTTP, etc.
  - Flexible
  - Extensible
- Moves intelligence to End System entities
  - End-to-end protocol
- Interoperability
- Scalability (although some state in network)
- Service creation easy
- URLs and Addresses are reused
- Same routing as SMTP
- Reuses infrastructure (all applications will use SIP entities for different services)
SIP overview – original ideas

- Simplicity
  - Text based (UTF-8 encoded) - simple tools for development but long messages
  - Lower call setup time than in H.323 (depends on sig ch speed)
  - basic protocol + extensions structure adopted
- Caller preferences, Ability to support many media types
- Runs over UDP or TCP or SCTP
- Used between both service and call control entities
- Has been adopted for 3G IP Multimedia signaling
- Originally subscriber signaling, in 3G also network to network signaling
- Development in 3GPP has added a lot of complexity
- A Market view: there are 3 important variants of SIP: 3G, IETF and Microsoft Messenger, + loads of open source implementations.
- Text based + transaction oriented → lengthy messages and flows
Sip Entities

- **User Agent**
  - Can act as a client and as a server

- **Servers (network based):**
  - **Redirect Servers**
    - Send back alternative location of the user (similar as HTTP servers)
  - **Proxy servers**
    - Act on behalf of client (forwards requests)
    - Forking proxies
  - **Registrars**
    - Accepts registrations and maps public SIP URIs to user locations.
  - **Location Servers (not part of SIP architecture)**
    - Gives back location of user (received from registrars)
    - E.g. HSS in 3GPP IMS architecture
    - Protocol between Location server and SIP server **not defined by SIP specs** (e.g. LDAP)
Addressing/Naming in SIP

• SIP URI – universal resource identifier = name
  – Globally unique (may point to a set of devices of the user)
  – Because we must name users, it was decided that SIP names must be resolved by SIP rather than DNS.
  – DNS usually resolves only names of servers. Servers do not move, while users do move. DNS may cache information → does not fit well with the idea of mobility.

• SIP has a registrar function in the network – P2P SIP removes the centralized registrar.
  – Translates SIP URI → address that usually points to a proxy that represents the UA
  – Registration has be distributed using a distributed hash table: take a hash of a name → node in a ring that sit in the same space as the hash values → log N -hops scalability where N is the nrof of nodes

• A Globally Routable UA URI (GRUU) is a SIP URI that has two properties:
  – It routes to a specific UA instance.
  – It can be successfully dereferenced by any user agent on the Internet, not just ones in the same domain or IP network as the UA instance to which the GRUU points.
  – 2 types: (1) GRUUs that expose the underlying AOR (2) GRUUs that hide the underlying AOR
Identification of users

- \texttt{sip:user@host[parameters][headers]}

- SIP URIs are like URLs, with prefix sip: which gives schema
  - \texttt{sip:joe.smith@hut.fi}
  - \texttt{sip:joe.smith@hut.fi?subject=Protocol}
  - \texttt{sip:sales@hotel.xy;geo.position:=48.54\_123.84\_120}

- Address must include host, other parameters are optional (username, port, etc…)

- Email-addresses can be reused

- “Click-to-call” on web-pages, MM messages, etc… are easily implemented
Identification of users in 3G IMS in R6

In Release 5 only one Private User id

NAI – Network Access Id (RFC 2486)

SIP URI (RFC 3261) or TEL URI (RFC 2806)

cmp IMSI in GSM

Private User Identity 1
username@operator.com

cmp. MSISDN in GSM

Public User Identity 1

Private User Identity 2
tel: +358-59-234-765

Public User Identity 2

Public User Identity 3

sip: +358-59-234-765@operator.com; user=phone

Public User Identity n

In Release 5 only one Private User id
Basic SIP call setup and release (IETF)

Caller

 registering

Proxy

INVITE

180 (ringing)

200 (OK)

ACK

Session Established

Callee

registering

Location Server

INVITE

180 (ringing)

200 (OK)

ACK

BYE

200 (OK)

Bye

Register

OK

Register

OK
”Basic call” Example (IETF)

• Caller sends INVITE
• Callee can accept, reject, forward the call
• If the callee accepts the call: responds with an optional provisional (1xx), and a final (≥200) response
• The caller confirms final response via ACK
• Conversation
• Caller or callee sends BYE
• BYE is acknowledged by 200 OK
• Low call setup times, post dial delay: 1.5 RTT!
• This flow is not Carrier Grade!
SIP messages have headers and a body

- Headers carry control information and are processed e.g. by Proxies
  - Usually protocols have a single header, SIP has many!
- Body can be e.g. SDP – session description protocol
  - end-to-end information (cmp H.245 in the H.323 framework) describing session requirements e.g. coding methods, etc
- Message delivery is transaction oriented= have request + reply: e.g. INVITE+200 OK
User Agent is split into User Agent Client (UAC) and User Agent Server (UAS)

- All Communication follows this transaction model, except 2xx and ACK.
- Server transactions filter incoming requests, absorbing retransmissions
- Only UAS can generate 2xx and only UAC can generate ACK (to success).
A Stateful Proxy can fork a transaction

Forking = multicast of INVITEs to N addresses
Redirect Server pushes processing to clients

1. Invite
2. Contacting
3. Precise Location
4. Address
5. ACK
6. Invite
7. OK
8. ACK

Redirect Server properties
- High capacity
- Minimal state overhead
- Service execution pushed to client

Caller
Redirect Server
Location Server (LDAP)
Callee
Stateful Proxy  vs  Stateless Proxy

- Maintains call context
- Repeats the actions of UAS/UAC to process requests and responses
- Call state and transaction state can be maintained
- Forking proxies require state
- TCP proxies must be stateful for reliability
- Enhanced services require state
- Can collect charging info
- All transactions in a session are processed by the same proxy (computer)

- No call context
- Response is not based on repeating UA actions
- Provides client anonymity
- Restricted gateway access
- High processing capacity
- Easier to replicate than the stateful proxy
- Consecutive transactions in a session can be processed on different computers → load sharing is easy to arrange for e.g. x00 million users
- Also semi-stateful is possible

UA = User Agent, UAC = UA Client
UAS = UA Server
When is stateful proxy necessary?

- On trust boundary: e.g., customer to operator or operator to operator
  - State allows to run sequences of actions for control of what the user is asking or what a remote user is trying to do
  - Allows to charge based on more than flat rate
  - Allows to hide a network from a competitor
  - Allows to switch between address spaces (realms) for signaling messages
- Advantages: reliable transport (TCP, SCTP) require state anyway, forking, intelligent services
- Was it a wise design choice to allow both stateless and stateful proxies?
  - One result is a rather high overhead for identifying a session
# List of SIP methods

<table>
<thead>
<tr>
<th>Method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ACK</td>
<td>Acknowledges the establishment of a session</td>
</tr>
<tr>
<td>BYE</td>
<td>Terminates a session</td>
</tr>
<tr>
<td>CANCEL</td>
<td>Cancels a pending request</td>
</tr>
<tr>
<td>INFO</td>
<td>Transports PSTN telephony signaling</td>
</tr>
<tr>
<td>INVITE*</td>
<td>Establishes a session</td>
</tr>
<tr>
<td>NOTIFY</td>
<td>Notifies a User Agent of a particular event</td>
</tr>
<tr>
<td>OPTIONS*</td>
<td>Queries a server about its capabilities</td>
</tr>
<tr>
<td>PRACK</td>
<td>Acknowledges a provisional response</td>
</tr>
<tr>
<td>PUBLISH*</td>
<td>Uploads information to a server</td>
</tr>
<tr>
<td>REGISTER</td>
<td>Maps a public URI with the current location of the user</td>
</tr>
<tr>
<td>SUBSCRIBE*</td>
<td>Requests to be notified about a particular event</td>
</tr>
<tr>
<td>UPDATE</td>
<td>Modifies some characteristic of a session</td>
</tr>
<tr>
<td>MESSAGE*</td>
<td>Carries an instant message</td>
</tr>
<tr>
<td>REFER*</td>
<td>Instructs a server to send a request</td>
</tr>
</tbody>
</table>

Blue methods* are candidates for AS processing includes both the base protocol and extensions from year 2004
Notes on SIP methods: INVITE

• Requests users to participate in a session. Body contains description of the session. If someone wants to modify the parameters of the session, he/she must re-INVITE carrying the modified parameters.

• One or more Provisional and one Final response are expected.

• **First line** in INVITE identifies the destination.
Notes on SIP methods: ACK

• Acknowledges the Final Response to INVITE even if INVITE was cancelled → result is 3-way handshake: INVITE-final-resp – ACK.
• Proxies can only ACK non-successful Final Resp.
• Purpose:
  – Lets the server know that session establishment was successful.
  – Forking may result in many final responses. Sending ACKs to every destination that sent a final response is essential to ensure working over UDP.
  – Allows sending INVITEs without session description. In this case the description is postponed to ACK.
• Has the same Cseq as the INVITE it acknowledges (see later for SIP headers).
Notes on SIP methods: CANCEL

• Purpose: to cancel pending transactions. Will be ignored by completed transactions = final response already sent.
  – useful for forking proxies. If one destination answered, the forking proxy can cancel all other pending INVITEs.

• Has the same Cseq as the request it cancels (see later for SIP headers).
Notes on SIP methods: REGISTER

• Purpose: to register the user’s current location.
  – This means that regular SIP needs a network based server to work. Need for other network based servers depends on existence of NAT’s, operator preferences etc.
  – There is an effort called Peer-to-Peer SIP that intends to replace Registrar by a distributed solution

• A user can be registered in several locations at the same time. Forking is used to find out where the user wants to answer the session invitation.

• A user can register from anywhere to his registrar → provides mobility → this is a major benefit of VOIP over CS in corporate context.
Notes on SIP methods: OPTIONS

• UA can query a server: which
  – methods and
  – extensions and
  – which session description protocols it supports.
  – which encoding for message bodies the server understands (e.g. compression to save bw).
Some SIP issues

- Parties can release the “call session” but since they have obtained each others’ IP-addresses, they can continue sending media streams to each other!!
- How to push INVITE to B-party, if B-party does not have a permanent IP address which is most often the case!
- Response messages (e.g. 180) are not reliably delivered. This may cause tear down of the call if it was initiated from ISDN
- If BYE is lost, Proxy does not know that call has ended
- Text based coding increases the signaling overhead → problem in Radio access

Integration of Proxy with Firewall and NAT, STUN, ICE SIP outbound

PRACK method

KeepAlive = re-INVITE mechanism

SigComp
Requests invoke SIP methods

- SIP methods are invoked on servers when requests arrive:
  - A REGISTER request sends location information of users to Registrars, registers with the location service
  - An INVITE request invites a user to participate in a session or conference
    - The message body contains a description of the session (usually SDP)
  - ACK requests are used to confirm responses for INVITE, for reliable message exchanges
  - CANCEL requests cancel the pending request of the session
  - BYE requests are used to terminate active sessions
    - Any party of the session can send it
  - OPTIONS requests are used to query information about servers' capabilities
  - PRACK requests are used to confirm provisional responses
SIP responses are classified by first digit

- HTTP look-alike
- Hierarchically organized three digit codes: status code - text associated with the code
- Provisional and final responses:
  - 1xx responses are informational messages e.g., 180 Ringing
  - 2xx response shows a successful transaction e.g., 200 OK
  - 3xx responses are redirect messages e.g., 301 Moved Permanently
  - 4xx responses indicate errors in requests e.g., 400 Bad Request
  - 5xx responses indicate server errors e.g., 500 Version not supported
  - 6xx responses indicate global failures e.g., 600 Busy everywhere
SIP Message Format

- **START-LINE**
  - SIP version used
  - In requests: address and method used
  - In responses: status code

- **HEADERS**
  - Information about call

- **BODY (payload)**
  - Usually SDP message

A Request line contains the Request URI = the name of the user that is the destination. This request URI is used for SIP routing while TO is used for local filtering!
C->S: INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142

v=0
o=UserA 2890844526 2890844526 IN IP4 here.com
s=Session SDP
c=IN IP4 pc33.atlanta.com
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
To and From header fields

• **To:** specifies the logical call destination
• **From:** specifies the logical call source
• **Present in all SIP messages**
• **Part of call identification**

```
INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301771
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142
```

NB: In IMS To and From header field are end to end information and not used nor asserted by the network. Instead the P-CSCF inserts the **P-Asserted-Identity** header field into the SIP messages indicating the network asserted identity of the originator of the call.
Call-ID and CSeq header fields

- **Call-ID**: It helps to uniquely identify a particular SIP dialog or registration
  - It helps to match requests and responses
  - It helps to detect duplicated messages

- **CSeq**: It is a number that uniquely identifies the transaction in a call

- **Present in all SIP messages**

```plaintext
INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142
```
Content-Type and Content-Length header fields

- **Content-Type**: It describes the media type of the message body
- **Content-Length**: The number of octets in the message body
  - It is mandatory in all SIP messages.

```
INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142
```
Max-Forwards

• **Max-Forwards field must be used with any SIP method**

• **It limits the number for proxies or gateways on the way of SIP message to the destination.**
  – i.e. is needed for message routing

```
INVITE sip:bob@biloxi.com SIP/2.0
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8
Max-Forwards: 70
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142
```
VIA header indicates path taken by the request so far

- Branch parameter is used to detect loops
- Contains transport protocol, client’s host name and possibly port number, and can contain other parameters
- Used to route responses back to requestor

```
INVITE sip:bob@192.0.2.4 SIP/2.0
Via: SIP/2.0/UDP server10.biloxi.com;branch=z9hG4bK4b43c2ff8.1
Via: SIP/2.0/UDP bigbox3.site3.atlanta.com;branch=z9hG4bK77ef4c2312983.1 ;received=192.0.2.2
Via: SIP/2.0/UDP pc33.atlanta.com;branch=z9hG4bKnashds8 ;received=192.0.2.1
Max-Forwards: 68
To: Bob <sip:bob@biloxi.com>
From: Alice <sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 314159 INVITE
Contact: <sip:alice@pc33.atlanta.com>
Content-Type: application/sdp
Content-Length: 142
```
Record-route and Route

- **Record-Route**: header is added by proxy, when proxy wants to stay in the route of all sip messaging
- **Route**: added by User Agent Client, after response came, with all Record-route headers in it (then UAC knows which relays want to stay on the signaling path)
- **NOT** the same as **Via** headers

Sample INVITE request with Record-Route headers:

```
INVITE sip:callee@u2.domain.com SIP/2.0
Contact: sip:caller@u1.example.com
Record-Route: <sip:p2.domain.com;lr>
Record-Route: <sip:p1.example.com;lr>
```

Sample BYE request with Route header:

```
BYE sip:callee@u2.domain.com SIP/2.0
Route: <sip:p1.example.com;lr>,<sip:p2.domain.com;lr>
```
”Basic Call” call flow

1) INVITE

2) 180 Ringing

3) 200 OK

4) ACK

5) BYE

6) 200 OK

UserA
UserA@atlanta.com
192.168.100.101

UserB
UserB@biloxi.com
192.168.200.201

Media stream
"Basic Call" call flow

UserA
UserA@atlanta.com
192.168.100.101

UserB
UserB@biloxi.com
192.168.200.201

1) INVITE

2) 180 Ringing

3) 200 OK

4) ACK

Media stream

5) BYE

6) 200 OK
"Basic Call” call flow

1) INVITE

2) 180 Ringing

3) 200 OK

4) ACK

Media stream

5) BYE

6) 200 OK
“Basic Call” call flow

UserA
UserA@atlanta.com
192.168.100.101

UserB
UserB@biloxi.com
192.168.200.201

1) INVITE

2) 180 Ringing

3) 200 OK

4) ACK

Media stream

5) BYE

6) 200 OK
”Basic Call” call flow

1) INVITE
2) 180 Ringing
3) 200 OK
4) ACK

Media stream

5) BYE
6) 200 OK
"Basic Call" call flow

UserA
UserA@atlanta.com
192.168.100.101

UserB
UserB@biloxi.com
192.168.200.201

1) INVITE
2) 180 Ringing
3) 200 OK
4) ACK

Media stream

5) BYE
6) 200 OK
SIP Extensions

• Needed to satisfy additional requirements
• Must conform to design rules
• SIP is not intended to solve every problem (another protocol might be used instead)
Feature Negotiation (OPTIONS)

- **Supported** features can be specified in requests and responses
  - **Supported** UAC and UAS tell features they support
- **Required** features can be specified in requests and responses
  - **Require** UAC tells UAS about required options
  - **Proxy-Require** required options for proxy/redirect servers
  - Many extensions use Require and Proxy-Require to specify their support
- New methods can be added without changing the protocol
  - server can respond with **405 Not Supported**
  - returns list of supported methods in **Allow** header
  - client can ask which methods are supported using OPTIONS
Reliable Provisional response in SIP

UAC

INVITE sip:uas@host SIP/2.0 supported 100rel

SIP/2.0 **180 Ringing** Require 100rel Rseq 223455

PRACK sip:uas@host SIP/2.0 Rack: 223455 1 INVITE

(retransmission of 180)

(retransmission of PRACK)

SIP/2.0 200 OK (for PRACK)

UAS

Retransmission algorithm starts

Retransmission algorithm stops

Tells that A-party (UAC) understands reliable responses to 1xx messages

Retransmission algorithm stops
QoS support - UPDATE

• Usage rule for 183-Session-Progress
  – If “a=qos” appeared in SDP, UAS sends 183 with “Session: qos” and SDP

• Additional Method - UPDATE
  – If “a=qos” appeared in SDP with “confirm” attribute, UAS/UAC sends UPDATE with success/failure status of each precondition.
  – 200 OK must acknowledge the UPDATE message
  – user B does not need to be prompted

• Additional Status Response - 580 Precondition Failure
  – If a mandatory precondition can’t be met, UAS terminates INVITE with this status response
Phone should not ring before QoS and Security are OK

SDP = Session Description Protocol (carried in SIP message body)

W/SDP = “a-qos:” strength direction
“a-secure:” strength direction
strength = mandatory|optional |success|failure
direction = send|recv|sendrecv

UPDATE confirms that preconditions are OK at the originator

PRACK method is used to ensure delivery of 183 and 180

Use case: 3G signaling!
SIP event notifications tell about remote significant events to the local party

Event-subscriber

SUBSCRIBE

200 (of SUBSCRIBE)

NOTIFY

200

NOTIFY

200

SUBSCRIBE Expires: 0 (unsubscribe)

200

Event-notifier

Generate immediate state response

Generate state change event

Use cases: automatic call-back, automatic buddy lists, message waiting indication, User's own registration status and “IN triggering”.

Raimo Kantola –S- 2015

Signaling Protocols

13 - 60
SIP MESSAGE provides Instant Messaging capability in Pager mode

Sender

MESSAGE sip:user2@domain.com SIP/2.0
Via: SIP/2.0/TCP user1pc.domain.com;branch=z9hG4bK776sgdkse
Max-Forwards: 70
From: sip:user1@domain.com;tag=49583
To: sip:user2@domain.com
Call-ID: asd88asd77a@1.2.3.4
CSeq: 1 MESSAGE
Content-Type: text/plain
Content-Length: 18

Watson, come here.

200 (of MESSAGE)

Destination

Note:
- body is MIME or S/MIME
- there is no dialog!

Use example:
- maps to SMS in GSM

Issue:
- authentication of sender and charging! Must use Proper security features
More SIP extensions

- **MESSAGE**
  - For instant messaging

- **INFO**
  - To transport mid-session information (very useful in SIP-PSTN gateways to carry all PSTN messages across SIP domains such that do not easily map to any other known SIP message)

- **Automatic configuration**
  - DHCP or Service Location Protocol (SLP)

- **Caller Preferences**
  - New headers: Accept-Contact, Reject-Contact, Request-Disposition (e.g. to express a preference for contacting the user at “fixed”, “business” connection).

- **REFER**
  - For session transfer (Refer-To: and Referred-By: )

- ...
Simplest deployment model: SIP island of users connected to ”PBX” from ISDN point of view

- SIP –server recognizes a numbering block, connects calls directly from IP-phone to IP-phone in the block
- Calls to all other numbers are routed to the gateway
- = SIP-server+Gateway are like a PBX
Signaling Efficiency using SIP

• This can be compared to ISDN signaling efficiency.
Text based Signaling in IMS produces a lot of bits to the air interface in a cellular network

This is based on Release 5 MO session setup procedures from 24.228
+ The MO flow will create as many air interface bits as talking for 18 seconds using AMR 4.75 codec.
+ About 70% of the bits are created by redundant lines of text.
+ There is more redundancy on information element level

NB: length of DSS1(L2+L3)
Call setup and release
< 1000 bits → SIP/SDP uses 100 times more bits
Zipping analysis of MO flow content shows that SIP/SDP carries a lot of context information in each transaction.

![Graph showing zipped lengths vs. SIP lengths]

- The MO flow is for 3G release 5.
- Lengths are in bytes. The dots are created by applying WinZip to all incrementally longer subflows of the MO flow. The curve shows that by keeping state, we would save quite a lot in signaling bits.
Signaling compression model

- **Originating SIP/SDP**
  - (Binary) Control Data
  - Text flow in UTF-8
  - Dictionary
- **SigComp**
  - Binary
- **Incoming Call control**
  - Binary?
  - Non std
- **Incoming SIP/SDP**
  - Text
- **SigComp**
  - Dictionary
  - Flow state

Raimo Kantola –S- 2015

Signaling Protocols
SigComp allows compression of Signaling Messages

• **NB:** *Signaling Compression* is different from *Header Compression*. Signaling Compression deals with the compression of the payload—signaling messages themselves and does not touch TCP/IP header (or RTP/UDP/IP header)!

• RFC 3320 and RFC 3321 specify a layer between the signaling transport and the signaling application

• Uses Global and User Specific Dictionaries to store state data over many SIP sessions
  – Dictionary is enlarged dynamically
  – Efficiency is improved by using all previous messages in a flow as source of the dictionary – incremental application of compression – this requires quite a lot of memory
  – After dictionary replacement, Huffman coding can be applied.

• Overall Compression/decompression architecture is based on a bytecode driven Universal Decompression Virtual Machine
  – Bytecode can be sent in SigComp messages by the Compressor
  – leaves a lot of detail for the implementor
SIP Summary

• SIP was introduced with the slogan of ”simple”
• Out-of-band idea adopted from ISDN era – why?
• Popularity → lots of people wanted to improve it and do many kinds of things with it
• Numerous WGs of IETF and 3GPP have made it quite complex → interoperability is a constant challenge
  – Like its predecessors has suffered from the controversy between need for fast and flexible methods for services differentiation and the need for control by the operators
  – Can still not provide PSTN like control by operators
• Text based (not ascii but UTF-8) + support for both stateful and stateless proxies → lengthy messages
• Has become the most important signaling protocol for IP based telephony and VOIP services → supports many kinds of services.
• NAT traversal took a long time to be specified → Room open for Skype
  – Still requires NAT traversal code to be integrated with SIP UA and scales poorly to battery powered devices