MASTER THESIS

QoS Estimation during Session Initiation of Video Streaming Session

under the direction of

Prof. Dr. Markuss Rupp
DI Michal Ries
Institut fur Nachrichtentechnik und Hochfrequenztechnik

handed in Technischen Universitat Wien
Fakultat fur Nachrichtentechnik und Hochfrequenztechnik

by

Iria Rodríguez
Escola Tècnica Superior
d´ Enginyeria de Telecomunicació de Barcelona
Universitat Politècnica de Catalunya

Matrikelnr: 0627627

Wien, July 2007

Executive Summary

Purposes of the work

The goal of this work is investigate the QoS provision mechanisms that can be used during the media session initialization, the translation of such mechanisms to Session Initiation Protocol an Session Description Protocol signalling and a physical analysis task over an IMS testbed, that give rise to the knowledge of the current IMS deployment in this field. The steps followed to achieve this aim are divided in two different parts. The first part comprises a research study of 3GPP technical specifications and reports, recommendations, proposals, Request for Comments and all the literature involved. The second part deals with the comparison evaluation of theoretical supported and available mechanisms, architectures or designs in front of the physical development of all this aspects. The main contribution of this work is, therefore, an accurate State of Art about IMS QoS provision and a lack collection in the available IMS test tools.

Organization of the work

The outlines of the work have its basis on a decisive an deep research in the meaningful IMS signalling plane aspects and the IMS QoS Provision, with the final merging of both issues by means of Session Initiation Protocol and Session Description Protocol mechanisms for deal with media sessions ensuring QoS. In parallel, by means of the traffic generated in ftw's testbed and its proper analysis, is obtained a physical approach to the IMS QoS management. The confrontation of the two parts of the work reveals, finally, the current lacks and limitations (also due to the restrictions of the testbed itself) and makes feasible a suggestion phase for a proposal that ends in a new configuration of existing elements. Closes the work the set of conclusions that previous phases give rise to, just as other outputs that serve as subject for a new project/research.

The final outline of the project and the chapters' organization is as follows:

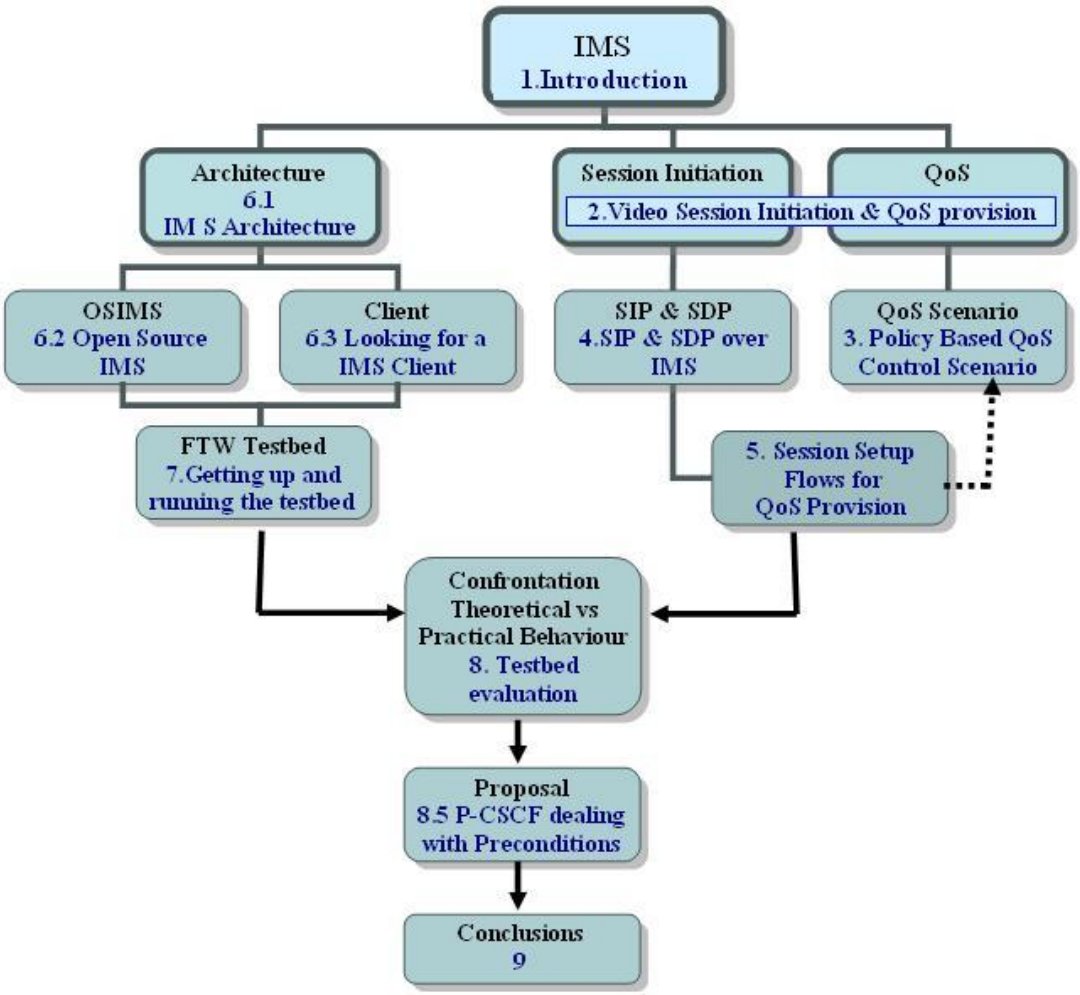


Figure 1: Outline of the work

Contents

List of Figures

List of Tables

Part I

Research Work

Chapter 1

Introduction

In the last years, an increasing demand for supporting real-time audiovisual services has been observed. The *IP Multimedia Subsystem* (IMS) provides all these services using the packet switched technology and one of the major requirements for achieve a successful and wide deployment is the efficient transmission of sensitive content (audio, video, image) over the network, a real challenge due to stringent delay, jitter and packet loss requirements. Taking this features into account, it becomes necessary provide *Quality of Service* (QoS) for media streams and, even more important and often unlikely correlated with this, achieve an acceptable user experience.

The current introduction chapter is organized as follows: First of all the background concepts, context and trends will be presented. Next, a more concrete overview of the involved issues will be discussed: Perceptual QoS, Session Initialization requirements and IMS features related with media streaming (Media Authorization, Resource Reservation), just as the codecs chosen in the IMS for video streaming. Finally, some important aspects related with QoS will be highlighted and IMS End-To-End QoS models presented.

1.1 Outline: Understanding Concepts

Before to start the technical overview in depth it is necessary to define some terms and try to answer some questions. The first one would be What? And it can be answered following the title on head "Quality estimation during session initiation of video streaming session". Three important concepts must be gathered from it:

Video is the media type we are focused on and its delivery over the packet-switched networks by means of media sessions performs an important *raison d'être* of the IMS. According to the *3rd Generation Partnership Project* (3GPP) *Technical Specification* (TS) 22.105 [1] multimedia services are services that handle several types of media, some of witch, need to be synchronised between them. A multimedia service may involve multiple parties, multiple connections, and the addition or deletion of resources and users within a single call.

Regarding the previous definition, one realizes that an important requirement of media delivering becomes the **synchronization**; it is there where the need of **streaming** takes place, since it refers to the ability of an application to play synchronised media streams continuously received by (approximately at the playout rate) and normally displayed to (as soon as the first few bytes of the stream arrive) the end-user, whilst it is being delivered by the provider over a data network, an infrastructure that were not designed with streaming in mind. Such ability frees up terminal memory but also allows media to sent live to clients as the media event happens [2]. Streaming media requires then that data be transmitted from a server to a client at a sustained bit rate that is high enough to maintain continuous and real-time sessions at the receiving client station. In order to achieve this, streaming technology involves audio and video compression, schemes for stream formatting and transmission packetization, networking protocols and routing, client designs for displaying and synchronizing different media streams, and server designs for content storage and delivery.

Streaming media over a packet switched network becomes not so easy since the bandwith is limited, delays and jitter appears and, finally, packets and data are lost. In such context it's necessary to establish QoS requirements. According again with 3GPP TS 22.105 [1], *Quality of Service* (QoS) is the collective effect of service performances which determine the degree of satisfaction of a user of a service. It is characterised by the combined aspects of performance factors applicable to all services, such as: service operability performance, service accessibility performance, service retention performance, service integrity performance, and other factors specific to each service.

Why IMS for video streaming services? One more time, the QoS discussed previously. IMS allows synchronizing session establishment with QoS provision, and by this way real time multimedia sessions. From a provider point of view, it allows charging multimedia sessions appropriately, giving rise to differentiated business models. Also, it's an important feature the integration of different services making possible a multi-vendor services creation industry and providing all the services

that nowadays Internet provides. Provision that is doing using the packet switched technology (more efficient than the circuit switched one) and creating a service environment [3].

Why a growing interest on video streaming services? Multimedia streaming are receiving considerable interest in the mobile network business since supporting reliable real-time services is a decisive factor for the increasing migration to the packet-based mobile networks. The issue of QoS emerged, exactly, with this move away from circuit switched networks towards packet switched or *Internet Protocol* (IP) networks, due to a important difference between them: in the latter, the resources are not explicitly allocated to individual pairs of hosts, the available network capacity is shared giving rise to possible packet losses, and as a last resort, to degradations in the end-user experienced and perceived quality. QoS appears in order to ensure the same reliable and stable level of quality users have been used to in previous generations [4].

1.2 Perceptual QoS

Usually, from a strict (or maybe customers) point of view, the sense of QoS is to ensure minimum requirements for bandwidth, delay, jitter and packet loss probably (parameters defined by the *European Telecommunications Standards Institute*, ETSI), but they do not reflect the end user quality. Defined parameters do not fit subjective media perception; they only take into account technical features just useful for network performance evaluation. One step forward is taken when signal processing algorithms take part: bandwidth efficient codecs which are capable of dealing with packet losses; concealment algorithms which compensates for lost packets, just as jitter buffer algorithms.

Of course, that which has a significant impact on the overall performance of a service is the quality perception of the user, however, it is often very difficult to extract a clear correlation between the data transport QoS (in terms of the most important objective transmission parameters) and the subjective perceptual quality evaluation. It makes necessary develop Subjective Quality Estimation Methods.

Nowadays, the focus of QoS research is increasing IP network reliability and availability in order to reach the very high level of circuit switched networks, since both concepts have always been crucial factors determining system usability and users satisfaction [4].

1.3 Multimedia Streaming Protocols

A multimedia session requires that three main aspects be considered: a session control, a session description and a media transport. Let's see by means of which protocols they are achieved in the IMS.

1.3.1 Session Control: SIP

Session Initiation Protocol (SIP, RFC 3261[5]), was originally proposed by the *Internet Engineering Task Force* (IETF) like an application-layer (probably more correct control plane, corresponding to the three-plane telecommunication architecture model) protocol to establish, modify and terminate multimedia sessions in all-IP networks. In November 2000, SIP was accepted as a 3GPP signalling protocol, later, it was chosen as the session control protocol for the IMS.

SIP is not a standalone integrated solution for multimedia sessions in a communications system. SIP is rather a component that can be used (should be used) in conjunction with other protocols to build a complete multimedia architecture able to provide a broad range of services for users, with special importance, audio and video communications. However, SIP works independently of any of these protocols and without dependency on the type of session that is being established.

SIP follows the client-server model (a SIP message is either a request from a client to a server, or a response from a server to a client) and its design principles were borrowed from the *Simple Mail Transfer Protocol* (SMTP, RFC 2821 [6]) and especially from the *HyperText Transfer Protocol* (HTTP, RFC 2616 [7]), this feature allows SIP service developers to take advantage of all the frameworks developed for HTTP. Like a text-based protocol, SIP becomes easier to extend, debug and be used to build services. It is important to remind that SIP does not provide services. Rather, SIP provides primitives that can be used to implement different services.

1.3.2 Session Description: SDP

Session Description Protocol (SDP), published by the IETF as RFC 4566 [8], is a format for describing multimedia sessions and its streaming initialization parameters for the purposes of session announcement, session invitation, and other forms of multimedia session initiation.

The media and transport information included in an SDP session description is the following:

- The type of media (video, audio, etc.)
- The transport protocol (*Real-time Transport Protocol* (RTP) / *User Datagram Protocol* (UDP) / *Internet Protocol* (IP), H.320, etc.)
- The format of the media (H.261 video, *Moving Picture Experts Group* (MPEG) video, etc.)

In addition, SDP conveys address and port details, this address and port are the destination address and destination port of the multicast stream, whether being sent, received, or both. Furthermore, information about the bandwidth to be used and contact information for the person responsible for the session are an additional information also desirable, due to the fact that resources necessary to participate in a session may be limited.

By means of a short structured textual description, SDP is able to meet an important video streaming requirement that is to convey the previous mentioned parameters (media details, transport addresses and other session description metadata) to the participants, since they allow them to gather the sufficient information to discover and participate in a multimedia session.

SDP provides a standard representation for such information; regardless of the transport protocol used (including, among others, *Session Announcement Protocol* (SAP), SIP, *Real Time Streaming Protocol* (RTSP), electronic mail using the *Multipurpose Internet Mail Extensions* (MIME) extensions, and the HTTP). It is important keep in mind that SDP is not intended to support negotiation of session content or media encodings: this is viewed as outside the scope of session description.

SDP also allows to group media streams and describes the semantic of the group [3]: *Lip Synchronization*(LS) semantics indicate that the play-out of the media streams need to be synchronized. *Single Reservation Flow* (SRF) semantics indicates that all the streams in the group should use the same resource reservation flow (the same *Packet Data Protocol* (PDP) Context)

1.3.3 Media Transport: RTP and RTCP

Finally, a media session requires a protocol responsible for transporting media, in this case, the protocols involved are the *Real-time Transport Protocol* (RTP) and the *Real-time Transport Control Protocol* (RTCP). RTP (RFC 3550 [9]) allows real-time media transport, playing out the media at the proper time over unreliable transports (UDP, *Datagram Congestion Control Protocol* (DCCP)) and

jitter presence, that is, when the data is being transported by the network without keeping timing relationship. The arrival times of the packets can not be used to establish a relationship between the media streams, such constraint is resolved by means of time stamps with witch the receivers can recover the data timing order.

Receivers use buffers (where they place, according to the time-stamps, the RTP packets) and interpolate techniques (to fill possible gaps, if a packet have not been received in time). There is a commitment between delay and quality so an important challenge is decide appropriately when to start playing media to the user; if the receiver starts as soon as the first packet is received is possible that some packets do not arrive when they have to be played, on the other hand, waiting before playing the media would rise in a difficult maintenance of a normal conversation.

RTP packets also carry sequence numbers, binary sender identifiers and the payload type (to identify the current speaker and the encoding and transport format of the data carried in the RTP packet respectively.

RTCP [9] [10] is always used together with RTP to monitor QoS and convey information about the participants in an ongoing session (media synchronization and mapping between RTP binary sender identifiers and humanreadable names). To develop QoS statistics it implemets a feedback on quality of reception of data by means of RTCP reports: senders report the number of packets they sent and receivers report number of packets received. RTCP also provides a mapping between RTP timestamps and a wall clock, allowing synchronization of different media streams, useful for example in lip synch (audio-video synchronization).

When SDP is used, RTP packets are normally sent to a port with an even number and RTCP messages are sent to the consecutive uneven port [3].

1.4 Video Streaming Codecs over IMS

The 3GPP TS 26.114 [11]) gathers the supported video codecs for IMS terminals. Just as such document shows, *Multimedia Telephony Service for IMS* (MTSI) terminals offering video communication shall support the *International Telecommunication Union - Telecommunication Standardization Sector* (ITU-T) Recommendation H.263 Profile 0 Level 45. In addition they should support the ITU-T Recommendation H.263 Profile 3 Level 45; MPEG-4 (Part 2) Visual Simple Profile Level 3 (with some constraints); and the ITU-T Recommendation H.264/MPEG-4 (Part 10) *Advanced Video Coding* (AVC) Baseline Profile Level 1b without requirements on output timing conformance.

1.4.1 H.263

The H.263 [12], was created by the ITU (more precisely in the domain of the ITU-T *Video Coding Experts Group* -VCEG-) in 1995/1996 as a low-bitrate video compressed format standard for videoconferencing and videotelephony applications. H.263 was developed to stream video at bandwidths as low as 20 kbps to 24 kbps and was based on the H.261 codec, the previous ITU-T standard for video compression. H.263 has largely replaced H.261 due to the fact that it requires, as a general rule, half the bandwidth to achieve the same video quality.

The coding algorithm of H.263 is similar to that used by H.261, but it has been enhanced with some improvements and changes to improve performance and error recovery. Half pixel precision is used for motion compensation whereas H.261 used full pixel precision and a loop filter. Some parts of the hierarchical structure of the datastream are now optional, so the codec can be configured for a lower datarate or better error recovery. There are now four optional negotiable options included to improve performance: Unrestricted Motion Vectors, Syntax-based arithmetic coding, Advance prediction, and forward and backward frame prediction similar to MPEG called P-B frames:

- **Unrestricted Motion Vector mode (UMV)**. In this mode motion vectors are allowed to point outside the picture. The edge pixels are used as prediction for the "not existing" pixels. With this mode a significant gain is achieved if there is movement along the edge of the pictures, especially for the smaller picture formats. Additionally, this mode includes an extension of the motion vector range so that larger motion vectors can be used. This is especially useful in case of camera movement.

- **Advanced Prediction mode**. This option means that *Overlapped Block Motion Compensation* (OBMC) is used for the P-frames. Four 8x8 vectors instead of one 16x16 vector are used for some of the macro blocks in the picture, and motion vectors are allowed to point outside the picture as in the UMV mode above. The encoder has to decide which type of vectors to use. Four vectors use more bits, but give better prediction. The use of this mode generally gives a considerable improvement, especially because (OBMC) results in less blocking artifacts.

- **Syntax-based Arithmetic Coding mode**. In this mode arithmetic coding is used instead of *Variable Length Codes* (VLC) coding. The *Signal to Noise Ratio* (SNR) and reconstructed frames will be the same, but generally fewer bits will be produced. The average gain for inter frames is 3-4%. This gain depends on the sequence, the bit rate and other options used. For intra blocks and frames, the gain is higher, on average about 10%.

- **PB-frames mode.** A PB-frame consists of two pictures being coded as one unit. The name PB comes from the name of picture types in MPEG where there are P-pictures and B-pictures. Thus a PB-frame consists of one P-picture which is predicted from the last decoded P-picture and one B-picture which is predicted from both the last decoded P-picture and the P-picture currently being decoded. This last picture is called a B-picture, because parts of it may be bi-directionally predicted from the past and future P-pictures. For relatively simple sequences, the frame rate can be doubled with this mode without increasing the bit rate much. For sequences with a lot of motion, PB-frames do not work as well as B-pictures in MPEG. This is because there are no separate bi-directional vectors in H.263, the forward vectors for the P-picture is scaled and added to a small delta-vector. The advantage over MPEG is much less overhead for the B-picture part, which is really useful for the low bit rates and relatively simple sequences most often generated by video phones.

In addition to Quarter-CIF and CIF (*Common Intermediate Format*) that were supported by H.261, the source coder can operate on three more standardized video source formats: sub-QCIF (SQCIF), 4CIF and 16CIF, and can also operate using a broad range of custom video formats. SQCIF is approximately half the resolution of QCIF. 4CIF and 16CIF are 4 and 16 times the resolution of CIF respectively. The support of 4CIF and 16CIF means the codec could then compete with other higher bitrate video coding standards such as the MPEG standards.

1.4.2 MPEG-4 (Part 2) Simple Profile

Moving Picture Experts Group 4 (MPEG-4) [13] provides a generic tool set for the transmission and storage of audiovisuals signals for numerous applications. Hence, MPEG-4 has to deal with a big number of natural and synthetic audio and video coding schemes being a global multimedia language. MPEG-4 also achieves to maintain a useful quality of service within error-prone channels introducing new concepts of object-based user interactivity as well as a rich feature set.

MPEG-4 video coding is a block-based predictive differential video coding scheme, such as MPEG and H.261/H.263, that uses the following techniques in order to perform the data compression:

- Divide the picture into 8x8-blocks or 16x16-macroblocks
- Transform coding with *Discrete Cosine Transform* (DCT)
- Motion compensated prediction

- Quantization and Run length and Huffman coding for VLC.

The video compression technology developed by MPEG is included concretely in MPEG-4 Part 2, that belongs to the MPEG-4 *International Organization for Standardization / International Electrotechnical Commission* (ISO/IEC) 14496-2 standard, and it is a DCT standard (similar to MPEG-1 and MPEG-2). Simple Profile is similar to H.263 and is aimed for use in low bit rate and low resolution situations due to the network bandwidth or the device size, among others. Some examples are cell phones, some low end video conferencing systems, surveillance systems, etc.

1.4.3 H.264 / MPEG-4 Part 10 AVC

The next enhanced video codec developed by ITU-T VCEG (in partnership with MPEG) after H.263 is the H.264 standard, also known as *Advanced Video Coding* AVC and MPEG-4 part 10 [14].

Both H.263 and H.264 are known primarily as video codecs designed for low-latency video conferencing applications. But H.264 appears a decade after than H.263 as its successor, providing some advantages. Technology advancements allow H.264 be able to provide much higher quality across the entire bandwidth spectrum.

The basis of H.264 is similar to the adopted in previous standards such as H.263 and MPEG-4, and consists of the following four main stages:

- Dividing each video frame into blocks of pixels so that processing of the video frame can be conducted at the block level.

- Exploiting the spatial redundancies that exist within the video frame by coding some of the original blocks through spatial prediction, transform, quantization and entropy coding (or VLC).

- Exploiting the temporal dependencies that exist between blocks in successive frames, so that only changes between successive frames need to be encoded. This is accomplished by using motion estimation and compensation. For any given block, a search is performed in the previously coded one or more frames or in a future frame to determine the motion vectors that are then used by the encoder and the decoder to predict the subject block.

- Exploiting any remaining spatial redundancies that exist within the video frame by coding the residual blocks, i.e., the difference between the original

blocks and the corresponding predicted blocks, again through transform, quantization and entropy coding.

1.5 IMS Media Authorization

IMS performs the media authorization by means of an offer/answer exchange between two user agents that is acceptable to the *Proxy-Call State Control Function* (P-CSCF) and the *Serving-Call State Control Function* (S-CSCF) , allowing the media exchange. The user's media traffic will be authorized as long as it complies with the previously authorized offer/answer exchange. If the offer/answer exchange is performed to establish an audio stream, video traffic exchange will not be authorized [3].

IMS uses *Common Open Policy Service* (COPS) protocol between the *Policy Decision Function* (PDF) and the *Gateway GPRS Support Node* GGSN in order to control the *General Packet Radio Service* (GPRS) resources, assigning them by policies that take into account quality objectives.

3GPP provides a mechanism for the network to authorize media streams. *Service Based Local Policy* (SBLP): a mechanism based on the network inserting a media authorization token that IMS terminals return to the network when requesting the establishment of a media stream. SBLP allows an interaction between the control and the user plane [3].

The complete Policy-Based Media Authorization scenario and mechanisms over IMS are detailed in chapter 3. Media Authorization is also overviewed in 4.2.2 and appendix A.4.2 at the level of SIP signalling extensions.

1.6 End-to-End QoS Models: Resource Reservation

Assuming that provide QoS becomes an unquestionable requirement, and besides allowing media authorization, it's also required by the terminals be able to map media streams of a session into resource reservations flows.

The IMS supports several end-to-end QoS models: Terminals may use link-layer resource reservation protocols (PDP Context Activation), *Resource ReSerVation Protocol* (RSVP), or *Differentiated Services* (DiffServ) codes directly; while networks use DiffServ and may use RSVP [3].

1.6.1 Terminals Resource Reservation: PDP Context

A *Packet Data Protocol* (PDP) Context is a logical association between a Mobile Station and a Public Data Network running across a GPRS network [15]. The context defines aspects such as routing (terminal's IP address), QoS (including the traffic class: best effort -background-, interactive, streaming and conversational), security, billing etc. The user has to activate an *Internet Protocol version 6* (IPv6) PDP context to access IMS and registrate to the P-CSCF (i.e. registrate to the IMS SIP server) using an IPv6 address. By this way, it exchange SIP signalling right after performing a GPRS attach. Secondary PDP context are established by the terminals to send and receive media, with the same IP address as the primary PDP context but with particular (may be different) QoS characteristics. The number of this additional PDP context depends on the SRF information received in session descriptions (from the P-CSCF)[3].

More details about PDP Context Activation are gathered in 2.4.3 and within the Policy-Based Media Authorization mechanisms summarized in chapter 4.

1.6.2 The Integrated Services Architecture: RSVP

Integrated Services (IntServ) ensures network wide per flow QoS by means of per flow Bandwidth reservations. For each new flow available network resources are checked and admission controls performed. A new enter is allowed if the current QoS of all others flows is preserved and if the required QoS will be provisioned. It relies on *Resource Reservation Protocol* (RSVP [16]). The problem here is the overhead in routers (signaling, memory, flow management) that limits the scalability of the IntServ and makes it only recommendable with a limited number of flows [4].

1.6.3 The Differential Service Architecture

On the other hand, *Differentiated Services* (DiffServ) architecture ensures QoS at the level of the entire Traffic aggregates (not individual flows). In order to do it different traffic classes are defined. The classes are Conversational/ Streaming/ Interactive/ Background, where the conversational is the most delay sensitive. Each class has a Different Level of QoS required. It uses also *Connection Admission Control* (CAC) in order to solve congestion problems [4].

Chapter 2

Video Session Initiation & QoS

The present chapter attempts to overview the IMS video streaming session initialization highlighting the IMS options for quality management during this first stage. The contents are organized as follows:

First of all, video session QoS requirements will be presented, just as the more widespread used *Key Performance Indicators* (KPI), basic toll when the provision of *Quality of Experience* (QoE) is in the spotlight. Once the requirements have been presented, in section 2 general issues of end-to-end QoS will be overviewed with special attention on IMS session signalling and resource allocation processes previous of a session setup request. Section 3 introduces QoS concept and architecture in *Universal Mobile Telecommunications System* (UMTS), like basic IMS access network. The layered architecture of a UMTS bearer service is depicted just as the UMTS Bearer Service (BS) attributes that perform the QoS profile are listed.

After clarify video session requirements, QoS issues, concept and architecture the last section gathers basic session setup procedures for an ongoing multimedia (video) session: codec negotiation and preconditions based on SIP/SDP signalling, PDP context activation at the UMTS BS level and RSVP like resource reservation protocol at the IP BS Level.

2.1 Video Session QoS Requirements

When video session is considered two different communication schemes apply: conversation and streams. Video conferencing applications (like other new multimedia applications) require the conversation scheme, traditionally known associated to telephony speech. In other video applications (less stringent in terms of QoS) such one way video tools the scheme stream is used (see Tables 2.2 and 2.3). The performance requirements for streaming services are gathered in 3GPP TS 22.105 [1].

Performance requirements for conversation scheme can be summarized in an overall requirement that is support conversational real time services with low transfer delay, that is, the time relation (variation) between information entities of the stream shall be preserved although there are less hard requirements on packet loss ratio; such features are strictly given by human perception scheme, being this the strongest and most stringent scheme in terms of QoS requirements. On the one hand, the human eye is tolerant to some loss of information so that some degree of packet loss is acceptable depending on the specific video coder and amount of error protection used; on the other hand the human ear is highly intolerant of short-term delay variation (jitter).

In videophone applications, video and audio streams are carried full-duplex in a conversational environment, therefore the delay requirements applied should be the same that conversational voice has (see Table 2.1), the problem here is given by the fact that audio and video must be synchronized in order to provide ̈lip-syncḧ, giving rise to limitations on the achievement of such requirements since they rely on video codecs used.

0 to 150 ms	**Preferred Range** <30ms, user does not notice any dellay at all. <100ms, user does not notice any dellay at all if echo cancellation is provided and there are no distorsions on the link.
150 to 400 ms	**Acceptable Range** (but with increasing degradation)
above 400 ms	**Uncceptable Range**

Table 2.1: ITU-T Recommendation G.114. General limits for one-way transmission time for conversational voice

Error Tolerant	Conversational voice and video	Voice Messaging	Sharing audio and video	Fax
Error Intolerant	Telnet interactive games	E-commerce, www browsing	FTP, still image, paging	E-mail arrival notification
Class	conversational (delay <<1sec)	Interactive (delay aprox 1sec)	Streaming (delay < 10sec)	Background (delay > 10sec)

Table 2.2: Summary: applications and associated communications schemes

Scheme for video media		**Conversational real-time services**	**Streaming services***
Application		Videophone	Movie clips, surveillance, real-time video
Degree of symetry		Two-way	Primarly one-way
Data rate		32-384 kbps	20-384 kbps
Key performance parameters and target values	**Start-up delay**	<150msec pref <400msec limit <100msec Lip-sync	<10sec
	Transport delay variation		<2sec
	Packet loss at session layer	<1%FER	<2% Packet loss ratio
*Bit rates 63 kb/s - 128 kb/s video streaming 3GPP codec. Content bit rate adaptation improves quality. Breaks in the connection due to mobility <3s - 5s and small bit rate variations.			

Table 2.3: End-user performance expectations

As it has been mentioned previously, when the service behind an IMS session consists on one way video delivering the scheme streams applies. The fundamental characteristics for QoS of such scheme consist on a real time one way (unidirectional) and continuous data flow always aiming at a live (human) destination and also the preservation of time relation between information entities of the stream in terms of jitter; the highest acceptable delay variation over the transmission media is given by the capability of the time alignment function of the application, that allows it an acceptable jitter greater than that limited by the human perception. The main distinguishing feature of one-way video is that there is no conversational element involved, meaning that the delay requirement will not be so stringent, and can follow that of streaming audio.

2.1.1 Key Performance Indicators

Network management requires real-time monitoring of current network status and performance and the ability to take prompt action to control the performance and resources of the network when necessary. In this way, operators can be able to maintain satisfactory levels of quality, providing a more predictable and reliable service perception.

Customer demand	Indicators	Mesures
Service Accessibility	Availabilty and coverage Access Success Rate Service Access Delay	Eo/No, RSCP Admission Control Service Setup Attach, PDP context activation, IP service setup
Service Integrity	Video Quality Audio Quality Web page download time E-mail sending time, etc.	BLER FER throughput delay, jitter
Service Retainability	Dropped Data Connection Connection Timeouts	Handover Failure No coverage, etc. Dropped PDP Contex/Attach

Table 2.4: KPI and mesures according to customers' demand

The network measurements involved are based on correctly defined KPIs for each service, since each of them will have different QoS requirements The most important challenge is to identify what among this measures are meaningful in terms of broader business objectives. On the other hand the increased use of KPI will make service and network provisioning more transparent to the customers, being feasible a better comparison between offers of different providers. Inadequate performance indicators and monitoring may give rise to hidden problems in network performance and user perceived QoS.

Three KPI categories related to customers' demands can be established: service accessibility, integrity and retainability. Table 2.4 summarizes the main KPI and their associated measures over a packet switched domain.

2.2 General issues of end-to-end QoS

End-to-End QoS Signalling and Resource-Allocation should be enforced for video sessions over IMS so they require, just as the first section show, QoS better than traditionally best-effort services or the background QoS Class. Then, it is indispensable considerer the need of end-to-end QoS indication, negotiation and resource allocation during the session setup in the IM *Core Network* (CN).

Guaranteed end-to-end QoS provision is only possible if all backbone and access networks on the path provide QoS guarantees. So all the network administrative domains in the path of the IP flow may need to include functionalities that make them able to receive and react to, process (provide IP flow admission control), and convey per IP flow/flow aggregate/service aggregate QoS information, although each administrative domain can use different QoS provisioning techniques for realizing the mentioned functionalities.

In any case, the network provides the information about the available QoS that can be guaranteed to the *User Equipment* (UE), which makes the decision to request a service which requires guaranteed end-to-end QoS. Even if the desired QoS cannot be guaranteed temporarily or QoS cannot be guaranteed at all the session can be established. Is under the UE responsibility to make the final decision whether to continue with such establishment or not.

2.2.1 Resource Check and IMS session setup

The IMS session signalling and the allocation of resources are clearly separated during the IMS session setup [17], which is started and afterwards set on hold

allowing both endpoints requesting at this time the required resources they are responsible for, at least in their access network. Note that the initiation of any required end-to-end QoS signalling, negotiation and resource allocation processes at different network segments shall take place after the initiation and delivery of a session setup request.

As it has been introduced few lines above, to establish if the resources that were granted by the network are sufficient for an application session is under UE responsibility, that can either accept insufficient QoS or may try to achieve the desired QoS at a later point in time, as long as only resources of the access network are taken into account. In the case of end-to-end resources, those can be guaranteed by a backbone network, but only statistically granted. It is also possible that feedback at all from a backbone network does not be present on the end-to-end path. Cases like that is not possible to receive guaranteed external resources at all or that QoS becomes insufficient during the IMS session force the UE to be able to handle such situations.

There are situations when even guaranteed resource in the backbone network or in the access network can be redrawn or made unavailable due to many events, e.g. network failure and urgent network resource re-allocation. In such situations, the network needs to provide the necessary network information to the decision points (network and/or UE) in order that reactive measures can be taken and that the application session is handled appropriately, in line with session policy and user wishes.

For the general end-to-end path a number of possibilities exist at which point in time and under which responsibility the external resources are requested. The external resource request may be coupled with the UMTS internal resource request, i.e. with the PDP context establishment. Both endpoints may be responsible for the resource request for the backbone network. Resources may either be requested by one of the endpoints for both directions or by both endpoints in either sending or receiving direction.

In establishing an IMS session [18], it becomes necessary the co-ordination between Session Control and QoS Signalling/Resource Allocation in order to allow the following options shall be possible:

- For an application, to request that the resources needed for bearer establishment be successfully allocated before the destination user is alerted.
- Dependent on the application being offered, to prevent the use of the bearer until the session establishment is completed.

- For a terminating application, to allow the destination user to participate in determining which bearers shall be established.
- Successful bearer establishment shall include the completion of any required end-to-end QoS signalling, negotiation and resource allocation.
- In establishing an IMS session, it shall be possible to use already allocated bearer resources, if these resources fulfil the needs of the session. However, note that QoS policy control mechanisms of the *IP-Connectivity Access Network* (IP-CAN) may not allow to use already allocated bearer resources.

Finally, it must be highlighted that QoS Signalling and Resource Allocation must be efficient, that is, the sequence of end-to-end QoS signalling, negotiation and resource allocation processes at different network segments should primarily consider the delay in negotiating end-to-end QoS and reserving resources that contributes to the session setup delay. Parallel or overlapping QoS negotiation and resource reservation shall be allowed where possible.

2.3 QoS Concept and architecture in UMTS

During the session setup in a IM CN subsystem, at least two levels of QoS signalling/negotiation and resource allocation should be included in selecting and setting up an appropriate bearer for the session [18]:

- At the IP BS Level: The QoS signalling and control at IP BS level is to pass and map the QoS requirements at the IP Multimedia application level to the UMTS BS level and performs any required end-to-end QoS signalling by interworking with the external network. The IP BS Manager at the UE and the GGSN is the functional entity to process the QoS signalling at the IP BS level.
- At the UMTS BS Level: The QoS signalling at the UMTS BS Level is to deliver the QoS requirements from the UE to the *Radio Access Network* (RAN), the CN, and the IP BS manager, where appropriate QoS negotiation and resource allocation are activated accordingly. When UMTS QoS negotiation mechanisms are used to negotiate end-to-end QoS, the translation function in the GGSN shall co-ordinate resource allocation between UMTS BS Manager and the IP BS Manager (see 3.3.2).

Interactions (QoS class selection, mapping, translation as well as reporting of resource allocation) between the QoS signalling/control at the IP BS Level and the UMTS BS Level take place at the UE and the GGSN which also serve as the

interaction points between the IM CN subsystem session control and the UMTS Bearer QoS control.

UMTS specific QoS signalling, negotiation and resource allocation mechanisms (e.g. *Radio Access Bearer* (RAB) QoS negotiation and PDP Context setup) shall be used at the UMTS BS Level. Other QoS signalling mechanisms such as RSVP at the IP BS Level shall only be used at the IP BS Level. It shall be possible to negotiate a single resource allocation at the UMTS BS level and utilise it for multiple sessions at the IP BS level.

2.3.1 UMTS QoS Architecture

A *Bearer Service* with clearly defined characteristics and functionality has to be setup from the source to the destination of a service in order to realise a certain network QoS. Therefore, such bearer service includes all aspects to enable the provision of a contracted QoS. These aspects are among others the control signalling, user plane transport and QoS management functionality.

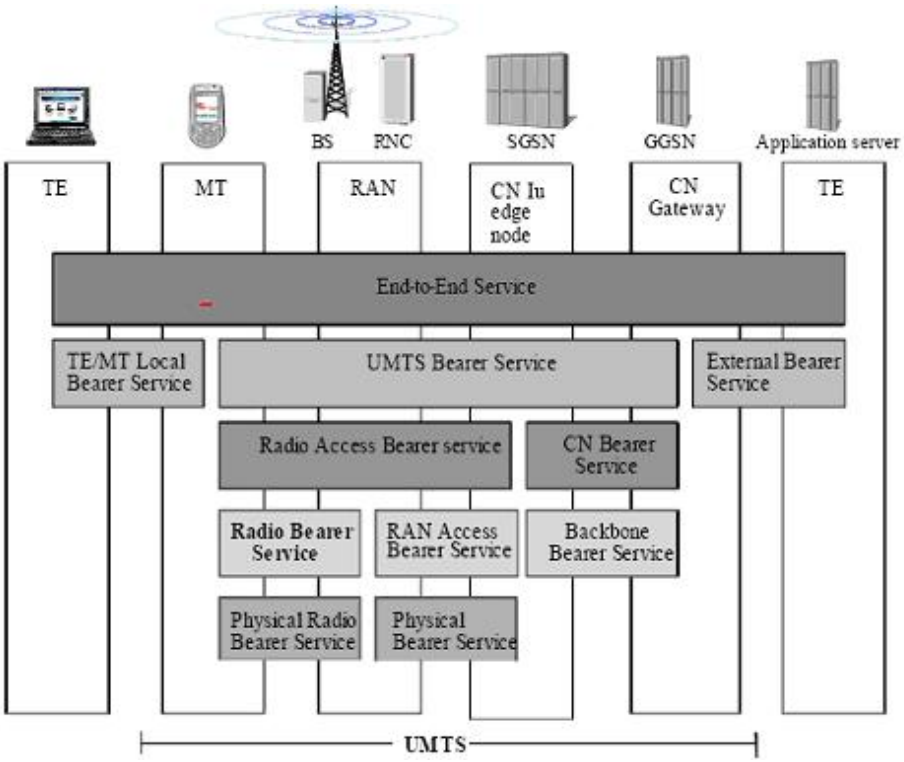


Figure 2.1: UMTS QoS Arquitecture

A UMTS BS can be decomposed into two basic parts, like it is showed in the

layered architecture depicted in Figure 2.1 : the RAB Service, that provides confidential transport of signalling and user data between the *Multimedia Terminal* (MT) and *Serving GPRS Support Node* (SGSN) with the QoS negotiated when the UMTS bearer is setup or with the default QoS for signalling; and the Core Network BS, which connects the CN edge node with the CN gateway towards external networks [4] [19]. The role of this service is to efficiently control and utilise the backbone network in order to provide the contracted UMTS BS quality.

The *Radio Bearer* (RB) Service, on the network's air interface, and the RAN Access BS, across the fixed part of the UMTS access network up to the CN edge node, together comprise the RAB Service. Each BS on a specific layer offers it's individual services using services provided by the layers below.

2.3.2 UMTS Bearer Service Attributes

UMTS BS attributes describe the service provided by the UMTS network to the user of the UMTS service. A set of QoS attributes (QoS profile) specifies this service [19][20]:

- **Traffic Class ('Conversational', 'Streaming', 'Interactive', 'Background').** Application type for which the UMTS BS is optimised. The main distinguishing factor between these QoS classes is how delay sensitive the traffic is: Conversational class is meant for traffic which is very delay sensitive, while Background class is the most delay insensitive traffic class. Conversational and Streaming classes are mainly intended for carrying real-time traffic flows, such as video telephony and audio/video streams.
- **Maximum Bit Rate (kb/s).** Upper bit rate limit a user/application can accept or provide.
- **Guaranteed Bit Rate (kb/s).** Describes the bitrate the UMTS BS shall guarantee to the user or application UMTS BS attributes, e.g. delay and reliability attributes, are guaranteed for traffic up to the Guaranteed Bit Rate. For the traffic exceeding the Guaranteed Bit Rate the UMTS BS attributes are not guaranteed.
- **Delivery Order (y/n).** Derived from the user protocol (PDP type) and specifies if out-ofsequence *Service Data Units* (SDUs) are acceptable or not.
- **Maximum SDU (Service Data Unit) Size (octets).** Maximum SDU size for which the network must satisfy the negotiated QoS. (Note: The *Maximum Transfer Unit* (MTU) of the IP layer and the Maximum SDU Size have no relationship.)

- **SDU Format Information (bits).** List of possible exact sizes of SDUs.
- **SDU Error Ratio.** Fraction of SDUs lost or detected as erroneous. It is defined only for conforming traffic.
- **Residual Bit Error Ratio.** Indicates the undetected bit error ratio in the delivered SDUs. If no error detection is requested, Residual bit error ratio indicates the bit error ratio in the delivered SDUs.
- **Delivery of Erroneous SDUs (y/n/-).** Indicates whether SDUs detected as erroneous must be delivered or discarded.
- **Transfer Delay (ms).** Indicates maximum delay for 95th percentile of the distribution of delay for all delivered SDUs during the lifetime of a BS, where delay for an SDU is defined as the time from a request to transfer an SDU at one *Service Access Point* (SAP) to its delivery at the other SAP.
- **Traffic Handling Priority.** Specifies the relative importance for handling of all SDUs belonging to the UMTS bearer compared to the SDUs of other bearers.
- **Allocation/Retention Priority.** It specifies the relative importance compared to other UMTS bearers for allocation/retention of the UMTS bearer.
- **Source Statistics Descriptor ('speech'/'unknown').** It specifies characteristics of the source of submitted SDUs.
- **Signalling Indication (Yes/No).** Indicates the signalling nature of the submitted SDUs. This attribute is additional to the other QoS attributes and does not over-ride them. This attribute is only defined for the Interactive traffic class.

Table 2.5 gathers UMTS QoS profile values for RAB Service.

2.4 Session setup procedures

Finally, this section will try to give a first overview of the procedures (in terms of fundamental SIP offer/answer exchanges) that take place during the initial phase of a video (or general multimedia) session in order to determine the set of negotiated characteristics between the endpoints of such session. That is, the flows that allow to determine the initial media characteristics (common codecs, resource reservation, bandwidth requirements) to be used. The scheme depicted in Figure 2.2 [21] shows basic video session flows and procedures , more detailed in the next subsections.

Traffic Class	Conversationa	Streaming	Interactive	Background
Maximum bit rate (kb/s)	<= 16000	<= 16000	<= 16000 – *overhead*	<= 16000 – *overhead*
Delivery order	Yes/No	Yes/No	Yes/No	Yes/No
Maximum SDU size (octets)	<= 1502	<= 1502	<= 1502	<= 1502
Delivery of erroneous SDUs	Yes/No/-	Yes/No/-	Yes/No/-	Yes/No/-
Residual BER	$5 * 10^{-2}$ - 10^{-6}	$5 * 10^{-2}$ - 10^{-6}	$4 * 10^{-3}$ - $6 * 10^{-8}$	$4 * 10^{-3}$ - $6 * 10^{-8}$
SDU error ratio	10^{-2} - 10^{-5}	10^{-2} - 10^{-5}	10^{-3} - 10^{-6}	10^{-3} - 10^{-6}
Transfer Delay (ms)	< 80	< 250	–	–
Guaranteed bit rate (kb/s)	<= 16000	<= 16000	–	–
Traffic handling priority	–	–	1,2,3	–
Allocation reservation priority	1,2,3	1,2,3	1,2,3	1,2,3

Table 2.5: UMTS QoS Profile for RAB Services

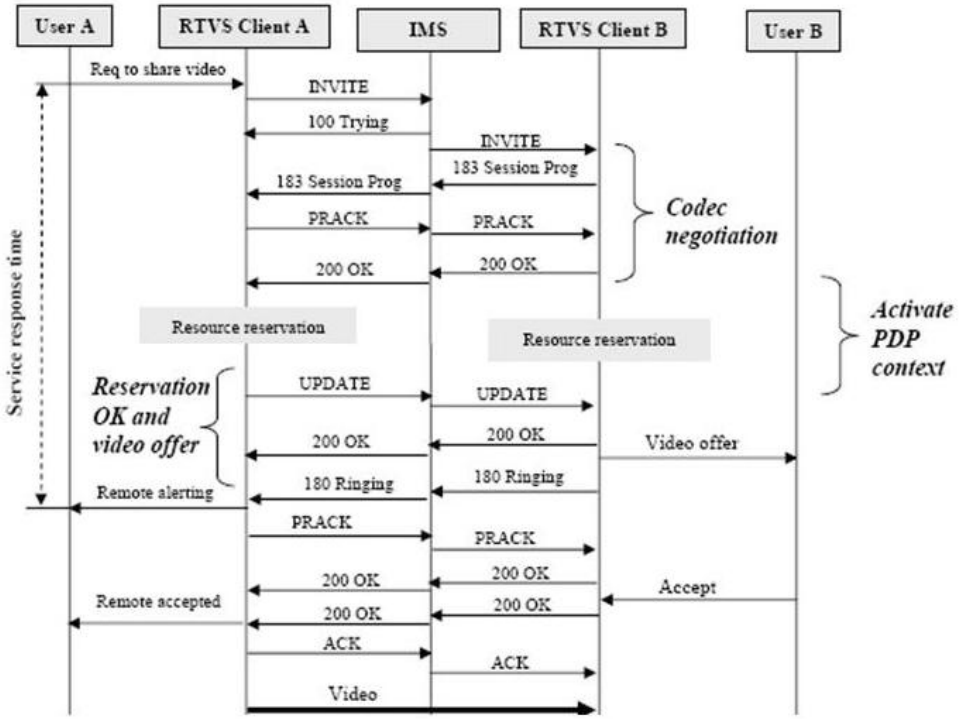


Figure 2.2: Basic video session flows

2.4.1 Codec negotiation

A negotiated set of media characteristics (including common codec or set of common codecs) that will be used for the multi-media session must be determined during initial session establishment in the IM CN subsystem. This is done by means of an end-to-end message exchange, performing a negotiation that may take multiple media offered and answered between the end points until the media set is agreed upon.

The session initiator includes an SDP in the SIP INVITE message that lists every media characteristics (including codecs) that the originator is willing to support for this session. Such offer is answered, when the message arrives at the destination endpoint, with the media characteristics (e.g. common subset of codecs) that the other part is also willing to support for the session. The session initiator, upon receiving the common subset, determines the media characteristics (including codecs) to be used initially. Media authorisation is performed for these media characteristics.

After the previous overview, let's see (taking Figure 2.3 as reference) the procedure more in deep, in order to understand how the negotiation can be established via SIP/SDP messages. The following description is only a summary of the detailed procedure gather in 3GPP 23.228 [18]. See also chapter 5 for a concrete emplacement of the codec negotiation within the basic session setup procedures.

The message exchange begins when UE1 inserts the codec(s) to a SDP payload and sends the initial INVITE message to P-CSCF1 containing this SDP. The inserted codec(s) shall reflect the UE1's terminal capabilities and user preferences for the session capable of supporting for this session. It builds a SDP containing bandwidth requirements and characteristics of each, and assigns local port numbers for each possible media flow. Multiple media flows may be offered, and for each media flow (m= line in SDP), there may be multiple codec choices offered.

The intermediate nodes, P-CSCF1, S-CSCF1, S-CSCF2 and P-CSCF2, examine the media parameters when the INVITE message is received. If they find media parameters that their local policy does not allow to be used within an IMS session, they reject the session initiation attempt. This rejection shall contain sufficient information for the originating UE to re-attempt session initiation with media parameters that are allowed by local policy. In Figure 2.5 the initial session initiation attempt is allowed at each node to continue. The Authorization-Token is generated by the PDF and included in the INVITE message finally forwarded to UE2 (see 4.2.2 and A.4.2 for more details on the Authorization Token mechanisms).

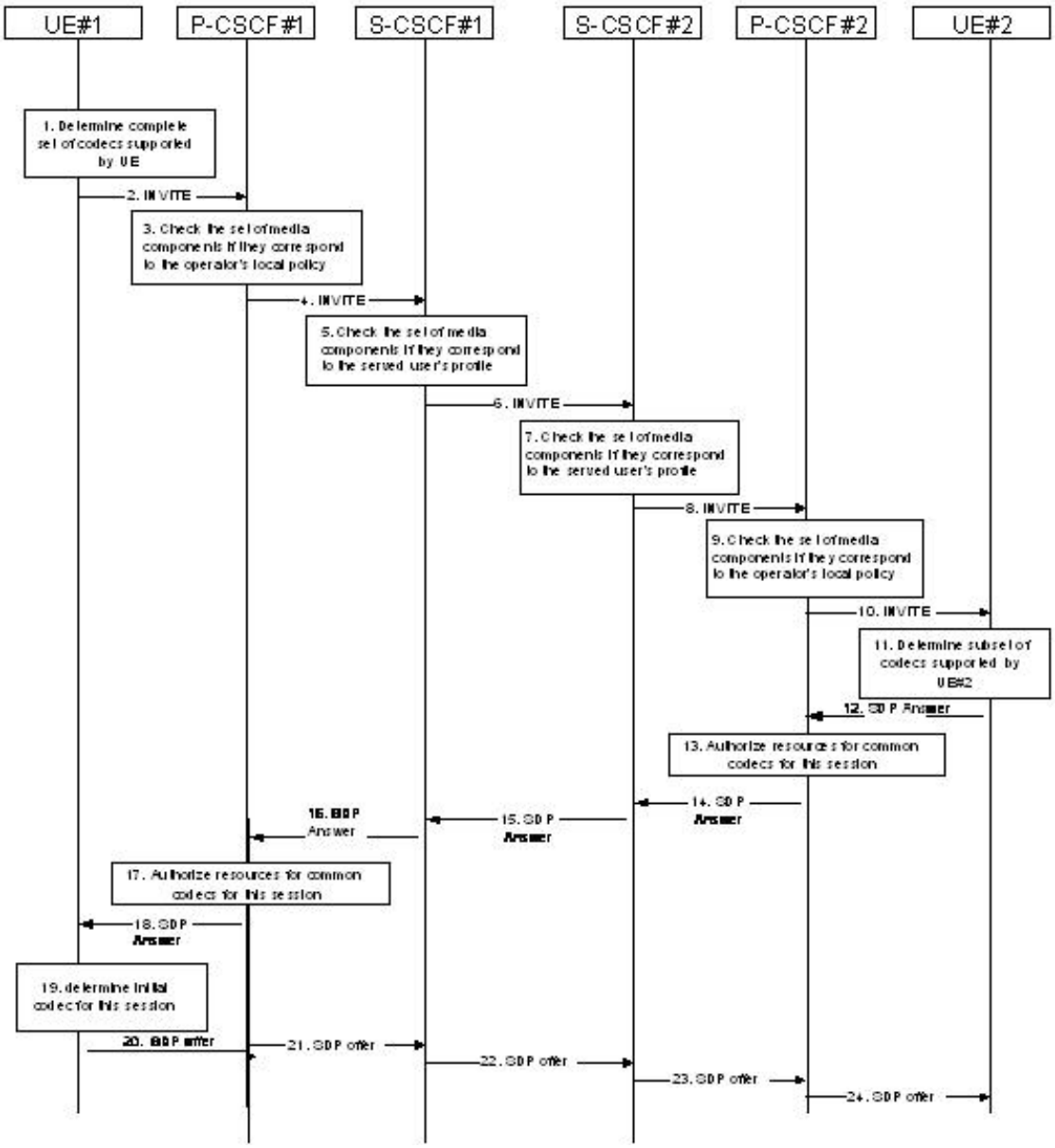


Figure 2.3: Codec Negotiation

UE2 determines the complete set of codecs that it is capable of supporting for this session. It determines the intersection with those appearing in the SDP in the INVITE message. For each media flow that is not supported, UE2 inserts a SDP entry for media (m= line) with port=0. For each media flow that is sup-

ported, UE2 inserts a SDP entry with an assigned port and with the codecs in common with those in the SDP from UE1. The SDP listing common media flows and codecs is returned to P-CSCF2, which authorises the QoS resources for the remaining media flows and codec choices.

The SDP response is forwarded through P-CSCF2, S-CSCF2, S-CSCF1, P-CSCF1. The latter authorises the QoS resources for the remaining media flows and codec choices. At this time, the Authorization- Token is generated by the PDF and included in the SDP message. Finally P-CSCF1 forwards the SDP response to UE1 that determines which media flows should be used for this session, and which codecs should be used for each of those media flows. If there was more than one media flow, or if there was more than one choice of codec for a media flow, then UE1 need to renegotiate the codecs by sending another offer to reduce codec to one with the UE2.

2.4.2 Preconditions

Preconditions [3][22] allow a session establishment conditioned to a specific achievement of a set of requirements, for example when a reservation-based quality-of-service is used. Since in the packet switched domain the resources are not preallocated before its usage is needed, it becomes indispensable ensure that the remote party will not decline the session establishment and know the bandwidth and codec that will be supported, before to start the reservation procedures. This extension, defined in RFC 3312 [23] allows user agents to express preconditions by means of a SIP option tag precondition and new SDP body attributes.

A first exchange of SDP in an INVITE request and a 183 (Session Progress) provisional response determines whether sufficient network resources are available. After that, establish a particular set of media streams becomes possible, since both parties are aware of the capabilities and willingness of the remote party. Only when this step succeeds, it's to say, the preconditions are met, the *User Agent* (UA) alerts the user (note that the previous answer provided by a 183 (Session Progress) response does not imply alerting or acceptance of the session).

In the IMS the local segmentation model for QoS reservation is used as the model of operation of preconditions extension. Each of the terminals is responsible for maintaining the appropriate resource reservation in its own local segment.

The resulting signalling flow is detailed in Figure 2.4. The INVITE request is forwarded to the destination user and it is responded with a 100 (Trying) provisional at each node. UE2 determines the complete set of codecs that it is capable

of supporting for this session. It determines the intersection with those appearing in the SDP in the INVITE request making the final codec selection. UE2 responds with a 183 (Session Progress) response containing SDP back to the originator. This response is sent to P-CSCF. UE2 uses a conf line in the SDP to request a confirmation from UE1 when the local resources are available at UE1. The terminating UAs setup the respective bearers in accordance with the media description received via SDP and the P-CSCFs authorize the necessary resources for this session. Next, a PRACK request / 200 (OK) exchange takes place.

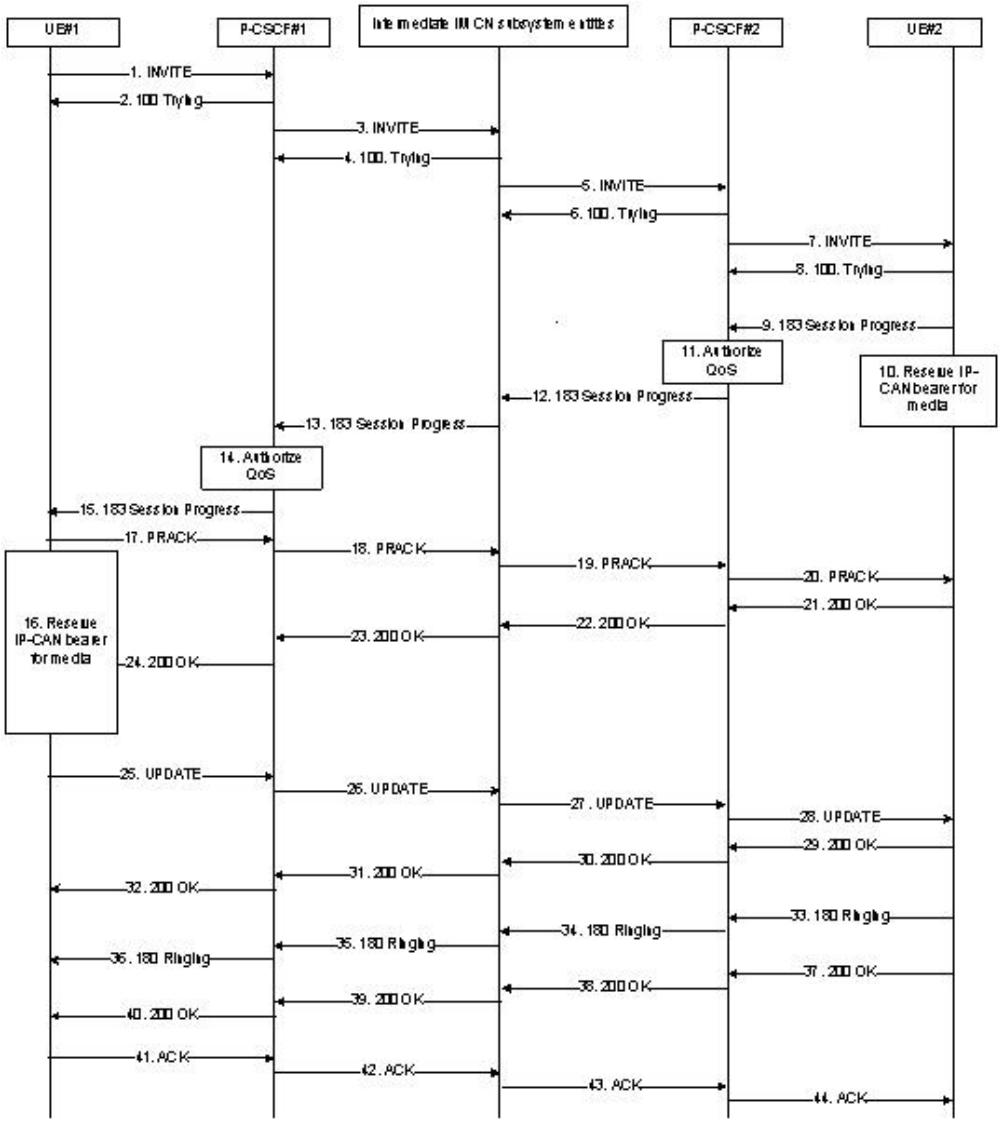


Figure 2.4: Siganlling Flow using Preconditions

When the UA that generate the session description obtains QoS in its access network it sends a UPDATE method which indicates that it can send and receive media as the necessary resources are available. UE acknowledges the UPDATE request with a 200 (OK) response, and it indicates that its local resources are also available. When the called party answers the UE sends a 200 (OK) final response to the INVITE request (numbered as 6) to P-CSCF and, with the QoS reservations finished and all the preconditions met, it sends a 180 (Ringing) method, alerting the called and allowing the subsequent start of the media flow(s) for this session.

2.4.3 PDP Context Activation

IMS capable UEs utilize the GPRS network as an access network for accessing the IMS. Thus, a UE has to have an activated PDP Context to the IMS before it can proceed to use the IMS services [3] [20] [24].

A *Packed Data Protocol* (PDP) Context is a virtual communication pipe established between the UE and the GGSN for delivering the data traffic stream. The PDP Context is defined in the UE, SGSN and GGSN by:

- A PDP Context Identifier (index of the PDP context)
- A PDP Type (e.g. *PPP or IP*)
- A PDP Address (e.g. an IP address)
- An *Access Point Name* (APN) (label describing the access point to the packet-data network)
- A QoS Profile (BS attributes).

This virtual pipe is created through the PDP Context activation process. During the activation the UE is configured with its IP address and other information needed to maintain IP access, e.g., *Domain Name System* (DNS) server address. There are three different types of PDP Contexts: *Internet Protocol version 4* (IPv4), *Internet Protocol version 6* (IPv6), and *Point-to-Point Protocol* (PPP). A UE can have one or more simultaneous PDP Contexts open to the same or to different *Gateway GPRS Support Node(s)* (GGSNs). The PDP Context can be either of the same or different types.

The IMS has been designed for work over IPv6. Thus, the activated PDP Context is of PDP Type IPv6. This means that a 3GPP IP MT uses exclusively (in principle) IPv6 to access the IMS, and the IMS SIP server and proxy support

exclusively (in principle) IPv6[1]. Hence, all the traffic going to the IMS is IPv6, even if the UE is dual stack capable - this comprises both signaling and user traffic.

PDP context activation takes time (maybe like 5 seconds or so, at least), so activating it when trying to contact a peer could lead to long waiting and unsatisfied users. Maybe it would be better if the PDP Context would be open all the time. However, some operators argue that having multiple PDP Contexts open just-in-case is not the best use of the network resources. Though, an idle PDP Context does not use much network resources.

There is a one to one correspondence between PDP context, UMTS bearer and RAB, as well as between RAB and RB Service, which, however, can be carried by more transport channels of the same type at the radio interface. A QoS profile is associated with each PDP context.

The PDP context activation procedure is depicted in Figure 2.5. The UE sends an Activate PDP Context Request message to the SGSN. The message includes: A *Network Sublayer Access Point Id* (NSAPI), *Transaction Identifier* (TI), PDP Type, PDP Address, APN, QoS Requested and PDP Configuration Options. The APN identifies the network to connect and the address space where the IP address belongs. In the case of an IMS terminal the APN indicates a desired connection to the IMS network and the connectivity type indicates IPv6.

The SGSN validates the Activate PDP Context Request using the PDP Type (optional), PDP Address (optional), and APN (optional) provided by the UE and the PDP context subscription records and chooses an appropriated GGSN. The SGSN sends a Create PDP Context Request message to the GGSN, which is responsible for allocating IP addresses belonging to the IMS address space, in the case of IMS, the GGSN provides the terminal with a 64-bit IPv6 prefix and includes it in a Create PDP Context Response message.

The SGSN forwards this prefix in an Activate PDP Context Accept. When the procedure is completed the IMS terminal has got a 64-bit IPv6, any 64-bit IPv6 suffix can be chosen by the terminal in order to form a 128-bit IPv6 address. If QoS Negotiated received from the SGSN is incompatible with the PDP context being activated, the GGSN rejects the Create PDP Context Request message. The GGSN confirms the new QoS attributes by sending an Update PDP Context Response to the SGSN.

[1]Nowadays, the IMS IPv4 compatible is been discussed

The SGSN selects Radio Priority and Packet Flow Id based on QoS negotiated, and returns an Activate PDP Context Accept message to the UE. The SGSN is now able to route PDP PDUs between the GGSN and the UE, and to start charging. For each PDP Address a different QoS profile may be requested. The UE either accepts the negotiated QoS profile, or deactivates the PDP context. If the PDP Context Activation Procedure fails or if the SGSN returns an Activate PDP Context Reject (Cause, PDP Configuration Options) message, the UE may attempt another activation to the same APN up to a maximum number of attempts.

Secondary PDP context are established by the terminals to send and receive media, with the same IP address as the primary PDP context but with particular (may be different) QoS characteristics. The number of this additional PDP context depends on the SRF information received in session descriptions (from the P-CSCF).

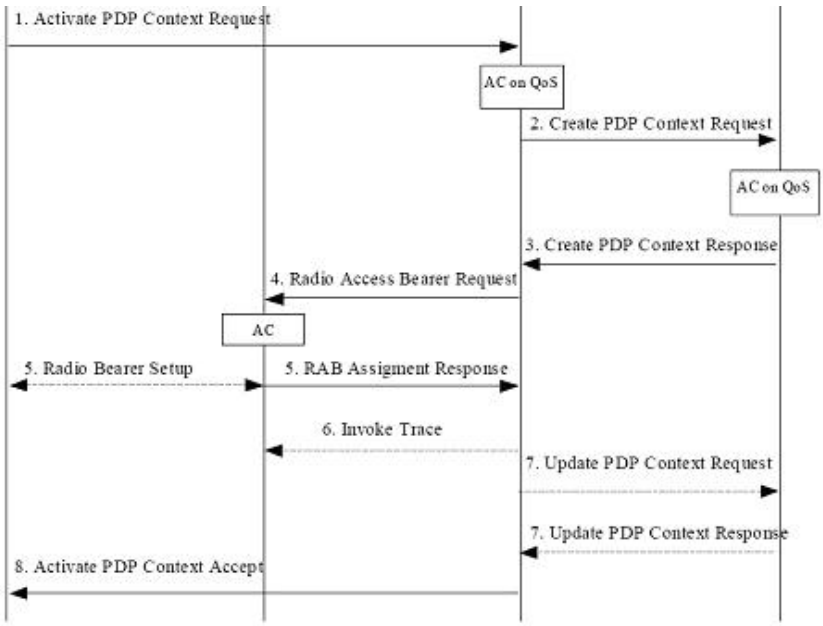


Figure 2.5: PDP Context activation

2.4.4 RSVP

Resource ReSerVation Protocol (RSVP, RFC 2205 [16]) is a control signalling protocol that allows endpoints to request a certain QoS for a flow [3] [17]. The reser-

vation needs to be renewed periodically by its requester since it has a determined time-out. Routers receiving RSVP messages obtain a description of the flow in order to apply the correct treatment to all the packets that belong to it. RSVP needs to ensure that the reservation requests for a flow are send to those routers that will route the packets of that flow, in other words, the RSVP messages have to follow the same path as the packets of the flow.

The procedure consists of a two-way handshake: a PATH message send and a RSVP message received. The network routes a PATH message as any other IP packet. The message stores the nodes it traverse when it is send end-to-end, allowing RSVP messages to find the way home when they are routed back towards the originator endpoint. Resource reservation takes place when routers receive the RSVP messages, therefore receivers initiate requests for resource reservations along the path that the packets will follow.

The Sender advertises a data flow by sending a PATH message to the receiver of the data flow. The Receiver of the data flow may initiate a reservation for the data flow by sending a RESV message. The RESV message follows the PATH message upstream hop-by-hop using the installed path states.

RSVP is an admission control protocol, in addition to being a resource reservation protocol : the resources need to be available and access policy conditions have to be met for a reservation to be successfully applied.

Chapter 3

Policy-based QoS Control Scenario for IMS

A policy-based QoS solution is adopted by the 3GPP with the purpose of satisfy a big challenge: ensuring that sufficient QoS resources are provided to authorized users in the UMTS network. Before the user can start an IMS session, the UE has to negotiate the media flows for this IMS session with the peering UE according to the service subscription. This is done with SIP based signalling in the application-layer signalling plane and the goal is that only those negotiated media flows are allowed to be transmitted in the transport-layer, i.e. in a secondary PDP context.

As [25] gathers, the 3GPP UMTS policy framework is aligned with the policy framework defined within IETF [26]. Under the IETF vision, one approach of a policy-controlled network is a state-machine model of the network which use policy to control the permanence or transition of a policy-controlled device to each state, deppending on if it is allowed to be in at any given time. In such machine-sate approach, policy is applied using a set of policy rules, being each policy rule a set of conditions and a set of actions. The policy rules are usually stored in a policy repository from which the Policy Decision Point (PDP) retrieves the appropriate policy rules in response to policy events that are triggered by the contracted IP QoS services.

The next pages intend to cover the main aspects about how Policy-Based QoS Control Scenario is implemented in IMS: models, protocols and entities involved, just as the mechanisms and procedures performed by this entities in order to guarantee QoS Control.

3.1 Model for a policy-based network

According to this IETF framework [26], the reference model of a policy-based network consists of two main elements, the *Policy Decision Point* (PDP) and the *Policy Enforcement Point* (PEP). PEPS often reside in policy-aware network nodes that carry out actions stipulated by policy rules. The actions taken are based on the decisions of a PDP, which retrieves the policy rules from the repository. The PDP is the final authority the PEP needs to refer to for actions to be taken [25].

The QoS control scenario for IMS services defined in 3GPP IMS architecture relies on SIP interacting with (among others) two IMS elements: the P-CSCF and PDF. A more concrete IMS architecture overview is provided in 6.1, however, can be summarized that the P-CSCF is a call agent that provides session handling functionality, being the first contact point for the UE within the IMS, and it is always located in the same network as the GGSN.

On the other hand, the PDF provides service-based policy control of the access to IMS services by the user in an operator-controlled manner and controls UE initiated bearer establishment by handling resource management requests coming from the GGSN via the Go interface. The PDF is a logical entity that can be either collocated with the P-CSCF (3GPP Release 5) or implemented as a stand-alone unit (3GPP Release 6) [27].

In the IMS, the PDF plays the role of the PDP and, since the gateway GGSN is in the data path, it is the logical location for the PEP. The policy repository can be an entity external to the PDF[1]

The COPS protocol is used between them over the Go interface allowing service-based local policy information be conveyed between the PEP (the GGSN) and the PDP (the PDF)[2]. Go interface allows two modes of operation. In the push mode, the PDF initiates communication with the PEP and sends the PEP its decision. In the pull mode, the PEP initiates communication with the PDF to request a decision for a particular IP flow [28].

Gq interface is used for the service-based policy setup information exchange (dy-

[1]In 3GPP Release 5 this entity is named *Policy Control Function*, later, in 3GPP release 6 the PCF is renamed to *Policy Decision Function* (PDF))

[2]Since the acronym for Policy Decision Point is the same than that used referred to the Packet Data Protocol and because in IMS, the role of the PDP is performed by the PDF the remainder of this chapter will try to avoid the use of PDP referred to the Policy Decision Point, instead of it, the acronym PDF will be used

namic QoS-related application information) between PDF and the *Application Function* (AF), an element offering applications that require the control of IP bearer resources. In the case we are focused on, the IM CN subsystem, the AF is the P-CSCF [27].

The relationships between the different functional entities involved are depicted in Figure 3.1 [29].

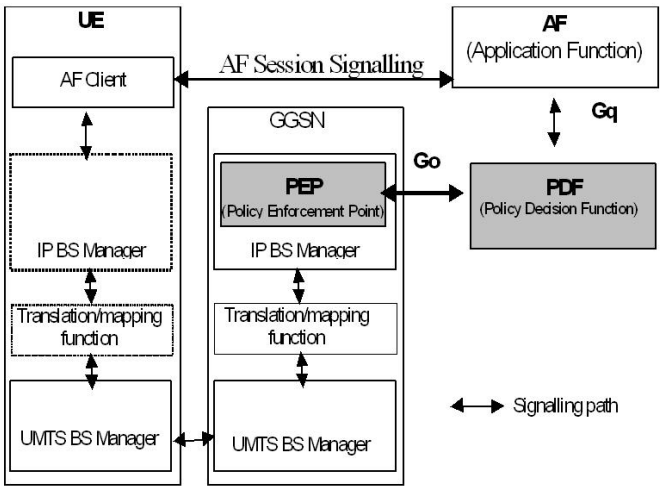


Figure 3.1: Functional entities involved in the policy-based QoS scenario

As mentioned earlier, a P-CSCF is the first contact point in the IMS domain for a UE [18] during the establishment of a SIP session, for this reason, it is the natural place to authorize usage of network resources such as the bandwidth requested by the UE. The QoS requirements of the UE are carried in the SDP within a SIP message. As well as the QoS requirements in the SDP description, the PDF also examines, in order to take an appropriate decision, the source and destination IP addresses and port numbers [18] [28].

The PDF based on the policy rules usually stored in a policy repository, governing the local domain, generates an authorization token that identifies the SIP session in an uniquely manner across multiple PDP Contexts terminated by a GGSN. This token is sent to the UE via SIP messages so that the UE can use it to identify the associated session flows to the PEP in the GGSN in subsequent transmissions of IP packets [28]. This mechanism is consistent with the IETF specification on supporting media authorization in the SIP protocol (See 4.2.2, A.4.2 and RFC 3313 [30]).

3.2 COPS Protocol over Go Interface

In the IMS, the session setup signalling is separated from the data path of the session [28] [29]. The PDF resides on the signalling path while the GGSN resides on the data path. The role of the PDF is to authorize the establishment of a session at the policy level. To actually establish the data path of the session, the GGSN must reserve the proper level of QoS resources. To tie policy-level authorization to the corresponding QoS resource reservation, the PEP component of the GGSN must validate the reservation request with the PDF. The Go interface facilitates the necessary communication between the PDF and the PEP to realize this validation. The 3GPP uses the COPS protocol as the communication protocol on the Go interface.

The COPS Protocol (RFC 2748),

- Request (REQ) - is used when PEP (the GGSN in IMS) requests SBLP and QoS inter-working information.
- Decision (DEC) -is a response message to the REQ message or an asynchronous notification from PDP (the PDF in IMS) to the GGSN.
- Report State (RPT) -is used to communicate the success, failure or changes to the client state of the GGSN in carrying out the PDF's decision indicated in the DEC message.
- Delete Request State (DRQ) - message from the GGSN to the PDF indicates that the state identified by the client handle is no longer available/relevant and the corresponding state may be removed from the PDF).

3.3 The PEP in the GGSN

In the PS domain, a GGSN is the network node that maintains connectivity to other PS external networks such as the Internet. Policing the 1P packets based on their source and destination IP addresses and port numbers, the GGSN controls, from the service point of view, which IP flows are permitted into the external IP network [28]. Taking the role of the PEP in the QoS control scenario, it ensures that only authorized IP flows are allowed to use network resources that have been reserved and allocated to them. The policy enforcement function in the GGSN is called a gate, and it comprises a packet classifier, a traffic meter, and the relevant packet handling mechanisms for packets that have been matched by the packet classifier. The gate description is received by the GGSN from the PDF in the

authorization decision

The PEP opens the gate for an IP flow once it has been authorized by the PDF to use the specified network resources, and effectively commits the network resources to the flow by allowing it to pass through the packet handling mechanisms (i.e., policing or marking). On the other hand, if the PDF does not permit an IP flow to use the requested resources, the PEP closes the gate and drops the IP packets of the flow [28]. This process is called policy-based admission control. It ensures that an IP flow is only allowed to use resources that have been approved by the policy rules. The above process takes place at the IP BS level. The translation mapping function within the GGSN will map this resource information into the format used by the admission control function at the UMTS BS level.

The authorized resources provide an upper bound on the resources that can be reserved or allocated for the set of IP flows. The authorized resources are expressed as a maximum authorized bandwidth and QoS class. The PDF generates a maximum authorized QoS class for the set of IP flows and this information is mapped by the Translattion/Mapping function in the GGSN to give the authorized resources for UMTS bearer admission control [31].

The P-CSCF, acting as AF, maps QoS related application level parameters (expressed in the SDP message) into policy setup information, and sends this information to the PDF in order to obtain the authorization of the QoS settings for the requested service. After that, the UE obtains an authorization token from the P-CSCF via SIP signalling during session setup. This token is used to provide the binding mechanism that associates the PDP context bearer to the IP flow in order to support IP policy enforcement in the GGSN. By examining this token received from the GGSN, the PDF can direct the GGSN to admit or drop the flow [25].

3.3.1 Policy control: Requesting authorization

In order to perform the initial authorization at PDP context activation the GGSN sends an authorisation request to the PDF including the binding information previously received from the UE. The required PDF is identified from the same binding information [29].

The GGSN authorisation request message to the PDF allows the GGSN to request policy information for authorisation of the media components carried by a PDP context identified by binding information. Once the GGSN receives the PDF

resolution regarding authorisation of this media components, it enforces the policy decision.

The PDF verifies the binding information by checking if the authorization token is associated with an ongoing SIP session at IMS level and by checking if the media components are allowed to be grouped. If so, the GGSN shall proceed with activation of the PDP context and map the authorized QoS resources into authorized resources for the bearer admission control. If not, or if the PDF is otherwise unable to authorise the binding information, the GGSN will receive a COPS decision message from the PDF indicating the rejection of the PDP context activation [29]. The authorization failure is indicated to UE in the Protocol Configuration Options information element as defined in [32].

After the successful establishment of the PDP context and in order to ensure charging correlation, the GGSN sends the GPRS charging identifier and GGSN address information to the PDF.

If the GGSN sends an authorization request to the PDF but the PDF doesn't respond with the decision message, the GGSN's action is according to the local policy in the GGSN (which may be configured by the operator) [29]. That is, the PEP may store decisions in a local PDP, thus allowing the GGSN to make admission control decisions without additional interactions with the PDF. As a result, the traffic over the Go interface can be reduced and the processing load on the PDF can be lessened [28]; however, the cost is that PEP must store more policy information than is necessary in order to handle all possible policy events. In some cases this feature is especially useful: the local policy decisions may be used to accept new PDP context activations while the connection to the PDF is lost.

The GGSN is also responsible for notifying the PDF when a procedure of PDP context modification of a previously authorized PDP context is performed following the indications gather in 3GPP TS 29.207 [29]. (The PEP shall inform the PDF when the bearer changes to or from a data rate of 0 kb/s -an indication of bearer loss/recovery-, and at bearer release [27]).

3.3.2 Mapping/translation function

Using Go interface between IMS PDF and GGSN, IMS performs QoS negotiation and media flow authorization between IP QoS parameters and UMTS QoS parameters.

GGSN communicates (using COPS with COPS-PR extensions) with PDF regarding SBLP control. The GGSN sends requests and receives decisions from the PDF. The GGSN has to perform proper mapping between the IP QoS information (requested by IMS session) and the UMTS QoS information (provided by UMTS Core Network). This mapping should be done by the Translation/Mapping function [29] that derives the highest allowed UMTS Traffic class for the PDP context from the QoS class in the "Authorized QoS" according to Table 3.1:

QoS Class	UMTS Traffic Class	Traffic Handling Priority
A	Conversational	N/A
B	Streaming	N/A
C	Interactive	1
D		2
E		3
F	Background	N/A
NOTE: QoS class represents the highest class that can be used for the bearer.		

Table 3.1: Mapping UMTS Traffic Class to IP QoS parameters

The QoS class values given by the PDF are equal for both the uplink and the downlink directions.

If the requested QoS exceeds the authorized QoS, the UMTS Bearer Service Manager shall downgrade the requested UMTS QoS information to the authorized QoS information. In the case of real-time UMTS bearers (conversational and streaming traffic classes), the GGSN shall consider, the Data rate value of the "Authorized QoS" information as the maximum value of the 'Guaranteed bitrate' UMTS QoS parameter, whereas the 'Maximum bitrate' UMTS QoS parameter is limited by the subscriber and service specific setting in the HLR/HSS (SGSN) and by the capacity/capabilities/service configuration of the network (GGSN, SGSN). In the case of non-real-time bearers (interactive and background traffic classes) the GGSN shall consider, the Data rate value of the "Authorized QoS" information as the maximum value of the 'Maximum bitrate' UMTS QoS parameter [29] [33].

3.4 Policy Decision Function (PDF)

Such as it has been mentioned previously, the PDF is one of the most important functions for providing QoS support within IMS. By means of standard IP mechanism this logical policy decision entity implements *Service Based Local Policy* (SBLP) in the IP bearer layer. It acts like Policy Decision Point (PDP) for the SBLP control. The PDF makes decisions in regard to SBLP using policy rules and communicates that decision to the IP BS Manager in the GGSN, which is the IP - PEP.

Upon receiving a bearer authorisation request from the PEP, the PDF shall be able to provide an authorisation decision according to the stored session and media related information obtained from the AF (the P-CSCF in IMS). Gq interface is used for the service-based policy setup information exchange (dynamic QoS-related application information) between PDF and the AF. The policy setup information provided by the P-CSCF to the PDF for each media component shall contain the following [29] :

- Destination IP address;
- Destination port number;
- Transport Protocol id;
- Media direction information;
- Direction of the source (originating or terminating side);
- Indication of the group that the media component belongs to;
- Media type information;
- Bandwidth parameter.

PDF may allow or deny the usage of the PDP Context for the selected IP flow(s) by controlling the correlated gate(s). The "Approval of QoS Commit" command may either be part of the authorisation decision, or the PDF may provide a separate decision with the "Approval of QoS Commit" command to open the gate. The "Removal of QoS Commit" command is a separate decision to close the gate(s) [27]. When the PEP informs the PDF of bearer deactivation, the PDF shall remove the corresponding authorisation request state. When the PDF receives an indication of bearer modification of the maximum bitrate to or from 0 kbits/s, the PDF shall inform the AF about this modification event. The PDF generates an

authorisation token for the AF session. To perform proper authorisation, the PDF shall map the necessary service information containing session and media related information to "Authorized QoS" parameters.

As 3GPP TS 29207 [29] gathers, the "Authorised QoS" information (consisting of maximum DiffServ Class and Data Rate) for a media component is extracted from the media type information and bandwidth parameter of the SDP. The PDF shall map the media type information into a DiffServ Class which is the highest class that can be used for the media. The PDF shall extract the Data Rate value from the "b=AS" SDP parameter (see 4.2.4 and 6.6). The "b=AS" parameter in the SDP shall contain all the overhead coming from the IP-layer and the layers above, e.g. UDP, RTP. The Data Rate shall also include the overhead coming from the possible usage of RTCP.

When receiving an AF session signalling message initiating a new AF session, the AF shall request an authorization for the session from the PDF. The AF shall indicate to the PDF whether the media IP flow(s) should be enabled or disabled at the bearer authorization. Depending on the application, the AF may instruct the PDF also during the session when the IP flow(s) are to be enabled or disabled to pass through the access network. During the AF session modification, the AF shall send an update for the session description information to the PDF within the AF session signalling.

3.5 QoS Control Scenario for IMS Services

The QoS control scenario for IMS services is shown in the next figure and described as following [27]

1. At IMS session establishment, the two users negotiate, using SIP, to establish a session and communicate the media characteristics they want to use to each other by attaching SDP media characteristics to the session initiation messages.

2. Knowing the media characteristics (QoS, bandwidth, etc) to be used for the session, the P-CSCF contacts the PDF to obtain authorization for session resources and a PDF-generated token. The PDF then checks wether the policy defined in its policy server authorizes the session with the provided session requirements and passes on the binding information to be used for PDP context creation. The P-CSCF includes the media authorization token within the message body attached to the response message sent back to the UE.

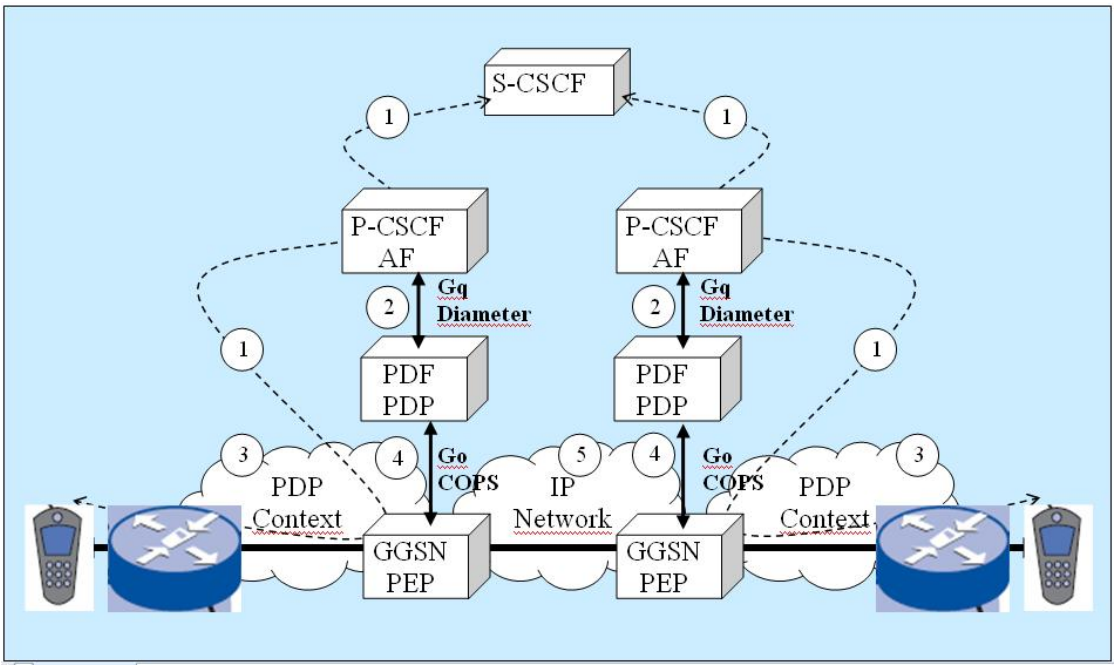


Figure 3.2: QoS Control scenario for IMS

3. After an agreement has been reached between the two users at the session level, the UE derives from the media characteristics the QoS parameters needed for the bearer reservation in the UMTS access network. To reserve the required resources, the UE initiates a setup of a PDP context, whereby the UE inserts the authorization token and derived QoS parameters in the PDP signalling.

4. On receipt of a PDP Context activation request from the UE, the GGSN contracts the PDP (with the authorization token) through the Go interface to request authorization for the bearer setup. If all checks are valid, the PDF informs the GGSN of its decision to authorize resource reservation, then the GGSN accepts the creation of the PDP context.

5. In addition, GGSN could establish QoS data path using either path-couple signalling (RSVP) or DiffServ between two GGSNs, if it is possible.

Chapter 4

SIP & SDP over IMS

A multimedia, and therefore a video session requires control and description tools that the IMS relegates to *Session Initiation Protocol* (SIP) and *Session Description Protocol* (SDP). SIP was originally proposed by IETF like an application-layer protocol to establish, modify and terminate multimedia sessions in all-IP networks, later it was chosen as the session control protocol for the IMS. By the other hand, SDP was published by the IETF as a format for describing multimedia sessions and its streaming initialization parameters for the purposes of session announcement, session invitation, and other forms of multimedia session initiation.

The next lines will be focused in the signalling flows for session setup in the IMS based on SIP and SDP. With such an aim in mind, the QoS or streaming related extensions of both protocols will be discussed.

4.1 SIP/SDP Overview

4.1.1 SIP Overview

Similar to HTTP, SIP is a text-based protocol, which makes it easy read and understand. A SIP message is either a request from a client to a server or a response from a server to a client. Both the request and the response contain a start-line followed by one or more headers and a message body. The request line specifies the type of request being issued, while the response line indicates the success or failure of a request. If a request is not executed, the status line indicates the type of failure or the reason for the failure. SIP RFC 3261 [5] defines the six original methods used for different types of requests; moreover, such requests had been

extended and enhanced, including the addition of new methods to SIP in order to provide event notification, instant messaging and call control. The original SIP methods are listed next [5]:

- INVITE: Initiates a session. This method includes information about the calling and called users and the type of media that is to be exchanged.
- ACK: Sent by the client who sends the INVITE. ACK is sent to confirm that the session is established. Media can then be exchanged.
- BYE: Terminates a session. This method can be sent by either user.
- CANCEL: Terminates a pending request, such as an outstanding INVITE. After a session is established, a BYE method needs to be used to terminate the session.
- OPTIONS: Queries the capabilities of the server or other devices. It can be used to check media capabilities before issuing an INVITE.
- REGISTER: Used by a client to login and register its address with a SIP registrar server.

Some important extensions are the next:

- SUBSCRIBE: Enables a user to subscribe to certain events.
- NOTIFY: Used to inform the user that a subscribed event has occurred.
- MESSAGE: SIP can also be used for Instant Messaging.
- INFO: Transferring information during a session, such as user activity.
- SERVICE: Used to carry a *Simple Object Access Protocol* (SOAP) message as its payload.
- NEGOTIATE: Used to negotiate various kinds of parameters, such as security mechanisms and algorithms.
- REFER: Enables the sender of the request to instruct the receiver to contact a third party using the contact details provided in the request.

The outcome of a request is indicated in the associated SIP responses by means of a status code, which consists in a three-digit number, along with a reason phrase, which provides a textual description that helps the user understand the response. Status codes defined in SIP have values between 100 and 699 and the first digit of the reason code indicates the response class [5]:

- 1xx: Provisional: Request received, continuing to process the request. For example, 180 indicates that the phone of the called user is ringing.
- 2xx: Success: Action was successfully received, understood, and accepted. Only 200 OK and 202 ACCEPTED have been defined in this class.
- 3xx: Redirection: Further action needs to be taken to complete the request. For example, a front-end server sends 302 to redirect the client to a home aerver. user.
- 4xx: Client Error: Request contains bad syntax or cannot be fulfilled at this server. For example, a Home Server sends a response, 401 Unauthorized, if the client needs to provide credentials.
- 5xx: Server Error: Server failed to fulfill a valid request. For example, a server sends a response, 504 Timeout, if *Multicast Transport Layer Security* (MTLS) has not been configured between the home servers.
- 6xx: Global Failure: Request cannot be fulfilled at any server. This is a new class defined for SIP, but is not currently used with Live Communications Server 2003.

SIP includes a number of message headers in a SIP message. These headers contain information that enables the receiver to understand the message better or handle the message properly. Some headers make sense only in certain requests or responses. In some cases, the presence of a particular header depends on the context. The presence of a particular header in a response might be reasonable only if the response is issued to a specific request.

Are known as general headers those that can be used in both requests and responses, such as the To:, From: and Call-ID header fields. On the other hand, request headers apply only to SIP requests and in the same way, response header fields apply only to response (status) messages. In the first case, they are used to provide additional information to the server about the request or the client. For example, Subject: can be used to provide a textual description of the topic of the session. Priority: is used to indicate the urgency of the request, such as emergency, urgent, normal, or nonurgent. In the second case, the header fields are used to provide further information about the response that cannot be included in the status line. For example, Unsupported: is used to identify those features that are not supported by the server. Retry-After: indicates when a called user will be available if the user is currently busy or unavailable.

The following list corresponds to basic SIP header fields [5]:

- Via: Indicates transport used and request route (each proxy adds a line to this field)
- From: originator of the request.
- To: recipient of the request.
- Call-Id: uniquely identifies a specific invitation to a session
- Cseq: It is started with a random number and it identifies sequentially each request.
- Contact : Address/es that can be used in order to contact with the user.
- User Agent: User agent that performs the communication.

Once overviewed the basic request/response messages that can be exchanged it is possible to give a broad outline of the SIP session establishment: an INVITE message sent from the caller to the callee starts the session establishment, inviting this one to share a conference. Before the called user accepts the call, the caller might receive several provisional responses such as those that inform when the called user is being alerted or the previous 183 (session progress). After a number of request/response exchanges, the call is answered by the callee generating a final OK response like INVITE answer. Once the calling user receives this OK message, it sends an ACK after which media, such voice, video or text, is exchanged. The call ends after a BYE message, generated when one of the users hangs up, and the session over confirmation of the other part.

4.1.2 SDP overview

In the previous section is mentioned that both, the SIP request and response, contain a start-line followed by one or more headers and a message body. The main use for entity body is carrying an SDP message describing multimedia sessions and its streaming initialization parameters for the purposes of session announcement, session invitation, and other forms of multimedia session initiation. The media and transport information included in an SDP session description is the following [8]:

- The type of media (video, audio, etc.)
- The transport protocol (RTP/UDP/IP, H.320, etc.)
- The format of the media (H.261 video, MPEG video, etc.)

Contrary to SIP SDP header fields need to be in a specific order and they have strict formatting rules with the aim of make the analysis simpler, so that most errors would result in malformed session announcements that could be detected easily and discarded (enhancement of error detectioin). The header fields' names are abbreviated by only one letter (exactly one case-significant character). SDP was not designed to be easily extensible, it is only possible define new attributes, nevertheless such attributes could be ignored. The SDP header fields are listed next (* optional item)[8]:

- Session description
  - v= (protocol version)
  - o= (owner/creator and session identifier).
  - s= (session name)
  - i=* (session information)
  - u=* (*Uniform Resource Identifier* (URI) of description)
  - e=* (email address)
  - p=* (phone number)
  - c=* (connection information - not required if included in all media)
  - b=* (bandwidth information)
  - One or more time descriptions
  - z=* (time zone adjustments)
  - k=* (encryption key)
  - a=* (zero or more session attribute lines)
  - Zero or more media descriptions
- Time description
  - t= (time the session is active)
  - r=* (zero or more repeat times)
- Media description
  - m= (media name and transport address)
  - i=* (media title)
  - c=* (connection information - optional if included at session-level)
  - b=* (bandwidth information)

  - k=* (encryption key)
  - a=* (zero or more media attribute lines)

An SDP session description consists of a session-level section followed by zero or more media-level sections. The session-level part starts with a "v=" line and continues to the first media-level section. Each media-level section starts with an "m=" line and continues to the next media-level section or end of the whole session description. In general, session-level values are the default for all media unless overridden by an equivalent media-level value.

Except under rare circumstances, the set of media types (intended to be small) should not be extended. RFC 4566 [8] registers the media types "audio", "video", "text", "application", and "message". The media types "control" and "data" were listed as valid in the previous version of the specification; however, their semantics were never fully specified and they are not widely used.

4.2 SIP/SDP and IMS

As it was seen previously, the 3GPP chose SIP as its base signalling protocol, since it complies with all IMS requirements, unlike previous signalling protocols. Many features justify such choose: SIP, as an Internet protocol, allows signalling between different network entities, meeting all the convergence needs of the IMS architecture; SIP uses Internet extensibility mechanisms, that is, IMS networks must be easily scalable to add more subscribers to a service provider; SIP is also very flexible, and uses standard extensions, enabling IMS networks to adapt and change signalling protocols to meet dynamic market needs; finally, SIP provides adequate security, with both internal and external security mechanisms [34].

Nevertheless, there were many gaps between the original SIP defined by the IETF, and the features required for full IMS support, being SIP in its IMS form a set of technological challenges. This problem gave rise to the need to define specific IMS networks SIP extensions, comprising the IMS SIP protocol, finally defined in 3GPP TS 24.229 standard [35]. The SIP features over IMS can be summarized as follows [36]:

- A few headers more
  - P-Headers, used to convey information not included in standard SIP
  - PATH and Service-Route:
- Additions to some headers

  - WWW-Authenticate and Authorize
  - VIA, Route ..

- Stricter routing paths (e.g., P-CSCF to S-CSCF to *Interrogating*-CSCF to S-CSCF to P-CSCF)
- *eXtensible Markup Language* (XML) body used for transporting information from HSS to the SIP elements (emergency)
- Specification of timer values (request retransmission ...)
- More intensive use of some of SIP and SDP extensions (PRACK, UPDATE, qos, offer-answer ...)

In the same way, IMS extends SDP with even more extensions, such as grouping of media lines, QoS and preconditions attributes, supplemental codec support, and bandwidth modifiers; such extensions are added to the original SDP content, that defines the basic negotiation process for the media streams, and includes the bit rate and codec to be used, as well as other media attributes.

The following subsections perform a review of some of SIP/SDP extensions.

4.2.1 Integration of Resource Management and SIP Preconditions RFC 3312

Preconditions allow a session establishment conditioned to a specific achievement of a set of requirements, for example when a reservation-based QoS is used. Since in the packet switched domain the resources are not preallocated before its usage is needed, it becomes indispensable ensure that the remote party will not decline the session establishment and know the bandwidth and codec that will be supported, before to start the reservation procedures [3]. This extension, defined in RFC 3312 [23] allows user agents to express preconditions by means of a SIP option tag precondition (signalling) and new SDP body attributes (to describe the parameters of the session). Note that the QoS preconditions are included in the SDP description rather than in the SIP header because preconditions are stream specific.

The concept of precondition appears as a solution for the problem that resources cannot be reserved without performing an initial offer/answer exchange, and the initial offer/answer exchange can't be done without performing resource reservation. By means of preconditions, the offer includes a set of constraints about the session, so its recipient generates an answer, but does not alert the user or otherwise proceed with session establishment until the preconditions are met.

Each media stream is affected by two different state variables in order to ensure that session establishment does not take place until certain preconditions are met, in this case, a certain QoS; these variables are *current status* and *desired status*. Being the latter a threshold for the current status, session establishment stops until the current status reaches or surpasses this threshold. Once this threshold is reached or surpassed, session establishment resumes. These two new variables are exchanged between two user agents using an offer and an answer in order to have a shared view of the status of the session. Media level SDP attributes are defined in RFC 3312 [23] as follows:

- Current status: Current status of network resources for a particular media stream. "a=curr:" precondition-type [ ] status-type [ ] direction-tag

- Desired status: Preconditions for a particular media stream. When the direction-tag of the current status attribute, with a given precondition-type/status-type for a particular stream is equal to (or better than) the direction-tag of the desired status attribute with the same precondition-type/status-type, for that stream, then the preconditions are considered to be met for that stream. "a=des:" precondition-type [ ] strength-tag [ ] status-type [ ] direction-tag

- Confirmation status: Threshold conditions for a media stream. When the status of network resources reach these conditions, the peer user agent will send an update of the session description containing an updated current status attribute for this particular media stream. "a=conf:" precondition-type [ ] status-type [ ] direction-tag

- Precondition type: RFC 3312 defines quality of service preconditions. Extensions may define other types of preconditions.precondition-type = "qos" — token

- Strength tag: The strength-tag indicates whether or not the callee can be alerted, in case the network fails to meet the preconditions. strength-tag = ("mandatory" — "optional" — "none" — "failure" — "unknown")

- Status type: two types of status are defined: end-to-end and segmented. The end-to-end status reflects the status of the end-to-end reservation of resources. The segmented status reflects the status of the access network reservations of both user agents. The end-to-end status corresponds to the tag "e2e", and the segmented status to the tags "local" and "remote". End-to-end status is useful when end-to-end resource reservation mechanisms are

available. The segmented status is useful when one or both UAs perform resource reservations on their respective access networks. status-type = ("e2e" — "local" — "remote")

- Direction tag: The direction-tag indicates the direction in which a particular attribute (current, desired or confirmation status) is applicable to. direction-tag = ("none" — "send" — "recv" — "sendrecv")

In order to make use of preconditions with the offer/answer model allowing a shared view of the session parameters for both user agents the model defined consists of three tables: both user agents implement a local status table, and each offer/answer exchange has a transaction status table associated to it. The first transaction table is identical to the offerer local status table, by means of which the answerer updates its local status table, updating also the transaction status table fields that were out of date and returns this in the answer. The offerer can at this point update its local status table with the new information received in the answer. As a result, the local status tables of both user agents are synchronised. Sessions that involve several media streams implement these tables per media stream.

4.2.2 Media authorization RFC 3313

This extension allows QoS provision for media streams established via the Session Initiation Protocol (SIP). The basic architecture is illustrated in Figure 4.1 [30].

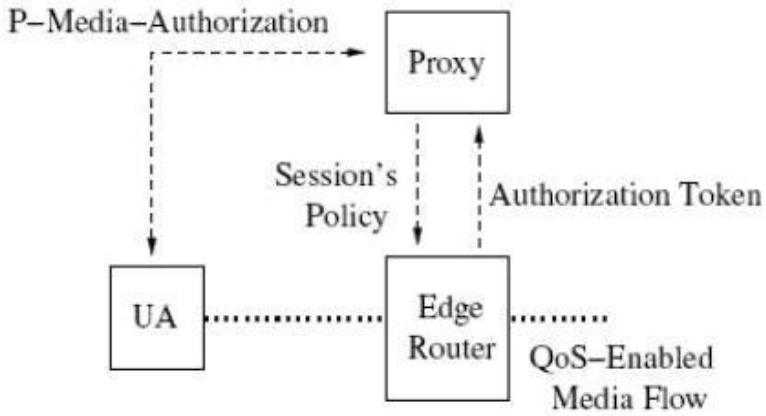


Figure 4.1: Media Authorization Architecture

In this architecture, is assumed a SIP UA connected to a QoS enabled network with an edge router (in IMS is the GGSN) acting as a PEP (already described in chapter 3). The SIP UA that wishes to obtain QoS initiates sessions through a

proxy which can interface with the QoS policy control for the data network being used. Such proxy, also referred as a QoS enabled proxy, corresponds to the P-CSCF in the 3GPP architecture.

The mechanism to authorize the establishment of media streams is based on the network inserting a media authorization token that IMS terminals return to the network when requesting the establishment of a media stream [3]. The authorization token that a SIP UA needs to present to the network in order to obtain QoS is obtained via SIP from the QoS enabled proxy by means of an extension SIP header. The proxy, in turn, communicates either directly with the edge router or with a PDF (not shown) in order to obtain a suitable authorization token for the UA.

The mechanism in 3GPP that makes use of the described SIP extension is known as the *Service-Based Local Policy* (SBLP) specified in 3GPP TS 29.208 [37], which follows the media authorization model described in RFC 3521 [38].

The SIP UA sends an INVITE to the QoS enabled proxy (P-CSCF), which adds to the incoming INVITE in a P-Media-Authorization header field the authentication token generated by the PDF. When the UA requests QoS, it includes the media authorization tokens with the messages it sends to perform resource reservation. A SIP UA may also receive an INVITE from its QoS enabled proxy which includes one or more media authorization tokens.

In order to support a media authorization scheme a new P-Media-Authorization general header field is defined. This header field, a private SIP extension, contains one or more media authorization tokens which are to be included in subsequent resource reservations for the media flows associated with the session.

```
P-Media-Authorization = "P-Media-Authorization" HCOLON
         P-Media-Authorization-Token
         *(COMMA P-Media-Authorization-Token)
P-Media-Authorization-Token = 1*HEXDIG
```

4.2.3 SDP Bandwidth Modifiers for RTCP Bandwidth (RFC 3556)

The RTP shall include a control protocol RTCP implementation, which synchronization information from data senders and feedback information from data receivers [11]. RTCP traffic bandwidth shall be described using the "RS" and "RR"

SDP additional bandwidth attribute modifiers at media level. The two new modifiers may be used to specify the RTCP bandwidth allowed for active data senders and the RTCP bandwidth allocated to other participants in the RTP session (i.e., receivers), respectively.

Normally, the amount of bandwidth allocated to RTCP in an RTP session is 5% of the session bandwidth. For some applications, it may be appropriate to specify the RTCP bandwidth independently of the session bandwidth. RFC 3556 [10] defines an extension to the SDP to specify RTCP bandwidth for senders and non-senders (receivers).

An IMS terminal shall include the "b=RS:" and "b=RR:" fields in SDP, and shall be able to interpret them. There shall be a limit on the allowed RTCP bandwidth for an RTP session signalled by the terminal. This limit is defined as follows (in 3GPP TS 26.114 [11]; RFC 3556 [10] does not impose limits on the values that may be specified):

- 4 000 bps for the RS field (at media level)
- 3 000 bps for the RR field (at media level)

The optional SDP bandwidth attribute specifies the proposed bandwidth to be used by the session or media by means of the following syntax:

```
b=<modifier>:<bandwidth-value> (in kilobits per second by default)
```

<modifier> is a single alphanumeric word giving the meaning of the bandwidth figure. A typical use is with the modifier "AS" (for *Application Specific* Maximum) which may be used to specify the total bandwidth for a single media stream from one site (source). The new bandwidth attribute modifiers are used here, where "modifier" takes the values "RS" and "RR" respectively.

For the RTP Audio/Video Profile, which specifies that the default RTCP interval algorithm defined in the RTP spec is to be used, at least RS/(RS+RR) of the RTCP bandwidth is dedicated to active data senders. If the proportion of senders to total participants is less than or equal to RS/(RS+RR), each sender gets RS divided by the number of senders. When the proportion of senders is greater than RS/(RS+RR), the senders get their proportion of the sum of these parameters, which means that a sender and a non-sender each get the same allocation. Therefore, it is not possible to constrain the data senders to use less RTCP bandwidth than is allowed for non-senders.

If either or both of the RS and RR bandwidth specifiers are omitted, the default values for these parameters are as specified in the RTP profile in use for the session in question. For the Audio/Video Profile, RFC 3551, the defaults follow the recommendations of the RTP spec.

4.2.4 Other extensions and constricts

(Genaral aspects for the extensions of the following list can be found in [3] [39])

- Update Method: RFC 3311 [40]

  SIP defines the UPDATE method for the initiation and modification of sessions. It is possible that aspects of the session need to be modified before the initial INVITE has been answered, but a re-INVITE cannot be used for this purpose, since this method has an impact on the state of the dialog, in addition to the session. The UPDATE method provides a solution allowing the caller or callee to provide updated session information before a final response to the initial INVITE request is generated, without impacting the dialog state itself.

- SigComp: RFC 3320 [41]

  The SigComp extension defines how to compress SIP textual signaling data, which can be very large and problematic to transmit, causing delay. SigComp solves the challenges of roundtrip delays, as well as mobile user equipment battery life. Typical SIP messages have not been optimized in terms of size (they can take values up to two thousand bytes or more) since they are engineered for bandwidth rich links, that is not the case of 3G cellular networks. SigComp offers robust, lossless compression of application messages that can improve the transmission delays.

- P-headers: RFCs 3455 and 3325 [42] [43]

  In order to solve specific IMS network problems the 3GPP defined, in addition to standard headers, new SIP headers, and submitted them for endorsement to the IETF. Most of them are directly related to the IMS concept as privately owned networks, whose usage is supposed to involve money exchange. These headers have a specific proprietary "P-" status, used in the IETF RFCs in order to indicate that their usage is not recommended for general SIP usage.

- Security Agreement: RFC 3329 [44]

  This IMS SIP extension specifies how to negotiate security capabilities for multiple types of endpoints. The evolution of security mechanisms often

introduces new algorithms, or uncovers problems in existing ones, making negotiation of mechanisms a necessity. The purpose of this specification is to define negotiation functionality for the Session Initiation Protocol (SIP) between a UA and its first-hop SIP entity. Three new SIP header fields, namely Security-Client, Security-Server and Security-Verify, are defined.

- AKA-MD5: RFC 3310 [45]

  This IMS SIP extension determines how terminals and networks are authenticated using mechanism like *IM Subscriber Identity Module* (ISIM), as well as specific key exchange. It specifies *Authorization Key Agreement* and *Message-Digest Algorithm 5* (MD5) based one-time password generation mechanism for HTTP Digest access authentication. AKA is a challenge-response based mechanism that uses symmetric cryptography and it performs user authentication and session key distribution in UMTS networks.

- IPSec: RFC 2401 [46]

  IPSec is used on various IMS interfaces and between different IMS networks in order to provide confidentiality and integrity protection at the network layer. A so-called security association is setup between the nodes that want to exchange secure IPsec-protected traffic, usually by means of *Internet Key Management* (IKE) as key management protocol. The security association contains the security parameters that the nodes use to protect their traffic and it is defined by the address of the nodes and by its *Security Parameter Index* (SPI).

- Mobile Registration: RFCs 3327 and 3608 [47] [48]

  RFC 3608 and RFC 3327 define the syntax and SIP entity usage of the Service-route and Path headers. The REGISTER function is used in a SIP system primarily to associate a temporary contact address with an address-of-record. Any request travelling from the user's home network to the registered must traverse a set of proxies during SIP registration, since the network topology may have one or more SIP proxies between the UA and the registrar. 3GPP established a requirement for discovering intermediate proxies but REGISTER method does not give us a mechanism to discover and record this sequence of proxies in the registrar for future use. Is an extension header field, "Path" which provides such a mechanism.

- IMS and IPv6: RFC 2460 [49]

  IMS prefers IPv6 networks, designed as the successor to IPv4, which offers distinct advantages. It permits a larger set of addresses, the header format is simplified in order to limit its bandwidth cost and it contains embedded

IPSec functionality that may eliminate the need for entities like *Network Address Translation(s)* (NATs) and firewalls.

- Bandwidth negotiation

  The SDP shall include bandwidth information for each media stream and also for the session in total. The bandwidth information for each media stream and for the session is defined by the Application Specific (AS) bandwidth modifier as defined in RFC 4566 [8] (see also A.6).

4.3 RTP/RTCP usage for video transport

According to 3GPP TS 26.114 [11] IMS terminals offering video shall support *Audio-Visual Profile with Feedback* (AVPF) (RFC 4585)[50] configured to operate in early mode, which defines an extension to the *Audio-visual Profile* (AVP) that enables receivers to provide, statistically, more immediate feedback to the senders and thus allows for short-term adaptation and efficient feedback-based repair mechanisms to be implemented, maintaining the AVP bandwidth constraints for RTCP and preserving scalability to large groups. The behaviour can be controlled by allocating enough RTCP bandwidth using "b=RR:" and "b=RS:" (already defined in 4.2.3) and setting the value of "trr-int". The attribute "trr-int" is used to specify the minimum interval T_rr_interval between two Regular (full compound) RTCP packets in milliseconds for a media session. If "trr-int" is not specified, a default value of 0 is assumed.

In order to receive and decode redundant media properly, the receiving application sort the received frames based on the RTP Timestamp and remove duplicated frames. If multiple versions of a frame are received, i.e. encoded with different bitrates, then the frame encoded with the highest bitrate is used for decoding. With the purpose of handle packet losses, AVPF NACK messages are used by IMS terminals to indicate non-received RTP packets for video. An IMS terminal transmitting video can use this information, as well as the AVPF *Picture Loss Indication* (PLI), to accommodate for losses in the encoding process. (see 4.3.1). The reception of both messages give rise to IMS terminal takes appropriate action to improve the situation for the terminal that sent NACK or PLI, although no action is mandated nor specified.

IMS terminals receiving RTCP Receiver Reports (RR) indicating nonzero packet loss should adjust their outgoing bitrate accordingly. Note that for IMS networks, which normally have nonzero packet loss and fairly long round-trip delay, the amount of bitrate reduction specified in RFC 3448 is generally too restrictive

for video and may, if used as specified, result in very low video bitrates already at (for IMS) moderate packet loss rates. 3GPP TS 26.114 [11]) gathers also as requirement for IMS video terminals the support of *Temporary Maximum Media Bit-rate Request* (TMMBR) and *Temporary Maximum Media Bit-rate Notification* (TMMBN) messages of *Codec-Control Messages* (CCM), detailed in 4.3.3.

Regarding media synchronization RTCP *Sender Report* (SR) shall be used by setting the *Network Time Protocol* (NTP) and RTP timestamps according to RFC 3550 [9](Wallclock time (absolute date and time) is represented using the timestamp format of the NTP; RTP timestamp corresponds to the same time as the NTP timestamp, but in the same units and with the same random offset as the RTP timestamps in data packets). To enable quick media synchronization when a new media component is added, or an IMS session is initiated, the RTP sender should send RTCP SR for all newly started media components as early as possible. An IMS sender can signal in SDP that no synchronization between media components is required.

4.3.1 Picture Loss Indication

Video encoding can support inter-coding mode, in which the codec takes advantage of the temporal redundancy of the pictures. That is, it encodes a picture referencing those pictures that were encoded previously. The encoder finds corresponding blocks of pixels between frames and represents their relationship using motion vectors, which are approximated using those motion vectors used in previous frames[3].

A decoder use the *Picture Loss Indication* (PLI) message [50] in order to inform the encoder about the loss of an undefined amount of coded video data belonging to one or more pictures. When used in conjunction with any video coding scheme that is based on inter-picture prediction, an encoder that receives a PLI becomes aware that the prediction chain may be broken. The sender reacts to a PLI by transmitting an intra-picture to achieve resynchronization; however congestion control restricts its ability to send an intra frame.

PLI messages typically set off the sending of full intra-pictures [50]. In this coding mode not references to other pictures are used to encode a particular picture, as a result, intra-pictures are several times larger then predicted (inter-)pictures. Their size is independent of the time they are generated. In most environments, especially when employing bandwidth-limited links, the use of an intra-picture implies an allowed delay that is a significant multitude of the typical frame duration. An example: If the sending frame rate is 10 fps, and an intra-picture is assumed to be

10 times as big as an inter-picture, then a la latency of 1 fps has to be accepted. In such an environment, there is no need for a particular short delay in sending the Feedback message.

4.3.2 Syncronization

In order to grant a good user experience a certain amount of synchronization delay between media streams acceptable to a session is need to be maintained at the receiver side. It takes the name of Synchronization jitter (also known as synchronization or inter-media skew) [11].

Although it is the most widespread use, the Synchronization Info attribute is not just limited to lip-sync between audio/video, but is also applicable to any two media streams that need to be synchronized during a session. This attribute allows a terminal to specify whether or not media streams should be synchronized. Its syntax is defined is as follows:

```
Synchronization-Info = "a" "=" "3gpp_sync_info" ":" sync-value
sync-value = "Sync" / "No Sync"
```

The value "Sync" indicates that synchronization between media shall be maintained. In the same way, the value "No Sync" indicates that No Synchronization is required between the media. The parameter "3gpp_sync_info" should be included in the SDP at the session level and/or at the media level. Its usage is governed by the following rules:

1. At the session level, the "3gpp_sync_info" attribute shall be used with the group attribute defined in RFC 3388 [51]. The group attribute indicates to the receiver which streams (identified by their mid attributes) that are to be synchronized. The "3gpp_sync_info" attribute shall follow the "group: LS" line in the SDP.

   SDP example with requirement on synchronization (session level)

   v=0
   o=Laura 289083124 289083124 IN IP4 one.example.com
   t=0 0
   c=IN IP4 224.2.17.12/127
   a=group:LS 1 2
   a=3gpp_sync_info:Sync
   m=audio 30000 RTP/AVP 0
   a=mid:1

```
m=video 30002 RTP/AVP 31
a=mid:2
m=audio 30004 RTP/AVP 2
i=This media stream contains the Spanish translation
a=mid:3
```

2. At the media level, the "3gpp_sync_info" attribute shall assume a value of "No Sync" only. It indicates to the receiver that this particular media stream is not required to be synchronized with any other media stream in the session.

   SDP example with no requirement on synchronization (media level)

```
v=0
o=Laura 289084412 2890841235 IN IP4 123.124.125.1
s=Demo
c=IN IP4 123.124.125.1
m=video 6000 RTP/AVP 98
a=rtpmap:98 MP4V-ES/90000
a=3gpp_sync_jitter:No Syn
m=video 5000 RTP/AVP 99
a=rtpmap 99 H263-2000/90000
m=audio 7000 RTP/AVP 100
a=rtpmap:100 AMR
```

3. When the "3gpp_sync_info" attribute is defined at both session level (with the "group" attribute) and media level, the media level attribute shall override the session level attribute. Thus if the "3gpp_sync_info" attribute is defined at the media level, then that particular media stream is not to be synchronized with any other media stream in the session (even if the "3gpp_sync_info" is defined at the session level for this media stream). The calling party (or the initiator or offerer of the multimedia stream) should include the "3gpp_sync_info" attribute in the SDP which is carried in the initial INVITE message. Upon reception of the INVITE message that includes the "3gpp_sync_info" attribute, the other party in the session should include its own "3gpp_sync_info" attribute (with its own wish for synchronization or no synchronization) in the 200/OK response message.

4. Default operation in the absence of the "3gpp_sync_info" attribute in SDP is to maintain synchronization between media streams.

4.3.3 Video adaptation

It is recommended that a video sender adapts its video output rate based on RTCP reports and TMMBR messages [11]. When the receiver is made aware of a reduction in downlink bandwidth capacity (e.g. due to changed radio conditions) it shall notify the sender of the new current maximum bitrate using TMMBR. TMMBR is used to signal temporary bitrate changes. The sending client, receiving TMMBR, shall respond by sending TMMBN , as described in CCM.

If the bandwidth changes further, or goes back to normal, the receiver should notify the sender by sending a new TMMBR message. However, if the changed bandwidth is likely to remain, a SIP re-negotiating should be initiated aiming at establishing the new rate. It is important that the receiver does not use TMMBR to make the sender reduce its bitrate due to the amount of packet losses or jitter experienced by the receiver, since this kind of information shall be reported in regular RTCP reports.

4.3.4 Video Bit Rate equalization in IMS - Circuit Switched interworking

Temporary video rate variations can occur on the IMS side for example due to congestion. The video rate on the circuit switched side, in contrast, is under full control of the circuit switched side UE and the *Media Gateway* (MGW). During session setup, the MGW shall negotiate a video bitrate on the IMS side that allows all video bits to be conveyed to/from the circuit switched link [11].

A buffer shall be maintained in the direction from the IMS to the circuit switched side. The size of the buffer should be kept small enough to allow for a low end-to-end delay, yet large enough to conceal most network jitter on the IMS side. Temporary uneven traffic on the IMS side, beyond the handling capability of the buffer, should be handled as follows: if the buffer overflows, RTP packets should be dropped and the resulting loss and observed jitter should be reported by the means of an RTCP RR at the earliest possible sending time. The drop strategy may preferably be implemented media aware (i.e. favouring dropping predicted information over non-predicted information and similar techniques), or may be drop-head. If the buffer runs empty, the circuit switched side should insert appropriate flag stuffing.

A buffer shall be maintained in the direction from the circuit switched to the IMS side. The size of the buffer should be kept small enough to allow for a low end-to-end delay, but large enough to conceal most network jitter on the circuit

switched side. If the buffer overflows, then video bits must be dropped, preferably in a media-aware fashion. MGCs may also take into account the type of media data, i.e. coded with or without prediction. If overflows occur frequently, the MGW may attempt to reduce the sending rate of the circuit switched UE by employing H.245's FlowControlCommand. When the buffer runs empty, no activity is required on the IMS side.

If the bandwidth resources on the IMS side during a significant period of time drops below the limit where all video bits from the circuit switched side can be forwarded, the MGW should drop the video component on the IMS side and change the circuit switched call to a voice-only call.The MGW should avoid dropping the entire call, so if the procedures in are not available or feasible, the circuit switched video call may be kept with the video component muted. If the video component was muted in the MGW for this reason and the available bandwidth on the IMS side increases, the MGW should restore the video component on the IMS side and un-mute the video on the circuit switched side. If the circuit switched video call is changed to a voice-only call, the video component on the IMS side shall be dropped.

Chapter 5

Session setup flows for QoS Provision

Taking the example used in 3GPP TR 24.930 V7.1.0 [22] it is time now to go in deep in the SIP/SDP messages involved. In the previous sections, an overview of some SIP/SDP extensions took place, showing the possibilities that such protocols give to IMS in order to perform appropriated multimedia sessions. Now, the messages will be detailed step by step, trying to allow the reader a good understanding.

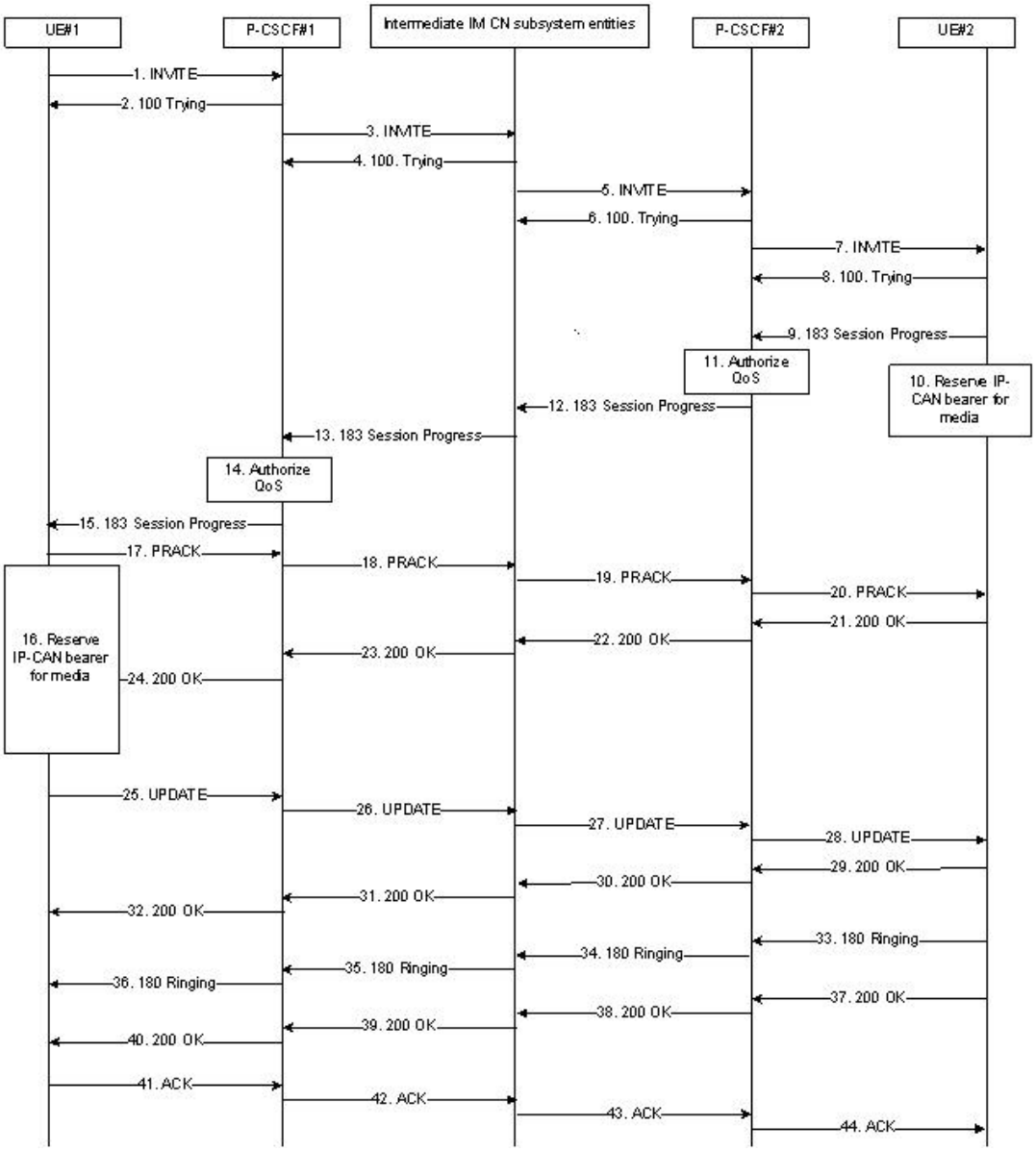


Figure 5.1: General Flow

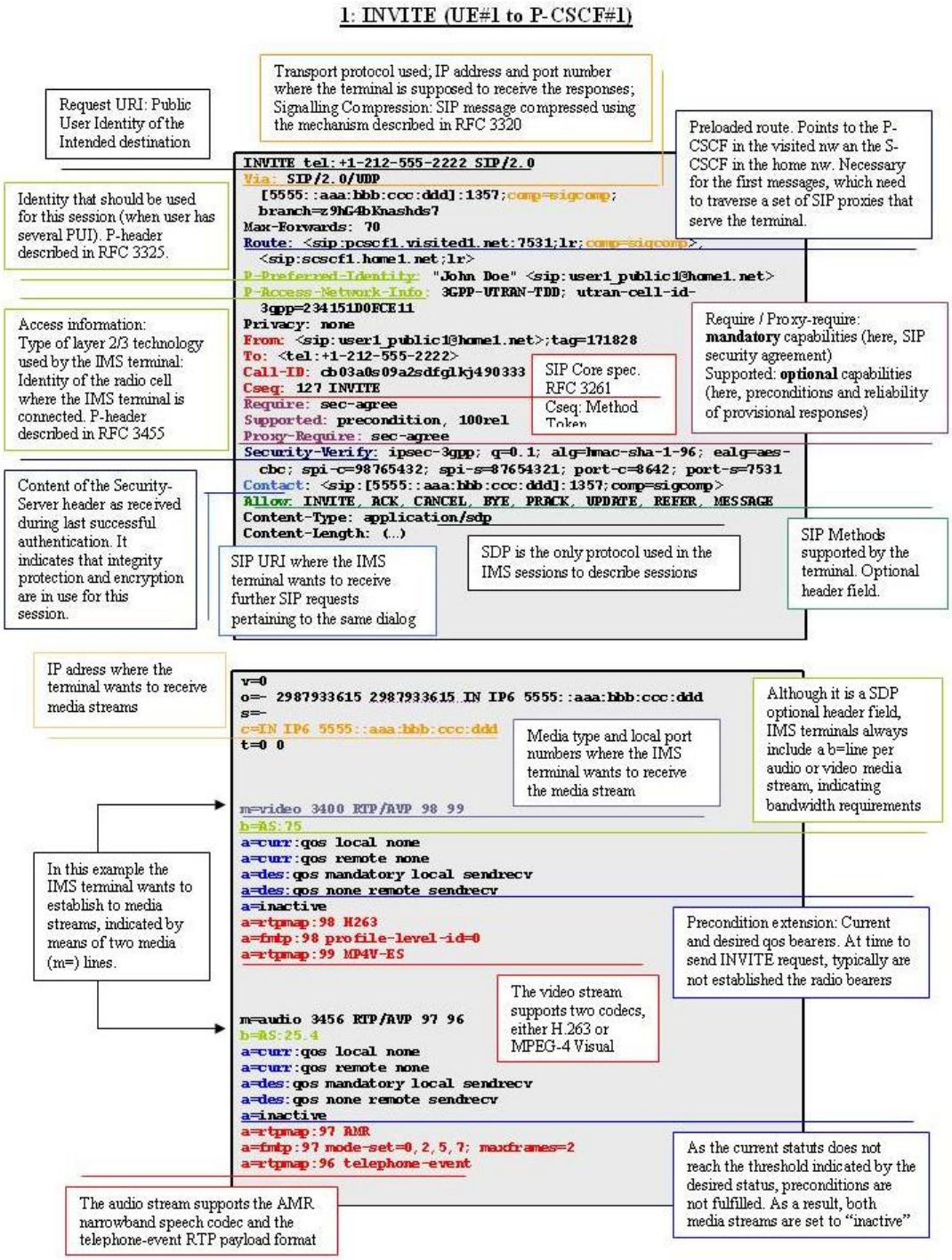


Figure 5.2: (1) Invite

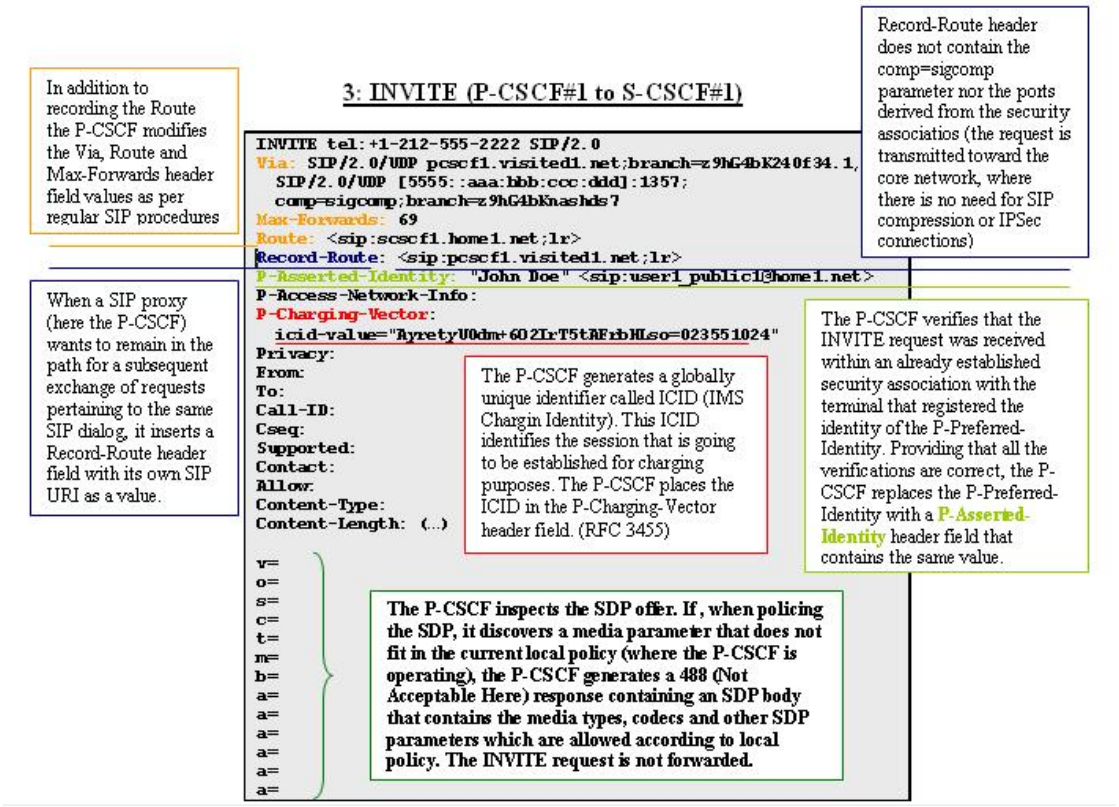


Figure 5.3: (3) Invite

Intermediate IM CN Subsystem entities flows

For the sake of simplicity the description flow chart uses the box "Intermediate IM CN Subsystem entities" that stands for the combination I-CSCF/S-CSCF on the originating and on the terminating side. Routing of messages between those nodes (and the *Home Subscriber Server* (HSS)) is described in the flow below (indicated with a-f the new flows added) [3]:

The S-CSCF allocated to the caller receives the INVITE request and, after the examination of the P-Asserted-Identity header to identify the originating user, evaluates the initial filter criteria [3]. The filter criteria, which contains the collection

of triggers that determine whether a request has to traverse one or more Application Servers, is downloaded as a part of the user profile at registration. S-CSCF evaluates initial filter criteria when receiving initial request in a dialog (INVITE, SUBSCRIBE) or a standalone request (MESSAGE, OPTIONS). S-CSCF does not evaluate on PRACK, NOTIFY, UPDATE or BYE. On registration the S-CSCF

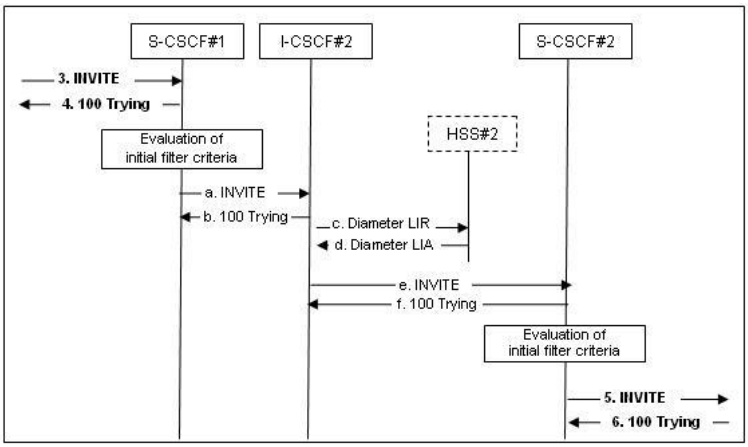


Figure 5.4: Intermediate IM CN Subsystem entities flows

receives a user profile from HSS, a data structure where all the data related to a user is stored. The user profile contains the Private User Identity to which the user profile is applicable and one or more service profiles. Each service profile contains one ore more Public User Identities to which the service profile is applicable and zero or more filter criteria. The figure 5.4 depicts a simplified representation of the structure of the user profile. Filter criteria determine the services that are applicable to the collection of Public User Identities of the profile.

The S-CSCF assesses the criteria in the order of their priority compared with the remaining filter criteria that are part of the same service profile [3]. Filter criteria contain trigger points, which are Boolean conditions that determine whether the SIP request will be forwarded to an *Application Server* (AS). If the trigger point fires, the request goes to the corresponding AS. After receiving back the request the next criteria is checked.

The S-CSCF, like the P-CSCF, polices those SDP media parameters that are not set according to local policy. Unlike the P-CSCF the S-CSCF gets from the HSS the user profile previously described. In the case the request does not fit with the policy, the S-CSCF may not process the INVITE request and answer with a 488 (Not Acceptable Here) response.

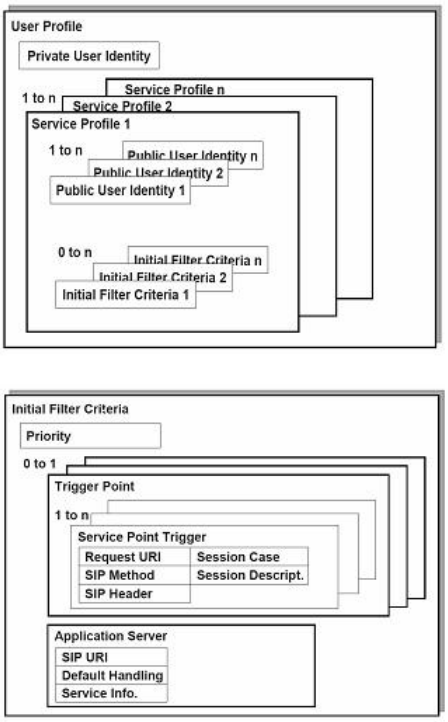


Figure 5.5: Initial filter criteria

The S-CSCF1 is the first node that tries to route the SIP request based on the Request-URI. If it finds a SIP URI, then regular SIP routing procedures apply. Basically, given a domain name the S-CSCF has to find a SIP server in that domain name (procedures described in RFC 3263. In the current example, however, the Request-URI contains a TEL *Uniform Resource Locator* (URL) (RFC 2806). The S-CSCF, beinga SIP proxy server, implements routing based on SIP, and not on telephone numbers. So first tries to map the TEL URI into a SIP URI, typically by means of another DNS service: the ENUM service (RFC 2916). If the telephone number is not associated with a SIP URI, there is no mapping to DNS, so DNS returns a negative response, indicating that no record was available for that telephone number. This is an indication that the user is not an IMS user nor a user defined at any other SIP domain. The S-CSCF is unable to forward the SIP request to its destination. Instead, the S-CSCF requires the services of a *Breakout Gateway Control Function* (BGCF).

The S-CSCF finds, through DNS procedures, a SIP server in the destination home network, in the IMS such server is the I-CSCF. I-CSCF has to forward the INVITE request to the S-CSCF allocated to the callee but, being not aware of its address, I-CSCF has to discover it querying the HSS with a Diameter *Location-Information-Request* (LIR) message. The HSS receives the Diameter LIR request, processes it and generates a Diameter *Location-Information-Answer* (LIA) message. Once the I-CSCF receives the Diameter LIA message, it knows where to route the INVITE request, whose routing was temporarily suspended. The I-CSCF at this stage does not modify or add any SIP header field, other than the SIP routing header fields.

By the same way, S-CSCF2 evaluates the initial filter criteria of the called user. The S-CSCF creates a new Request-URI with the contents of the Contact header value registered by the callee during the registration. This procedure is known as retarget.

5: INVITE (S-CSCF#2 to P-CSCF#2)

The Request URI is now the SIP URI of the Contact header field.

```
INVITE sip:[5555::eee:fff:aaa:bbb]:8805;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf2.home2.net;branch=z9hG4bK764z87.1,
   SIP/2.0/UDP icscf2_s.home2.net;branch=z9hG4bK871y12.1,
   SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1,
   SIP/2.0/UDP pcscf1.visited1.net;branch=z9hG4bK240f34.1,
   SIP/2.0/UDP
   [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashd
   s7
Max-Forwards: 66
Route: <sip:pcscf2.visited2.net;lr>
Record-Route: <sip:scscf2.home2.net;lr>,
   <sip:scscf1.home1.net;lr>, <sip:pcscf1.visited1.net;lr>
P-Asserted-Identity:
P-Charging-Vector: icid-
   value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"
Privacy:
From:
To:
Call-ID:
Cseq:
Require:
Supported:
Contact:
Allow:
P-Called-Party-ID: <sip:user2_public1@home2.net>
Content-Type:
Content-Length: (...)

v=
o=
s=
c=
t=
m=
b=
a=
a=
a=
a=
a=
a=
a=
```

The Via header field keeps track of all the proxies a request has traversed. The response uses these Via entries so that it traverses the same proxies as the request did in the opposite direction

(3GPP extensions to SIP RFC 3455). When the S-CSCF re-targets, it also inserts a P-Called-Party-ID header field, whose value is set to the original Request-URI., in order to allow the IMS terminal, after the inspection of this header content, be able to determine to which of the various identities the INVITE request is destined.

Figure 5.6: (5) Invite

7: INVITE (P-CSCF#2 to UE#2)

```
INVITE sip:user2_public1@home2.net SIP/2.0
Via: SIP/2.0/UDP
    pcscf2.visited2.net:5088;comp=sigcomp;branch=z9hG4bK361k21.1,
    SIP/2.0/UDP scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP
    icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
    scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
    pcscf1.visited1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
    [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
Max-Forwards: 67
Route: <sip:scscf2.home2.net;lr>
Record-Route: <sip:pcscf2.visited2.net:5088;lr;comp=sigcomp>,
    <sip:scscf2.home2.net;lr>, <sip:scscf1.home1.net;lr>,
    <sip:pcscf1.visited1.net;lr>
P-Asserted-Identity:
Privacy:
P-Media-Authorization:
    00200001001001017064663 12e686f6d65312e6e6574000c02013331533134363231
From:
To:
Call-ID:
Cseq:
Require:
Supported:
Contact:
Allow:
Content-Type:
Content-Length: (...)

v=
o=
s=
c=
t=
m=
b=
a=
a=
a=
a=
a=
a=
```

Request-URI already contains a SIP URI that includes the IP address (or host name) of the IMS terminal. The PUI of the callee is extracted by the P-CSCF from the P-Called-Party-ID header field

P-CSCF#2 adds its own SIP URI to the Record-Route header field value

The P-CSCF inserts a P-Media-Authorization header field. The terminal adds this token to the messages it sends to reserve network resources

Figure 5.7: (7) Invite

9: 183 Session Progress (UE#2 to P-CSCF#2)

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP
    pcscf2.visited2.net:5088;comp=sigcomp;branch=z9hG4bK361k21.1,
    SIP/2.0/UDP scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP
    icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
    scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
    pcscf1.visited1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
    [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
Record-Route: <sip:pcscf2.visited2.net:5088;lr;comp=sigcomp>,
    <sip:scscf2.home2.net;lr>, <sip:scscf1.home1.net;lr>,
    <sip:pcscf1.visited1.net;lr>
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11
Privacy: none
From:
To: <tel:+1-212-555-2222>;tag=314159
Call-ID:
CSeq:
Require: 100rel, precondition
Contact: <sip:[5555::eee:fff:aaa:bbb]:8805;comp=sigcomp>
Allow: INVITE, ACK, CANCEL, BYE, PRACK, UPDATE, REFER, MESSAGE
Supported: 9021
Content-Type: application/sdp
Content-Length: (...)
```

The IMS terminal inserts a P-Access-Network-Info header field.

Require: **mandatory** capabilities. (Here, preconditions and reliability of provisional responses)

Contact header field: SIP URI that includes the IP address or Fully Qualified Domain Name of the IMS terminal.

Precondition extension requires that provisional responses be transmitted reliably. This is the reason the callee's IMS terminal inserts a require header field with the tag 100rel in the 183 (Session Progress) provisional response.

```
v=0
o=- 2987933623 2987933623 IN IP6 5555::eee:fff:aaa:bbb
s=-
c=IN IP6 5555::eee:fff:aaa:bbb
t=0 0
m=video 10001 RTP/AVP 98
b=AS:75
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=inactive
a=conf:qos remote sendrecv
a=rtpmap:98 H263
a=fmtp:98 profile-level-id=0
m=audio 6544 RTP/AVP 97 96
b=AS:25.4
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=inactive
a=conf:qos remote sendrecv
a=rtpmap:97 AMR
a=fmtp:97 mode-set=0,2,5,7; maxframes=2
a=rtpmap:96 telephone-event
```

UE#2 uses a conf line in the SDP to request a confirmation from UE#1 (by means of an updated SDP) when the local resources are available at UE#1.

When receiving the INVITE request the IMS terminal (UE#2) determines the complete set of codecs that it is capable of supporting for this session. It determines the intersection with those appearing in the SDP in the INVITE request and it makes the final codec selection (in this example UE#2 chooses H.263 and AMR)

Figure 5.8: (9) Session Progress

12: 183 Session Progress (P-CSCF#2 to S-CSCF#2)

(Forwarded in steps 13 and 15)

P-CSCF forwards the 183 (Session Progress) response based on the contents of the value of the topmost Via header field, to the S-CSCF

P-CSCF verifies that Via and Recor-Route header field contain the values that the IMS terminal should have used in a response to the INVITE request.

```
SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP scscf2.home2.net;branch=z9hG4bK764z87.1, SIP/2.0/UDP
    icscf2_s.home2.net;branch=z9hG4bK871y12.1, SIP/2.0/UDP
    scscf1.home1.net;branch=z9hG4bK332b23.1, SIP/2.0/UDP
    pcscf1.visited1.net;branch=z9hG4bK240f34.1, SIP/2.0/UDP
    [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7)
Record-Route:
P-Asserted-Identity: "John Smith" <sip:user2_public1@home2.net>
P-Access-Network-Info:
P-Charging-Vector: icid-value="AyretyU0dm+602IrT5tAFrbHLso=023551(
Privacy:
From:
To:
Call-ID:
CSeq:
Require:
Contact:
Allow:
RSeq:
Content-Type:
Content-Length:

v=
o=
s=
c=
t=
m=
b=
a=
a=
a=
a=
a=
a=
a=
a=
m=
```

P-Asserted-Identity header field whose value is the same as that included in the P-Called-Party-ID header field of the INVITE request.. Allows the nodes get the PUI of the callee being used for this session.

Figure 5.9: (12) Session Progress

17-24. PRACK request / 200(OK) exchange

Due to the presence of the 100rel tag in the Require header field, the 183 (Session Progress) response requires to be acknowledged [3]. For this reason, the caller's IMS terminal creates a PRACK request that will be answered by a 200 (OK) response that should not be confused with a 200 (OK) for the INVITE that will be occur later. The PRACK request does not carry SDP as the final codec decision is already made as part of the initial offer/answer exchange.

In parallel with the generation of the PRACK request the IMS terminal starts the resource reservation mechanisms, dependent on the underlying IP CAN.

The PRACK request traverses all the proxies that asked to remain in the path for subsequent signalling by adding their own SIP URIs to the Record-Route header field on the INVITE Request. The value of the Record-Route included in the 183 (Session Progress) response sets then the path of the Route header field in the PRACK request, which will be a subset of the proxies that the INVITE request traversed.

On receiving the PRACK request, the callee's IMS terminal generates a 200 (OK) response that traverses the same set of proxies that the PRACK request traversed. At the same time, the callee starts resource reservation in its own segment.

25. UPDATE request (UE#1 to P-CSCF#1) (forwarded in steps 26-28)

```
UPDATE <sip:[5555::eee:fff:aaa:bbb]:8805;comp=sigcomp
Via: SIP/2.0/UDP [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashds7
Max-Forwards: 70
Route: <sip:pcscf1.visited1.net:7531;lr;comp=sigcomp>, <sip:orig@scscf1.home1.net;lr>
From: <sip:user1_public1@home1.net>; tag=171828
To: <tel:+12125552222> tag=314159
Call-ID: cb03a0s09a2sdfglkj490333
Cseq: 129 UPDATE
Require: sec-agree
Proxy-Require: sec-agree
Security-Verify: ipsec-3gpp; q=0.1; alg=hmac-sha-1-96; ealg=aes-cbc; spi-c=98765432;
   spi-s=87654321; port-c=8642; port-s=7531
Contact: <sip:[5555::aaa:bbb:ccc:ddd]:1357;comp=sigcompContent-Type: application/sdp
Content-Length: (...)

v=0
o=- 2987933615 2987933615 IN IP6 5555::aaa:bbb:ccc:ddd
s=-
c=IN IP6 5555::aaa:bbb:ccc:ddd
t=0 0
m=video 3400 RTP/AVP 98
b=AS:75
a=curr:qos local sendrececv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=sendrecv
a=rtpmap:98 H263
a=fmtp:98 profile-level-id=0
m=audio 3456 RTP/AVP 97 96
b=AS:25.4
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=sendrecv
a=rtpmap:97 AMR
a=fmtp:97 mode-set=0,2,5,7; maxframes
```

Once the caller has got the required resources from the network it sends an UPDATE request, due to the request for a confirmation message (a=conf lines in the SDP answer).

In the new SDP offer the terminal indicates that resources are reserved at his **local** segment.

Thus, UE#1 indicates that it can send and receive media as the necessary resources are available.

Figure 5.10: (25) Update

29. 200 (OK) response (UE#2 to P-CSCF#1)
(Forwarded in steps 30-32)

UE acknowledges the UPDATE request with a 200 OK response.

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP
  pcscf2.visited2.net:5088;comp=sigcomp;branch=z9hG4bK361k21.1,
  SIP/2.0/UDP scscf2.home2.net;branch=z9hG4bK764z87.1,
  SIP/2.0/UDP icscf2_s.home2.net;branch=z9hG4bK871y12.1,
  SIP/2.0/UDP scscf1.home1.net;branch=z9hG4bK332b23.1,
  SIP/2.0/UDP pcscf1.visited1.net;branch=z9hG4bK240f34.1,
  SIP/2.0/UDP
  [5555::aaa:bbb:ccc:ddd]:1357;comp=sigcomp;branch=z9hG4bKnashd
  s7
P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-
  3gpp=234151D0FCE11
From: <sip:user1_public1@home1.net>;tag=171828
To: <tel:+12125552222>;tag=314159
Call-ID: cb03a0s09a2sdfglkj490333
Cseq: 129 UPDATE
Content-Type: application/sdp
Content-Length: (...)

v=0
o=- 2987933615 2987933615 IN IP6 5555::eee:fff:aaa:bbb
s=-
c=IN IP6 5555::eee:fff:aaa:bbb
t=0 0
m=video 3400 RTP/AVP 98
b=AS:75
a=curr:qos local sendrecevc
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=rtpmap:98 H263
a=fmtp:98 profile-level-id=0
m=audio 3456 RTP/AVP 97 96
b=AS:25.4
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=rtpmap:97 AMR
a=fmtp:97 mode-set=0,2,5,7; maxframes
```

At this stage the callee's IMS Terminal may have already finished its resource reservation or not, depending on how much time it takes to complete the process. UE#2 indicates its local QoS status, which may either be complete or not.

Remote (UE#1) resources are available, as indicated the previous UPDATE request.

Local (UE#2) resources are available

Figure 5.11: (29) 200 (OK)

33 -36 . 180 (Ringing) response

Before the callee is alerted two conditions have to be met. Terminal needs to complete its local resource reservation process and receive an UPDATE request (remote resource process finished successfully). With this two conditions fulfilled, the terminal is aware that the resources are available at both sides of the session.

UE2 indicates that it is ringing generating a 180 (Ringing) provisional response, which traverses those proxies that the INVITE request traversed. The UE2 does not use Require "100rel" as the 180 (Ringing) does not have a SDP and therefore need not to be sent reliable. The no presence of a SDP is due to the fact that all the session parameters have already been negotiated in the previous exchanges.

37 -40 .200 (OK) response and 40-44 ACK request

When the called party answers the UE sends a 200 OK final response to the INVITE request to P-CSCF, indicating it is ready for start the media flow(s) for this session. The calling party responds to the 200 (OK) response with an ACK request. At this stage the caller's IMS terminal starts generating media-plane traffic.

Since session setup is complete, both users can generate and share their audio and video media streams, which are in general sent end to end via IP-CAN routers.

Part II

Practical Work

Chapter 6

Work environment

6.1 Previous: IMS Arquitecture

Figure 6.1 shows the basic IMS architecture.

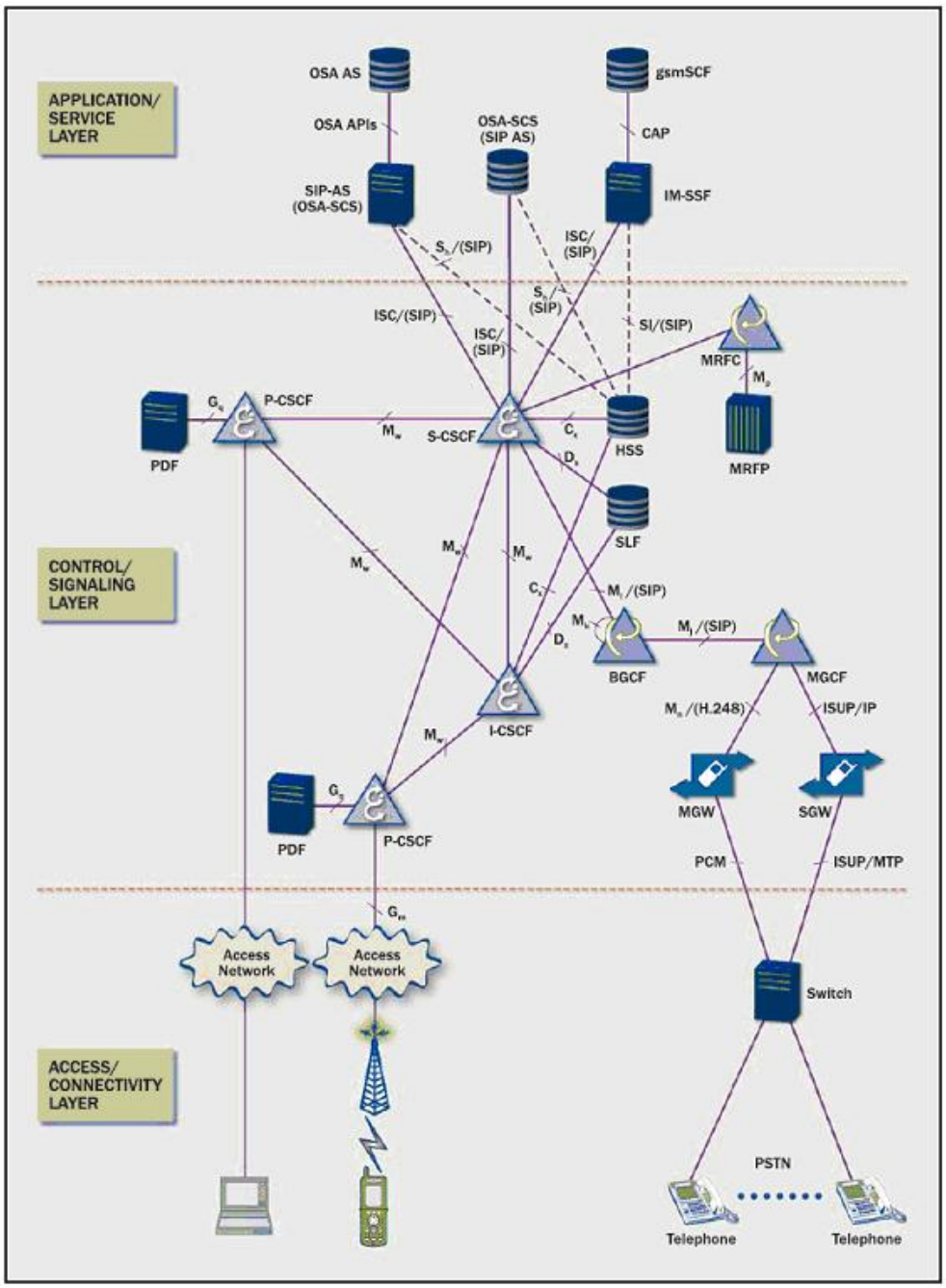


Figure 6.1: ims architecture

The key functional entity is the CSCF (Call State Control Function), basically a SIP server (since it processes SIP signalling) with proxy functionalities. The CSCF performs three different roles within the IMS operative:

- **Proxy-CSCF:** Entrance point to IMS which directly receives the IMS signalling from the terminal. It implements security functions, providing data integrity and confidentiality by using IP security (IPSec [46]; see also 4.2.4), as a result, it allows in this way a security association between the IMS network and each terminal. It also provides QoS authorization and control by providing the necessary information to the PDF for resource authorization and QoS control in the transport subsystem (see chapter 3).

  The P-CSCF handles the charging records for billing purposes by creating and maintaining a charging data record. In a roaming scenario it is the node in the visited network that deals with the register and session signalling routing from the terminals towards the native network. Furthermore, it performs de common functions shared with the rest of CSCF: it processes and routes signalling queries the *Home Subscriber Server* (HSS) about user's profile and performs charging functionalities.

  IMS networks usually have several P-CSCFs for the sake of scalability and redundancy, and each P-CSCF serves a certain number of IMS terminals, based on its capacity.

- **Serving-CSCF:** Each registered user has an S-CSCF assigned, which routes the sessions initiated by or destined for the user, that is, it controls the session. CSCF is essentially a SIP proxy that relays SIP messages, a SIP UA that initiates and terminates SIP transactions, and a SIP registrar that authenticates users during registration. The S-CSCF interfaces with the HSS via Diameter protocol and downloads the user's profile and authentication vector to be used in user authentication. Such information is used to control user's access (applying operator policies) to different Application Servers (AS), allowing IMS services provision. As well, the S-CSCF also collects data for charging purposes

  An IMS network may have several S-CSCFs for the sake of scalability and redundancy. Both the S-CSCF and the P-CSCF maintain session timers, that is to say, they are stateful proxies.

- **Interrogating-CSCF:** Intermediate proxy server node located at the edge of an administrative domain that gives support to the IMS and that interfaces (based on Diameter protocol) with *Subscriber Local Function* (SLF)

and HSS databases. It retrieves user location information and helps the other nodes to route the next SIP request step, just as to establish a signalling path. During registering, the P-CSCF relies upon the I-CSCF in order to determine which S-CSCF has to serve each user. The assignment is based on the information queried from the HSS through the Diameter-based Cx interface.

Optionally, in roaming scenarios and interoperator sessions, the I-CSCF hides the IMS topology to external networks, in order to avoid them discovering how the internal signalling is managed or accessing sensitive information, such as the number, names or capacities of the CSCFs involved). An IMS network may have several I-CSCFs for the sake of scalability and redundancy.

Other relevant nodes are the next:

- **Home Subscriber Server (HSS):** It Inherits the *Home Location Register* (HLR) functionalities: It holds and manages the subscriber's IMS service profile. It holds security keys and generates the authentication vectors, it holds location information, user profile *Service Switching Point* (SSP), trigger points and filter criteria, it registers the subscribers' state and stores the S-CSCF assigned to each user.

  Subscriber data stored in the HSS is the key enabler for service mobility across different types of access networks and for user roaming between different network operators. A network may require more than one HSS due to the number of subscribers and the capacity of the HSS. Because of its importance, the HSS is always implemented in redundant configuration.

- **Media Gateway Control Function (MGCF):** it is part of the interworking architecture between the IMS and the circuit switched networks, being the main node of the *Public Switched Telephone Network / Circuit Switched* (PSTN/CS) gateway. It implements the control plane, translating IMS SIP/SDP signalling to *Signalling System 7* (SS7) and vice versa. It is also in charge of controlling the IM-MGW operation.

- **IP Multimedia Media Gateway (IM-MGW):** It implements the user's plane in the interworking architecture with the circuit switched networks. Over *Time-Division Multiplex* (TDM) networks it is in charge of transcoding IMS flows over IP towards user data.

- **Application servers and service plane gateways:** 3GPP defines interfaces IMS between the S-CSCF and the service plane, by this way signalling

can be forwarded towards the service plane taking into account subscriber's profile criteria, stored in the HSS and downloaded by the S-CSCF during registering each user.

- **Subscriber Locator Function (SLF):** A Diameter-based redirect agent or server that maps the user's address to a specific HSS. A network with a single HSS does not require a SLF.
- **Policy Decision Function (PDF):** It may be part of the P-CSCF or a standalone entity. It interacts with the P-CSCF via the Diameter-based Gq interface and with the PEP at the *Packet Data Gateway* (PDG) via the COPS-based Go interface. The PDG for the GPRS/UMTS network is the GGSN.
- **Application Server (AS):** A SIP entity that hosts and executes services. New IMS-specific services are expected to be developed in SIP ASs. An AS may host several different applications.
- **Breakout Gateway Control Function (BGCF):** It is the SIP server with routing functionality based on telephone numbers. It is used in sessions that are initiated by IMS terminals and addressed to users in a CS network such as PSTN or other cellular network. The BGCF's main functionality is to select an appropriate PSTN/CS gateway.

6.2 Open Source IMS

The Fraunhofer Institute FOKUS has launched the "Open IMS Playground" in July 2004 and approximately two years later, in November 2006 the *Open Source IMS Core* (OSIMS Core) was ready for be downloaded under a GNU General Public License on the FOKUS BerliOS (The Open Source Mediator).

> "The Open Source IMS Core project aims to fill the currently existing IMS void in the Open Source Software landscape with a flexible and extendable solution that has already proven its conformance and performance in several national and international R&D projects [...].The idea for users of this Open Source software is to enable the development of IMS services and the trial of concepts around core IMS elements.".
>
> "Also towards the access network layer, the Open Source IMS Core enables the development of components and concepts that come with the attachment of various access networks to the overlay architecture IMS."

Figures 6.2 and 6.3 depict the Open Source IMS Core Components and its place in the Open IMS Playground.

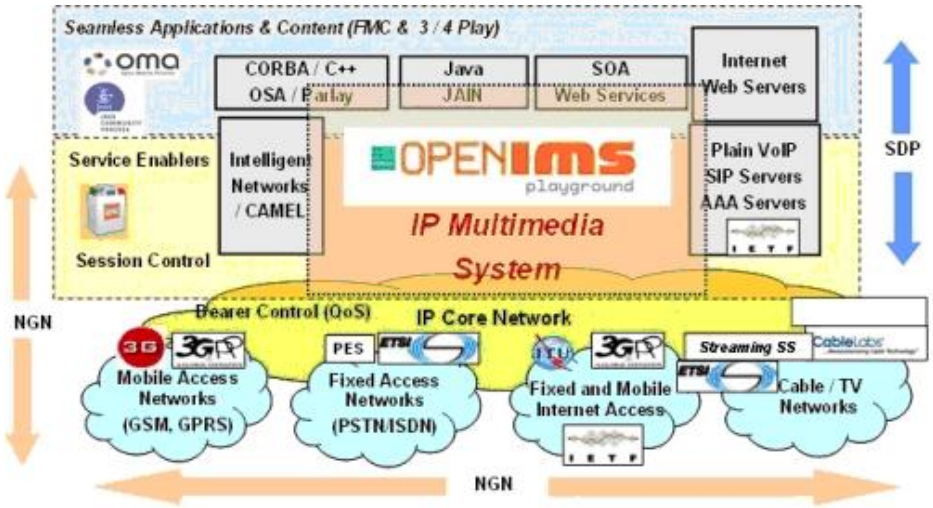


Figure 6.2: Overview of IMS and NGN at FOKUS

The central components of the Open Source IMS Core project, being them the central routing elements for any SIP signalling, are the Open IMS CSCFs (Proxy, Interrogating, and Serving) which were developed at FOKUS as extensions to the *SIP Express Router* (SER).

The *FOKUS Home Subscriber Server* (FHoSS) is also part of the OSIMS Core project, since it is necessary a HSS to manage user profiles and associated routing rules. The features of each component within the OSIMS project are gathered next:

- **Proxy-CSCF: OSIMS features**
  - signaling firewall and user identity assertion (P-Preferred-Identity, P-Asserted-Identity header support)
  - local registrar synchronization through "reg" event RFC 3680
  - Path header support
  - Service-Route verification/enforcement
  - Dialog statefulness and Record-Route verification/enforcement
  - IPSec setup using *Cipher Key* CK and *Integrity Key*IK from AKA
  - Integrity-protection for authentication

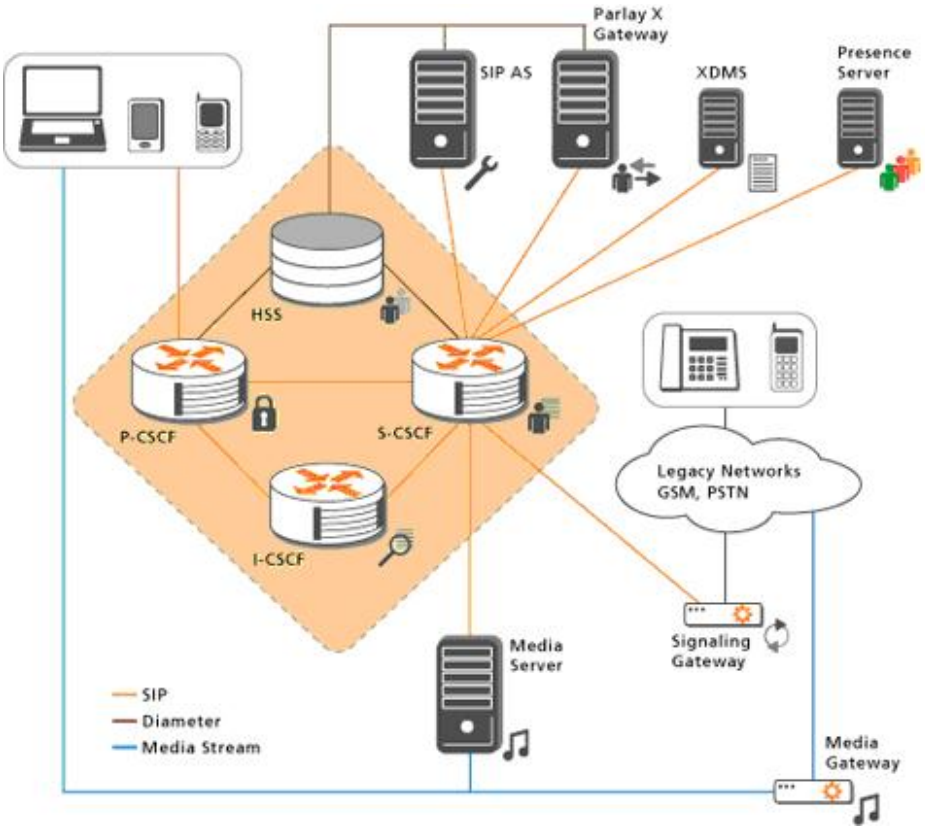


Figure 6.3: Open Source IMS Core in the Open IMS Playground

- Security-Client, Security-Server, Security-Verify header support
- basic P-Charging-Vector support
- Visited-Network-ID header support
- NAT support for signaling
- NAT support for media through RTPProxy

- **Interrogating-CSCF: OSIMS features**
  - full Cx interface support (LIR, UAR)
  - S-CSCF selection based on user capabilities
  - Serial forking for forwarding to S-CSCF
  - Visited-Network-ID header support and roaming
  - Permission verification
  - Topology Hiding Interwork Gateway (THIG)

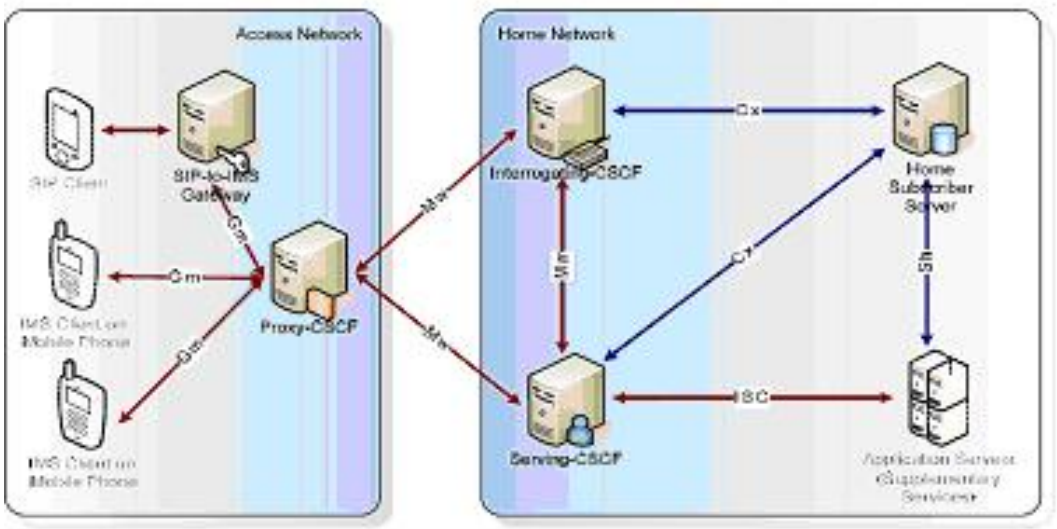


Figure 6.4: Open Source IMS Core Components

  - Network Domain Security (NDS)

- **Serving-CSCF**
  - full Cx interface support
  - Authentication through AKAv1-MD5, AKAv2-MD5 and MD5
  - Service-Route header support
  - Path header support
  - P-Asserted-Identity header support
  - Visited-Network-ID header support
  - Download of Service-Profile from HSS
  - Initial Filter Criteria triggering
  - ISC interface routing towards Application Servers
  - ”reg” event server with access restrictions
  - Dialog statefulness

- **Home Subscriber Server**
  - support for the 3GPP Cx Diameter application
  - support for the 3GPP Sh Diameter application

  - support for the 3GPP Zh Diameter application
  - integrated simple AuC functionality
  - Java Diameter Stack implementation
  - web-based management console

- **SIP2IMS Gateway**
  - allows transformation of IETF SIP messages to IMS conformant messages
  - Translates MD5 authentication to IMS AKA authentication

In order to end the OSIMS overview, and inviting the reader to visit the FOKUS OSIMS homepage for extended and accurate information and documentation, next a synopsis of Specification Guidelines for this Open Source.

- IETF RFCs:
  - SIP: Session Initiation Protocol - RFC 3261
  - Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA) - RFC 3310
  - SIP Private Header Extensions - RFC3455
  - Diameter Base Protocol - RFC 3588
  - SIP Event Package for Registration - RFC3680
- 3GPP IMS Release 6 Specifications (selection)
  - TS 23.228 - IMS Stage 2 (Rel.6)
  - TS 24.229 - IP Multimedia Call Control Protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage3 (Rel.6)
  - TS 29.228 - Cx and Dx Interfaces, Signalling flows and message contents (Rel.6)
  - TS 33.102 - 3G Security; Security architecture (Rel. 6)
  - TS 33.203 - Access security for IP-based services (Rel.6)

6.3 Looking for an IMS client

The search for a proper IMS client that allows analyzing the testbed behaviour regarding QoS related aspects has given rise, finally, to a large set of possible clients, most of them, understood as SIP clients rather than IMS ones. The tables gather in Appendix C summarize which are these clients and some particularity of each one.

IMS-Communicator has been the final choice, due basically to its Precondition mechanisms support, a feature with considerable impact in the session initiation stage for to ensure and negotiate QoS Provision.

IMS-Communicator, developed by Portugal Telecom Inovacao is a SIP softphone based on the old version of the sip-communicator java project[1], implemented on top of the JAIN-SIP stack[2] and the Java Media Framework API[3]. Originally, the SIP-Communicator client supported normal SIP registration of one public user identity, voice and video session initiation, and some IM functionality, on Windows and Linux.

Some changes have been made to the JAIN-SIP stack to support IMS, such as new SIP headers and SDP functionalities, contribution that have been submitted to the community and is already shipped within the latest JAIN-SIP stack. The main

IMS UE procedures were implemented, as described in the 3GPP specifications TS 24.229 [35], TS 23.228 [18] and others (check http://www.tech-invite.com/ for IMS call flows examples). Some of the IMS functionalities implemented include IMS Registration (Authorization, Security Agreement and Subscription to the reg event package), IMS session initiation (PRACK, UPDATE and Precondition Mechanism) and Call Transfer.

[1] http://sip-communicator.org/
[2] https://jain-sip.dev.java.net/
[3] http://java.sun.com/products/java-media/jmf/

Leaving the configuration process of the softphone chosen for the next chapter, it only remains to overview the IMS-Communicator supported standards and features, listed both as follows:

Supported Standards list

- SIP and SDP support (modified JAIN-SIP stack to support IMS)
- Procedures at the UE (SIP and SDP) - 3GPP TS 24.229
- SIP PRACK method (RFC 3262)
- SIP UPDATE method (RFC 3311)
- SIP Precondition Mechanism (RFC 3312 + RFC 4032)
- A SIP Event Package for Registrations (RFC 3680)
- SIP Security Agreement (RFC 3329 + 3GPP TS 33.203 Annex H)
- SIP REFER method (RFC 3515) and Session Transfer Procedures (3GPP TS 23.228)
- SIP Referred-By Mechanism (RFC 3892)
- 3GPP TS 24.229 - IP multimedia call control protocol based on SIP and SDP
- HTTP AKA (RFC 3310)
- HTTP Digest Authentication (RFC 2617)
- MILENAGE 3GPP Authentication Algorithm (TS 35.205, TS 35.206, TS 35.207, TS 35.208)

Supported features list

- Setup wizard
- IMS user Registration and Authentication (AKAv1-MD5)
- IMS Call Initiation
- Voice and video calls
- JMF 2.1.1 - Supported Formats (See 7.2.1.1, Table 7.3)

- SIP Call Transfer (Blind and Consultative Transfer)
- Dial history
- Contact list, IM and Presence support

6.4 Ethereal & SIP Scenario Generator

6.4.1 Network Protocol Analyzer-tool: Ethereal

Ethereal is a free software protocol analyzer, or "packet sniffer" application, used for network troubleshooting, analysis, software and protocol development, and education, being able to display encapsulation and single fields and interpret their meaning . It has all of the standard features of a protocol analyzer. In June 2006 the project was renamed from Ethereal to Wireshark due to trademark issues.

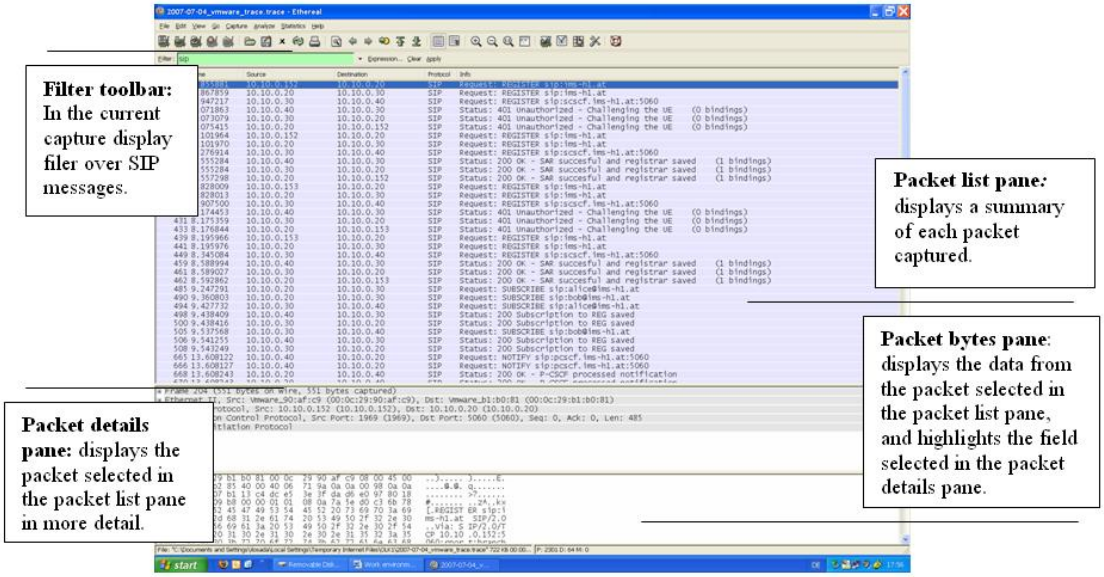


Figure 6.5: Ethereal panes

Wireshark is released under the GNU General Public License, and it uses the cross-platform GTK+ widget toolkit. It runs on Unix and Unix-like systems, including Linux, Solaris, FreeBSD, NetBSD, OpenBSD and Mac OS X (although GTK+ only works with X11 on Mac OS X, so the user will need to run an X server such as X11.app), and on Windows.

Gerald Combs started, in late 1997, to writing Ethereal in order to develop a

tool for capturing and analyzing packets and tracking down networking problems. Ethereal was initially released, after several pauses in development, in July 1998 as version 0.2.0. Not long after, Gilbert Ramirez, Guy Harris and Richard Sharpe started contributing dissectors and contributing patches. The list of people who have contributed to Ethereal has become very long since then, and almost all of them started with a protocol that they needed that Ethereal did not already handle. So they copied an existing dissector and contributed the code back to the team. As of now there are over 500 contributing authors while Gerald continues to maintain the overall code and issues releases of new versions.

In the current work, Ethereal (maybe rather Wireshark) is used like a tool for analyse a network protocol (SIP) over an IMS architecture.

The following are some of the many features this software provides:

- Available for UNIX and Windows.
- Data can be captured "from the wire" from a live network connection or read from a capture file.
- Live data can be read from Ethernet, FDDI, PPP, token ring, IEEE 802.11, Classical IP over ATM, and loopback interfaces (at least on some platforms)
- Display packets with very detailed protocol information. (Displaying can be refined using a display filter which can also be used to selectively highlight and color packet summary information)
- Open and Save packet data captured.
- Import and Export packet data from and to a lot of other capture programs.
- Filter packets on many criteria.
- Search for packets on many criteria.
- Statistics reports.

6.4.2 Creating SIP Call Flows: SIP Scenario Generator

The SIP Scenario Generator creates SIP Call Flows or SIP scenario diagrams, in html format, of SIP messages from ethernet capture files. SIP Scenario Generator shows the actual call processing trace in a format that is easily understood using browser technology. Clicking on a sip message hyperlink displays the contents of

```
⊟ Session Initiation Protocol
  ⊟ Request-Line: INVITE sip:alice@ims-h1.at SIP/2.0
      Method: INVITE
      [Resent Packet: False]
  ⊟ Message Header
      Via: SIP/2.0/TCP 10.10.0.153:5060;rport;branch=z9hG4bK65BADDF8
      CSeq: 3661 INVITE
    ⊞ To: <sip:alice@ims-h1.at>
      Content-Type: application/sdp
    ⊞ From: "bob" <sip:bob@ims-h1.at>;tag=587C380E
      Call-ID: 1304992310@10.10.0.153
      Subject: sip:bob@ims-h1.at
      Content-Length: 303
      User-Agent: KPhoneSI/1.1
      Max-Forwards: 70
    ⊞ Contact: "bob" <sip:bob@10.10.0.153;transport=tcp>
  ⊟ Message body
    ⊟ Session Description Protocol
        Session Description Protocol Version (v): 0
      ⊞ Owner/Creator, Session Id (o): username 0 0 IN IP4 10.10.0.153
        Session Name (s): The Funky Flow
      ⊞ Connection Information (c): IN IP4 10.10.0.153
      ⊞ Time Description, active time (t): 0 0
      ⊞ Media Description, name and address (m): audio 8000 RTP/AVP 0 8 3 97 98 101
      ⊞ Media Attribute (a): rtpmap:0 PCMU/8000
      ⊞ Media Attribute (a): rtpmap:8 PCMA/8000
      ⊞ Media Attribute (a): rtpmap:3 GSM/8000
      ⊞ Media Attribute (a): rtpmap:97 ILBC/8000
      ⊞ Media Attribute (a): rtpmap:98 SPEEX/8000
      ⊞ Media Attribute (a): rtpmap:101 telephone-event/8000
      ⊞ Media Attribute (a): fmtp:97 mode=20
        Media Attribute (a): sendrecv
```

Figure 6.6: Paquet details for a INVITE Request -Ethereal Capture-

the traced SIP message.

The SIP Scenario Generator is a freeware program that will run on most computers. For Windows Platforms, SIP Scenario is an executable program. For other platforms, SIP Scenario is a Perl script and requires the Perl interpreter.

The SIP Scenario Generator program was started as a solution for the need of an inexpensive tool to debug and analyze SIP messages traces. Ethereal doesn't display the information as a scenario diagram so it can be difficult to understand the message flow. By means of HTML formatting and browser technology, SIP Scenario deals with the display problem that appears when the text files managed have a very wide width (depending on what is being traced).

Chapter 7

Getting up and running the testbed

7.1 ftw testbed

By means of OSIMS, ftw's testbed brings an appropriate environment for IMS components testing, just as it becomes a useful tool when research and development purposes are in mind. The OSIMS implementation finally tested has the architecture configuration depicted in Figure 7.1 (for future references), where UE1 (Alice) and UE2(Bob) are IMS-Communicator clients, P/S/I-CSCF are the SER implementation developed in the OSIMS project and HSS is the FHoSS:

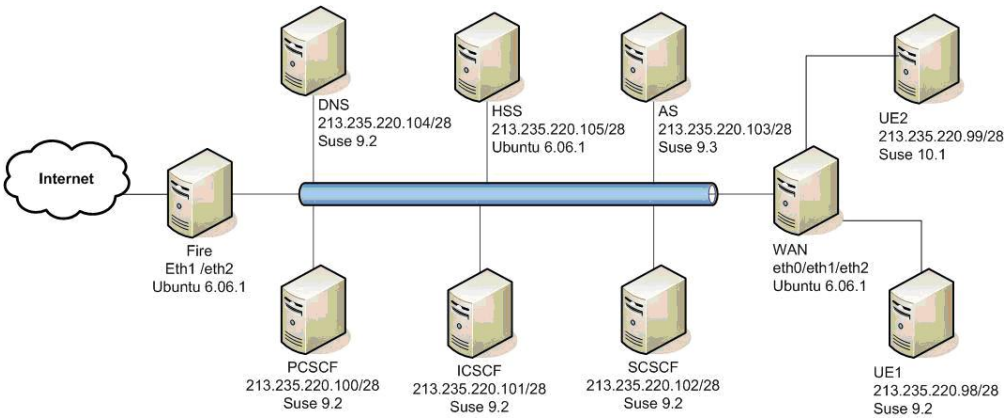


Figure 7.1: FTW testbed configuration

7.2 Client: IMS-Communicator

7.2.1 Configuration

The configuration of the IMS-Communicator is done manually in the xml file (IMS-Communicator.xml), by the menu Settings > Configure at the UA GUI or by running the Setup Wizard. The complete configuration performed via the menu Settings and the resulting xml file are annexed in appendix B.1 and B.2 respectively, in order to allow the complete test repeatability in the same circumstances. It is extremely important restart the IMS-Communicator client after making changes in the configuration file, otherwise, the client will not apply them.

Tables 7.1 and 7.2 gather a more concrete explanation of the configuration properties which are in the spotlight of the present work, whether it be because they are QoS related, are changed during different tests or are key factors for the overall testbed running. The table shows as well how these properties affect the SIP/SDP message.

Property	Description	Value	SIP/SDP
PREFERRED_AUDIO_ENCODING	number of the preferred audio codec, from the audio codecssupported by JMF	26,31,34	m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
PREFERRED_VIDEO_ENCODING	number of the preferred audio codec, from the audio codecssupported by JMF	0,3,4,5,6,8,15,18	m=video 22222 RTP/AVP 34 26 31
NO_AUDIO_DESCRIPTION_IN_SDP	if audio description should or not be included in the SDP	FALSE	(if true: NO "m=audio ..." line)
NO_VIDEO_DESCRIPTION_IN_SDP	if video description should or not be included in the SDP	FALSE	(if true: NO "m=video ..." line)
AUDIO_PORT	port where to receive incoming audio data	default= 22224	m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
VIDEO_PORT	port where to receive incoming video data	default= 22222	m=video 22222 RTP/AVP 34 26 31
PUBLIC_ADDRESS	SIP public user identity	alice@ims3.ftw.at	INVITE sip:alice@ims3.ftw.at
TRANSPORT	transport protocol used by the client.	TCP (UDP by default)	Via: SIP/2.0/TCP 213.235.220.98:5060
PREFERRED_LOCAL_PORT	if none configured or port already in use, other will be generated	5060	Via: SIP/2.0/TCP 213.235.220.98:5060

Table 7.1: IMS-Communicator: Parametres Configuration

Property	Description	Value	SIP/SDP
DISPLAY_NAME	SIP display-name, to be included in the headers From, To and Contact	alice	From: "alice" <sip:alice@ims3.ftw.at>; tag=32820206
PREFERRED_ADDRESS	SIP address to be include in the P-Preferred-Identity header	alice@ims3.ftw.at	P-Preferred-Identity: <sip:alice@ims3.ftw.at>
ACCESS_TYPE	access network type, to be included in the P-Access-Network header	IEEE-802.11	P-Access-Network-Info: IEEE-802.11
DEFAULT_LOCAL_PRECONDITION	precondition mechanism, e2e or sendrecv	sendrecv	a=des:qos mandatory local sendrecv
VIDEO_BANDWIDTH	bandwidth value to be include in the video media description offer/answer (resource reservation)	100	m=video 22222 RTP/AVP 34 26 31 b=AS:100
AUDIO_BANDWIDTH	bandwidth value to be include in the audio media description offer/answer (resource reservation)	25	m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18 b=AS:25
IP_ADDRESS	network interface address used for route SIP messages.	213.235.220.98	Via: SIP/2.0/TCP 213.235.220.98:5060
OUTBOUND_PROXY	SIP outbound proxy address, in the format IPaddress:port/transport	213.235.220.100: 5060/tcp	Route: <sip:213.235.220.100: 5060;transport=tcp>

Table 7.2: IMS-Communicator: Parametres Configuration

7.2.1.1 Property: Preferred audio/video codec

These options affect, just as the table shows, directly the "a=rtpmap" attribute line. As the captured traces will point up, the "a=rtpmap:" maps from an RTP payload type number (as used in an "m=" line) to an encoding name denoting the payload format to be used.

IMS-Communicator allows the codec selection by means of the payload type number associated, the problem here is that usually such assignment is not static, instead, as RFC 4566 [8] specifies, it is more common for that assignment to be done dynamically (actually is this fact which motivates the "a=rtpmap:" line presence).

The user is remitted to RFC 1890 [**?**] for further details, since the number associated to each audio/video codec follows the payload types for standard audio and video codecs gathered in this document. Nevertheless, since the support of video capture in IMS-Communicator is achieved by the Java Media Framework (JMF) API the list of codecs supported are reduced to those which appear in Table 7.3.

Media Type	RTP Payload	JMF 2.1.1 Solaris/Linux Performance Pack
Audio: G.711 (U-law) 8 kHz	0	R,T
Audio: GSM mono	3	R,T
Audio: G.723 mono	4	R,T
Audio: 4-bit mono DVI 8 kHz	5	R,T
Audio: 4-bit mono DVI 11.025 kHz	16	R,T
Audio: 4-bit mono DVI 22.05 kHz	17	R,T
Audio: MPEG Layer I, II	14	R,T
Video: JPEG (420, 422, 444)[1]	26	R,T
Video: H.261	31	R
Video: H.263[2]	34	R,T
Video: MPEG-I[3]	32	R,T

Table 7.3: JMF 2.1.1 - Supported Formats

Due to the fact that H263 has not payload type numbers assigned, it's hardly surprising that the used numbers differ for this present in the flow example of chapter 5, where H263 has 98 as payload number.

7.2.1.2 Transport protocol

During the setup stage, after the configuration previously mentioned and although ensuring the correct use of IP addresses, the proper working system in terms of P/I/S-CSCF, the successful of the registration process and basic performance parameters, a basic problem appears: the clients couldn't establish a call between them. When Alice tried to call Bob (or vice versa), the P-CSCF answered with a 477 error, that the client GUI showed as follows (See also Appendix D.2):

Oddly, the call session between an IMS-Communicator client and a k-phone client (previously configured over our testbed) was performed properly, just as the call between two k-phones.

In order to find the cause that give rise to the problem and attack it properly, the procedure was simply analyze k-Phone call flows and compare them with the initial IMS-Communicator call flows (those performed before the error message appears). Ethereal was also useful in this first stage since the traces obtained revealed that the basic fields remain unchanged except one: k-Phone was running over TCP while IMS-Communicator was default configured for run over UDP. It would be a further work to detect how to solve this problem but by the moment achieve an overall operation is enough for the current objectives of the work, then the default configuration has been changed towards the connection oriented protocol.

7.2.2 Starting the clients

7.2.2.1 Registration

The start of the clients is performed (after the configuration procedures) just running the shell script. At this moment, the IMS-Communicator GUI appears

waiting for the user registration, as can be realised in the "Non Registered" red coloured phrase. When successful registration, there should be visible a green

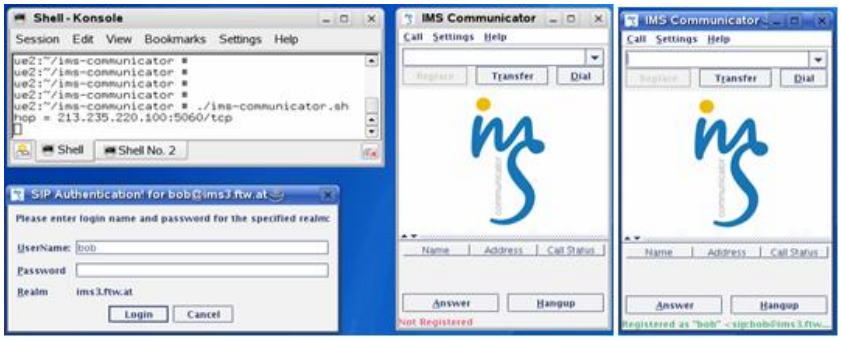


Figure 7.2: IMS-Communicator: Registration

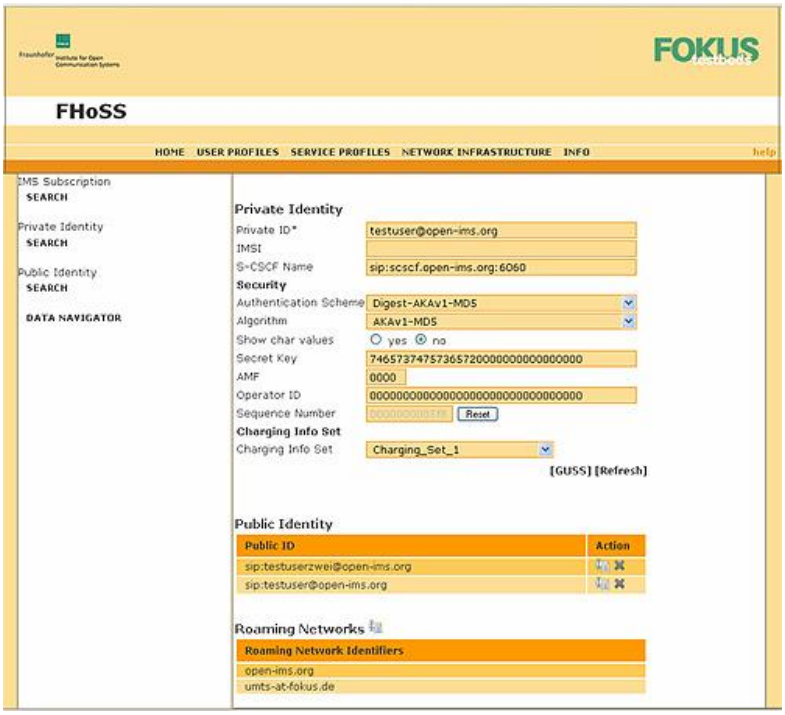


Figure 7.3: FHoSS: User Profile

coloured phrase saying "Registered as "bob" <sip:bob@ims3.ftw.at>" (the same for Alice). The registration process is outside the focus of this project; nevertheless, for the sink of completeness, Appendix D.1 gathers the flows performed during such stage. The flow diagram has been generated during a real registration process by means of Ethereal captures an their treatment with SIP Scenario Generator.

7.2.2.2 Calling Process

From that moment, a UA can place a call with a peer UA. Write the SIP address of the peer UA on the combobox just above the menu and click on the Dial button. Previous called peers are saved in the dialhistory.txt file, and are loaded into the SIP address combo-box.

On pressing the dial button the caller UA will send an INVITE to the peer UA (the called). Both (Alice and Bob in this example) will negotiate media requirements and, if the resources are available for the session (See 8.3) and media negotiation is successful, the called will start to ring. At this moment, the peer UA can refuse or accept the call by pressing the Hangup or the Accept button respectively. Before the peer UA accepts the call, the caller can also cancel the invitation by pressing the Hangup button.

7.2.3 Other utilities

Runtime Information

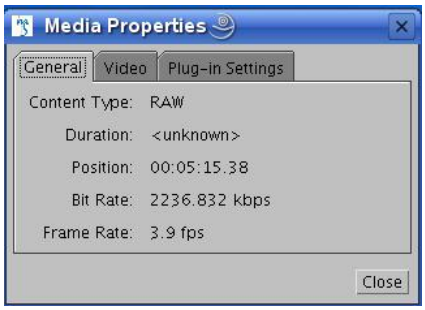


Figure 7.4: IMS-Communicator: Media Properties

Runtime information can be obtained during execution. A button placed in the bottom right side of the image, when pressed, will give us some information about media properties such as frame rate, video format, audio format and the flow graph for the current video media. The last one is perhaps the most interesting and is viewed via the the Plug-in Settings and the PlugIn Viewer. Additionally, information is logged by JMF to the file jmf.log in the directory from where the program was started.

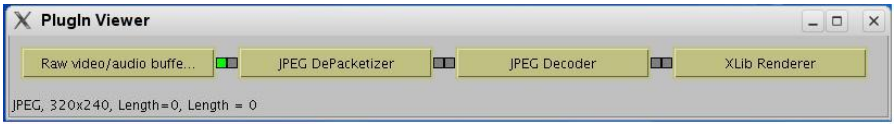


Figure 7.5: IMS-Communicator: Plugin Viewer

Contact-List

The IMS-Communicator supports Contact-list, which enables Instant Messaging and Presence. To show or hide the Contact-list GUI, there's an option available at Settings > Show/Hide Contact-list.

Figure 7.6: IMS-Communicator: Media Properties

There is also the property <net.java.sip.communicator.gui.IM_GUI_MODE> which with the "true" value enables the Contact-list GUI by default on the IMS-Communicator start-up. Contacts and Groups can be added. Those actions are accessible thru the mouse's right button over the entries on the Contact-list GUI.

7.3 Ethereal

The proper capture of traffic traces is basic for obtaining useful and meaningful results, so it is an important task decide where the capture has to be done, since the testbed connection that allows the communication between the implied entities limits the simultaneous capture of all the traffics belonging to a single media session.

The choice here is relatively easy: the P-CSCF is the key entity.

The testbed is designed as a simple local IMS architecture with only one CSCF for each type (Interrogating/Serving/Proxy). If the traffic is monitored at the P-CSCF during a video session call all the flows (except those specific to the Intermediate CN Subsystem entities, that is, between S-CSFC, I-CSCF and HSS) of interest will be captured. Notice that the same P-CSCF is connected simultaneously with Alice, Bob and the S-CSCF .

The diagram used in chapter 5 can be redrawn then as Figure 7.7 shows (the IP addresses are just a remainder that allows the basic figure depicted here be a quick reference point for identify the involved nodes in the ethereal captures and the outputs of the scenario generator tool)

P-CSCF becomes in such configuration the origin/destination entity of all the flows under the interest of the study from the point of view of the SIP signalling flows for QoS provision.

Concerning the network protocol analyzer-tool used, Ethereal, it is important to highlight some basic configuration options. The capture has been performed over the eth0 interface (213.235.220.100) with SIP packets filtering and the "update list of packets in real time" option activated.

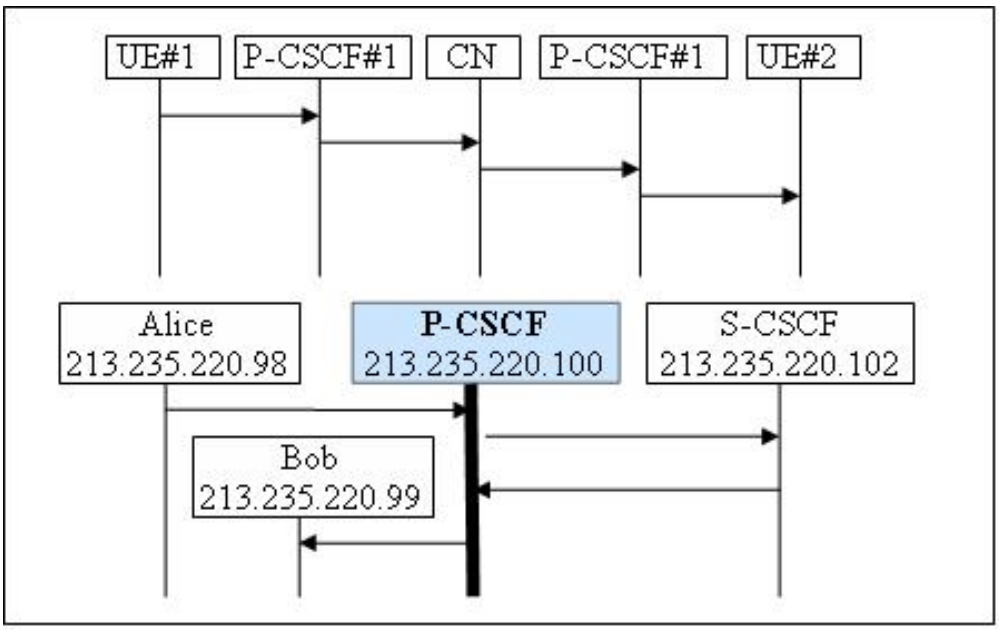


Figure 7.7: Entities involved in the Ethereal captures

Chapter 8

Testbed evaluation

Let's see now if the tandem OSIMS & IMS-communicator allows QoS provision by means of the mechanisms introduced in the previous chapters. Which of them are supported, which not; which of them would be, according to the research developed, strongly recommended for an overall user satisfaction.

In other words, it is an important task, since IMS is now in an advanced development stage, to identify where are the more significant gaps in the last chance within the implantation process of such system: offer high quality services and be this services appreciated as such for the users, objective that requires, from a providers point of view, be able to ensure QoS provision.

This chapter is organized as follows: First of all, a list of the QoS mechanisms that have been observed during the analysis of the captured traces. Next, the missing but desired QoS features just as the estrange behaviour observed over those features that are (in principle) considered in the scenario under study. Finally, some proposal for deal with detected lacks.

8.1 Preconditions Support

The IMS-communicator choice as the suitable client for perform real traffic flows over IMS architecture and analyze them in terms of QoS management was motivated by the fact that such client supports Preconditions mechanism, an important cornerstone for QoS provision.

Preconditions mechanism relies in three basic aspects: an SDP media offer with

an associate QoS desirable (or mandatory) level, the set on hold of the session until the satisfaction of required preconditions and the session initiation based on "Session Progress" and "Update" messages previous to the media exchange.

The traces analysis show how IMS-communicator over OSIMS follows the expected evolution of the lines "current status" and "desired status" during the progress of the session establishment and the pass of the signalling flows across the corresponding IMS entities. Next, an extract of such lines from the complete set of traces of Appendix D.3:

```
SIP MESSAGE 1 213.235.220.98:47858() -> 213.235.220.100:5060()
INVITE sip:bob@ims3.ftw.at SIP/2.0

a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
-----------------------------------------------------------------------
SIP MESSAGE 8        213.235.220.99:35791() -> 213.235.220.100:5060()
SIP/2.0 183 Session progress

a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
-----------------------------------------------------------------------
SIP MESSAGE 20       213.235.220.98:47858() -> 213.235.220.100:5060()
UPDATE sip:bob@213.235.220.99:5060 SIP/2.0

a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
-----------------------------------------------------------------------
SIP MESSAGE 24       213.235.220.99:35791() -> 213.235.220.100:5060()
SIP/2.0 200 OK

a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

In the same way, the SIP headers Supported and Required are affected by the Preconditions mechanisms and their changes are performed as follows:

```
SIP MESSAGE 1        213.235.220.98:47858() -> 213.235.220.100:5060()
INVITE sip:bob@ims3.ftw.at SIP/2.0

Supported: 100rel,precondition,early-session
-----------------------------------------------------------------------
SIP MESSAGE 8        213.235.220.99:35791() -> 213.235.220.100:5060()
SIP/2.0 183 Session progress

Require: 100rel, precondition
-----------------------------------------------------------------------
SIP MESSAGE 20       213.235.220.98:47858() -> 213.235.220.100:5060()
UPDATE sip:bob@213.235.220.99:5060 SIP/2.0

Absence of supported/require lines
```

It is also important notice the presence in IMS-communicator.xml file the line that configures the type of precondition mechanism to be applied "e2e" or "sendrecv".

```
<DEFAULT_LOCAL_PRECONDITION value="sendrecv" />
```

8.2 Codec Negotiation Procedures

In spite of the possibility of select a preferred codec that IMS-communicator offers, there is not a overall session codec negotiation, since each communication way (send/receive) is affected by the choice of its destination part, that is, the preferences apply only in media reception but not in transmission. Such behaviour is logic since the clients in both sides are the same, that is, even they express different preferences, they support the same codecs, so they can satisfy (on sending) the peer part requirements.

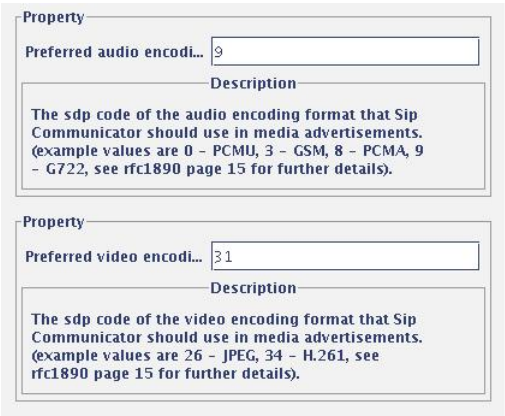


Figure 8.1: Preferred codecs selection (Settings > Configure)

Is true also, that maybe the preferences are motivated for the access network performance, available resources or other limitations that not only imply terminal capabilities and then the codec selected is also relevant on sending way (video and audio codecs affect the bandwith used just as other QoS technical parameters such as the packet loss ratio or efficient error concealment)

In the previous chapter, devote to the basis of the configuration procedures, the supported audio and video codecs supported by the terminals are listed from the Java Media Framework (JMF) API specification. Nevertheless and in spite of the offers that the traces show (see Figure 8.2), the analysis of sessions with different codec preferences over different access networks (considering that maybe this factor affects final codec used) give rise to summarize de codec negotiation procedures as follows:

1. The preference option affects the "m=" line in terms of the payload type number order. Then if we select as preferred video codec 26 (jpeg) the resulting video codecs list will be 26-34-31.
2. The absolute preference is listed 26-34-31 (if not preference is selected by the user).
3. The codec used depends on the destination part (which plays the media stream).
4. The h261 choice is always ignored and jpeg used instead.
5. It becomes really important to restart the clients after any preference change, otherwise the client behaviour is not consistent (maybe affected by previous sessions memory)

```
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

Figure 8.2: Media and Codec information on SDP Body

8.3 P-Access-Network-Info and Bandwidth Negotiation

Just as the previous chapter indicates, IMS-Communicator allows the possibility to choice the Access Type used among those contemplated in 3GPP TS 24.229 [35]; nevertheless, such property seems have not effect on the video sessions performed, excepting on the appropriate SIP lines during the session negotiation flows. Then, it is predictable that the Core IMS implemented in OSIMS doesn't deal, until now, such information for provide an efficient level of service. Concerning the values for the required audio

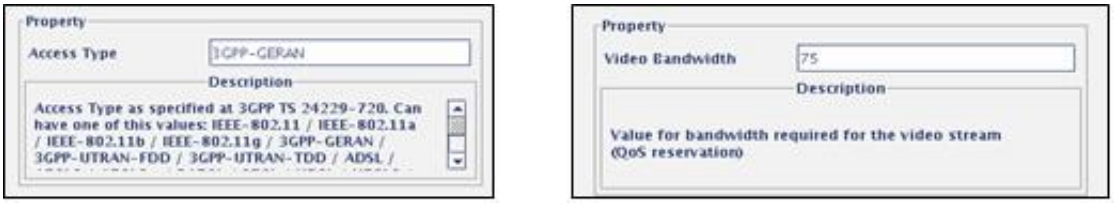


Figure 8.3: Access Network and Video Bandwidth selection (Settings > Configure)

and video bandwidths for the streams involved in the session and being them indispensables for the QoS reservation procedures, Portugal Telecom Inovacao (developer of IMS-Communicator) informs the user that the current IMS-Communicator version has several limitations, including among them, one especially interest in this aspect; when placing a call, resources for the media session are considered always as available. One more time, chances on this properties don't reflect different behaviours and the traffic analysis can be done only under the perfect conditions mentioned: required resources always available, nothing further from reality.

In any case, the traffic flows follow the theoretical SIP/SDP exchange detailed in chapter 5, that is to say, in spite of the always ensured resources, the UAs exchange Session Progress and Update messages as if they has to deal with real IP-CAN bearer reservation for each media session, emulating the need to wait for the preconditions achievement, even when, by definition, the QoS preconditions will be always meet.

8.4 What is missing?

The terminal informs the SIP methods allowed by means of the header created witch such aim in mind:

Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE

It is obvious that several mechanisms that SIP provides in order to allow media sessions be established in a QoS ensured manner are not contemplated in the IMS-Communicator

client, such as the OPTIONS SIP method that allows the UA to query another UA or proxy server as to its capabilities or the NEGOTIATE SIP method that allows meta-session parameters negotiation previous to the session establishment.

But is not only the client which has limitations; OSIMS framework, even being a extremely useful solution for fill the void in the Open Source Software landscape with the aim to enable the development and research of IMS services, concept and architecture, has an important lack when the QoS provision is in the spotlight: the Media Authorization mechanisms that performs a Policy Based QoS Control Scenario in IMS (See chapter 3).

PDF functionality is currently not part of the P-CSCF (neither a standalone unit) that is, the policy-based QoS solution adopted by the 3GPP is not available over our testbed. It is then a pending challenge ensure that sufficient QoS resources are provided to authorized users, ensure that previous to the session start the UE negotiates the media flows for this IMS session with the peering UE according to the service subscription.

SIP based signalling in the application-layer signalling plane is already capable to deal with such purpose but the lack of a QoS Policy Based Service Architecture implemented in OSIMS has been a key factor limiting the scope of the study.

Last but not lest, is missing the UA capabilities extension that allows them to provide more information about themselves when they register defining mechanisms by which a SIP UA convey its capabilities and characteristics (as parameters of the Contact header field) to other UAs and to the registrar for its domain. And in the same way, is missing the caller preferences extension that allows callers to indicate the type of UA they want to reach. This information is conveyed as parameters of the Accept-Contact, Reject-Contact and Request-Disposition headers, by means of which a SIP request can incorporate and code preferences among service capabilities.

Remembering the first chapters one has to realize that this extension is closely related with, more than the QoS concept, with the QoE notion, basically because it fills the basis of subjective perception: preferences. Nowadays, are been developed several studies with the goal to find the best manner to manage and deal with caller preferences for achieve the best service provision.

8.5 Proposal:P-CSCF dealing with the precondition mechanism

Finally, here a modification proposal for the P-CSCF Source Code that allow this entity to be able to extract AS bandwidth information from the SDP message in the SIP body and take the descision to continue with the session initiation signalling or refuse the

request by answering with a a ”580 - Precondition Failure”

- To be added within **/openimscore/ser_ims/trunk/modules/pcscf/sdp_util.c**

```
/*Extract AS bw information, that is, the bw requested by the UA*/
static int request_bw(struct sip_msg* msg)
{
str body, bwline, bwparse;
int m_req_bw, total_req_bw;
char *cp;
char *v1p, *v2p, *m1p, *m2p, *b1p, *bodylimit;

total_req_bw = 0;

if (extract_body(msg, &body) == -1) {
LOG(L_ERR, "ERROR: request_bw: can’t extract body from the message\n");
return -1;
}

bodylimit = body.s + body.len;


/* find the first "v=" line that identifies a session*/
v2p = find_sdp_line(body.s, bodylimit, ’v’);
if (v1p == NULL) {
LOG(L_ERR, "ERROR: request_bw: no sessions in SDP\n");
return -1;
}
for(;;) {
/* Per-session iteration. */
v1p = v2p;
if (v1p == NULL || v1p >= bodylimit)
break; /* No sessions left */
v2p = find_next_sdp_line(v1p, bodylimit, ’v’, bodylimit);
/* v2p is text limit for session parsing. */
m2p = find_sdp_line(v1p, v2p, ’m’);
/* Have this session media description? */
if (m1p == NULL) {
LOG(L_ERR, "ERROR: request_bw: no m= in session\n");
return -1;
}
/* Iterate media descriptions in session */
```

```
for (;;) {
m1p = m2p;
if (m1p == NULL || m1p >= v2p)
break;
m2p = find_next_sdp_line(m1p, v2p, 'm', v2p);
b1p = find_sdp_line(m1p, m2p, 'b');

/* Extract bandwidth info */

bwline.s = b1p;
cp = eat_token_end (bwline.s, 12);
If (cp == bwline.s) {
LOG(L_ERR, "ERROR: request_bw: no bw in ‘b=’\n");
return -1;
}
/* Extract the nummeric value from de b=AS: line */
for (i=5; i< ( cp - bwline.s ); i++){
bwparse[i-5]= bwline[i];
}

/* Convert the bw info to integer and increment */
/* the counter of the total request bw */
m_req_bw = atoi (bwparse);
total_req_bw + = m_req_bw;
}
}

return (total_req_bw);
}
```

- To be added within **the Proxy-CSCF configuration file**

```
# main routing logic

route{

    route(Sanity_Checks);

    force_rport();
```

```
if (method=="REGISTER") {
    route(REGISTER);
    break;
}

if (method=="NOTIFY"&&uri==myself){
    route(NOTIFY);
    break;
}

if (!P_mobile_terminating()){

    # Request Initiated by the UE


    if (P_is_in_dialog("orig")){
        route(Orig_Subsequent);
        break;
    }

    if (P_is_in_dialog("term")){
        route(Term_Subsequent);
        break;
    }

    # No dialog yet
    if (method=="ACK"){
        t_relay();
        break;
    }else
    if (method=="INVITE" || method=="SUBSCRIBE"){
        route(Orig_Initial);
        break;
    }else{
        if (method==UPDATE){
            sl_send_reply("403","Forbidden -
            Target refresh outside dialog not allowed");
            break;
        }
        if (method=="BYE" || method=="PRACK"){
            sl_send_reply("403","Forbidden -
            Originating subsequent requests outside dialog not allowed");
            break;
```

```
                }
                route(Orig_Standalone);
                break;
            }

        }else{

            # TODO - check if this does come from an UE and that UE is
             unregistered

            # Request Terminated by the UE

            if (!P_is_in_dialog("term") &&
                (method=="INVITE" || method=="SUBSCRIBE")){
                route(Term_Initial);
                break;
            } else {
                if (P_is_in_dialog("term")){
                    route(Term_Subsequent);
                    break;
                }else{
                    if (method==UPDATE){
                        sl_send_reply("403","Forbidden -
                        Target refresh outside dialog not allowed");
                        break;
                    }
                    if (method=="BYE" || method=="ACK" || method=="PRACK"){
                    sl_send_reply("403","Forbidden - Terminating subsequent
                    requests outside dialog not allowed");
                        break;
                    }
                    route(Term_Standalone);
                    break;
                }
            }
            break;
        }

    }
    #*********************************************************************
    # Complete code in https://www.openimscore.org/docs/ser_ims/PCSCF.html
    # Here only the ORIGINATING route (corresponding to the INVITE method
    #*********************************************************************
```

```
route[Orig_Initial]
{
    log(1,">>       Orig_Initial\n");
    if (!P_is_registered()){
            sl_send_reply("403","Forbidden - Not Registered! You must
            register first with a S-CSCF");
            break;
    };
    if (!P_assert_identity()){
            sl_send_reply("403","Forbidden - You must register first
            with a S-CSCF");
            break;
    };

#*********************************************************************
# Deal with request and available bandwidth
#*********************************************************************

if (request_bw() > avail_bw){
# The value of the bandwidth that is available is determined by  the
# P-CSCF by means of specific mechanisms. If there is not bw enough the
# P-CSCF answers with a 580 - Precondition Failure warning message

       sl_send_reply("580","Precondition Failure");
            break;
    };

#*********************************************************************
#*********************************************************************

    # add IBCF/THIG route here if required
    loose_route();
    if (!P_follows_service_routes()){
        #Variant 1 - deny access to the network
        #sl_send_reply("400","Bad Request - Not following indicated
        Service-Routes");
        #break;
        #Variant 2 - enforce routes and let the dialog continue
        P_enforce_service_routes();
    }

    }
```

The resulting flow can be depisted as follows:

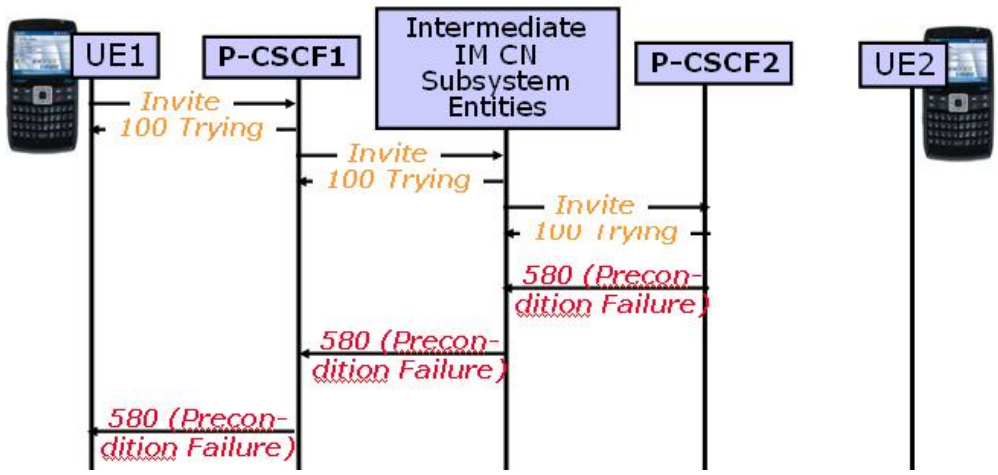


Figure 8.4: Preconditions failure

Chapter 9

Conclusions

9.1 Achievements and Results

In recent years a considerable amount of research within the field of QoS has been performed or is ongoing since supporting reliable real-time services is a decisive aspect in packet-switched based networks. Providing QoS for multimedia streaming services implies harmonized interworking between protocols and mechanisms specified by IETF and 3GPP, among other standardisation bodies.

The first seconds of a streaming session have been in the spotlight of the work, due to the importance to sustain an initially negotiated QoS level, avoiding the session establishment in case that QoS can not be provided at all. This document has reviewed the main aspects in the field of QoS and the protocols used for coordinate its provision over IMS. With this goal, the cornerstones have been the two threads that have divided the work:

- An exhaustive research that gathers a complete State of Art in IMS Session Initiation Procedures and Mechanisms for QoS Provision . An overview of the QoS architecture, Policy-Based Control Scenario and QoS requirements; outlines about SIP/SDP in the signalling plane for QoS provision; in brief, all the information considered as a basic background, with 3GPP specifications as main contribution.

- Evaluation of the implementation level of such features that has been achieved in a real IMS Open Source environment (FOKUS OSIMS), as basic model of the real IMS implantation. The conclusion is that in spite of the existence of consistent proposals, studies and current researches in the IMS QoS field (in fact, one of the most important contributions of the IMS), in spite of the fact that SIP/SDP signalling is extended enough for deal with the new IMS requirements (especially QoS), that is, in spite of have identified the problem and proposed its solution, the available and tested Open Source IMS Core Network implementation (OSIMS) is far from a complete QoS support.

As a main contributions of the present work can be hightlighted, then:

- State-of-Art: The first chapters can be understood as a complet IMS QoS State of Art, usefull as a reference point for future works with this field in the spotlight.
- SIP/SDP in detail: Chapter five becomes a key point of the work since it gives the reader a detailed analysis of SIP/SDP traffic flows in a Preconditions scenario. After an arduos research task, all the exchanged messages have been explained in a comprehensible manner, understandig their origin, their function and, above all, their changes across the IMS network. It allows the developper identify where (entities) or when (session porgress) improvements are necessary or can be included.
- Testbed: Next chapters bring the complete setup for running an IMS testbed and relevant results between the traffic that (following 3GPP specifications) wolud be espected and the traffic finaly obtained.
- Detection of lacks for complete and real QoS provision.
- Improvement proposal for the model scenario studied (the testbed). The proposl allows deal with preconditions from the Core Network, freeing up the terminal for extra operations, making the establishment process faster and giving more responsibility to the P-CSCF, a key entity in the QoS architecture.

9.2 Possible improvements

Also, the developed research allows the detection of possible improvements or basis for further works. Considering the first stage of the investigation, the improvements proposed can be summarized as follows:

- Addressing future design proposals towards less terminal participation in the first negotiation stage (extending the use of User Profiles for manage useful information from the Core Network)
- Flexible QoS management in order to meet different user requirements and be able to dynamically change.
- Dynamic negotiation, that is, not restricted only at the initial stream setup.
- Renegotiation of the media stream properties in handover scenarios or changing client device during a media stream session.

Concerning now the features of OSIMS as development and research open tool it would be extremely recommended:

- Implement Policy-Based QoS Architecture, a PDF that allows QoS Provision Solutions be tested properly and succesfully.
- Implement IMS clients capable of dealing whith SIP QoS signalling allowing an appropiate client-Core Network interworking

Part III

Appendix

Appendix A

Summary: SIP & Session Negotiation

A.1 Preconditions (RFC 3312)

Preconditions (RFC 3312 [23]) allow a session establishment conditioned to a specific achievement of a set of requirements, for example when a reservation-based QoS is used. Since in the packet switched domain the resources are not preallocated before its usage is needed, it becomes indispensable ensure that the remote party will not decline the session establishment and know the bandwidth and codec that will be supported, before to start the reservation procedures. This extension, defined in RFC 3312 [23] allows user agents to express preconditions by means of a SIP option tag precondition (signalling) and new SDP body attributes (to describe the parameters of the session). Note that the QoS preconditions are included in the SDP description rather than in the SIP header because preconditions are stream specific.

A.2 Methods

A.2.1 Options (RFC 3261)

All SIP UAs must support the OPTIONS method (RFC 3261 [5]) that allows them to query another UA or proxy server as to its capabilities. By means of this method, a client is able to discover information about the supported methods, content types, extensions, codecs, etc. without "ringing" the other party [5]. So, an OPTIONS request is sent as part of an established dialog to query the peer on capabilities that may be utilized later in the dialog. It can also be used to determine the basic state of a UAS, which can be an indication of the INVITE request acceptation, since a 200 (OK) will be returned if the *User Agent Server* (UAS) is ready to accept a call and a 486 (Busy Here) would be returned if it is busy.

A.2.2 Negotiatie (Internet Draft)

NEGOTIATE [52] allows negotiation (prior to the session establishment) of meta-session parameters, settings and algorithms when setting up sessions using SIP. Among others, it includes compression algorithms, code book size, message integrity mechanisms, and encryption algorithms that have to be negotiated in a generic manner between any to SIP entities. The NEGOTIATE follows the next offer-answer model: First, the *User Agent Client* (UAC) makes an offer on one or more session parameters, negotiate carries key. Next, UAS accepts (200 OK) or rejects (488 or 606) offer and key, which is used to maintain negotiation state. Finally, the session establishment (INVITE request) takes place with negotiated parameters and key. A message body is required for this method in order to carry meaningful meta-session information. Negotiations expire and must be refreshed; terminating a Negotiation is achieved by sending a NEGOTIATE with the "Expires" header set to "0".

A.3 Indicating User Agent Capabilities in the Session Initiation Protocol (SIP): SIP Capabilities (RFC 3840) & Caller Preferences for SIP (RFC 3841)

The UA capabilities extension (RFC 3840 [53]) allows user agents to provide more information about themselves when they register defining mechanisms by which a SIP UA can convey its capabilities and characteristics (as parameters of the Contact header field) to other user agents and to the registrar for its domain. The caller preferences extension (RFC 3841 [54]) allows callers to indicate the type of user agent they want to reach. This information is conveyed as parameters of the Accept-Contact, Reject-Contact and Request-Disposition headers.

By means of the Contact and Accept-Contact headers the features of one registered device at a callee's proxy server and caller preferences that are matched against those capabilities are defined. Accept-Contact headers are used to select the best matching communication endpoint, described by its capabilities stored as Contacts in network proxy servers. During session negotiation SIP Accept-Contacts headers are matched against available SIP session contacts to determine the best caller-callee match for the session.

A caller can add one or more 'Accept-Contact' header fields to his request, each of them containing a set of feature parameters that define a feature set. How a SIP request can incorporate and code preferences among service capabilities can be understood with the examples that tables A.1 and A.2 provide (this example is extracted from [55], which includes a depth explanation about how to manage and order such preferences).

Accept-Contact	q-val
*;type=video/mpeg,video/h261; description=<high resolution>; language=fr	1.0
*;type=video/quicktime; description=<high resolution>; language=fr	1.0
*;type=video/mpeg; description=<low resolution>; language=de	0.8
*;type=video/mpeg; description=<low resolution>; language=jp	0.7
*;type=video/mpeg; description=<high resolution>; language=fr	1.0
*;type=video/h261; description=<high resolution>; language=fr	1.0
*;type=video/h261; description=<low resolution>; language=fr	0.8

Table A.1: SIP Accept-Contact headers

Contact
sip:u1@h.example.com; type=video/mpeg; description=<low resolution>; language=de
sip:u2@h.example.com; type=video/quicktime; description=<high resolution>; language=fr
sip:u2@h.example.com; type=video/h261; description=<low resolution>; language=jp

Table A.2: SIP Contact headers

SIP also offers the use of so-called implicit preferences. Implicit preferences only occur if no explicit preferences in an Accept-Contract have been given. In such a case, some typical assumptions or stereotypes can help to lead to better quality in service provisioning. For example an appropriate preference would be high resolution content over low resolutions, if a user did not specify something explicitly. Thus, with respect to the available bandwidth (e.g. the constraints added by network nodes) a user would be generally better served, if -in the case that there is a choice between high and low resolution content- always the high resolution content would be delivered. However, if a user would have given an explicit preference on low resolution content (maybe due to limited capabilities of his/her client device) this preference would have to be respected and no other implicit preference would have been added that could possibly overwrite

the user preference.

A.4 SIP Headers

A.4.1 P-Access-Network-Info (RFC 3455)

This header field is an extension created by the IETF as a consequence of 3GPP requirements and useful in any SIP-based network that also provides layer 2/layer 3 connectivity through different access technologies. SIP UAs may use this header to convey information about the access technology to proxies that are providing services, which make use of this information to optimize services for the UA. The SIP server that provides the user with services may desire knowledge about the access network: information about the type of access network that the UA is currently using, crude location, identity of the cell the user is being served by, etc. Such functionality is achieved by defining this new private SIP extension header, a P-Access-Network-Info header that carries information relating to the access network between the UAC and its serving proxy in the home network.

The header field provides two types of access information:

- The type of layer 2/3 technology used by the IMS terminal. Such an information allows AS to customize the service depending on the characteristics of the access network and determine the available bandwidth to the terminal.
- The Identity of the radio cell where the IMS terminal is connected. The cell ID implicitly contains some rough location information that may be used to provide a personalized service to the user.

A.4.2 P-Media Authorization (RFC 3313)

This extension (RFC 3313 [30]) allows QoS provision for media streams established via the SIP. The mechanism to authorize the establishment of media streams is based on the network inserting a media authorization token that IMS terminals return to the network when requesting the establishment of a media stream. The authorization token that a SIP UA needs to present to the network in order to obtain QoS is obtained via SIP from the QoS enabled proxy by means of an extension SIP header. The mechanism in 3GPP that makes use of the described SIP extension is known as the SBLP specified in 3GPP TS 29.208 [37], which follows the media authorization model described in RFC 3521 [38].

In order to support a media authorization scheme a new P-Media-Authorization general header field is defined. This header field, a private SIP extension, contains one or more media authorization tokens which are to be included in subsequent resource reservations for the media flows associated with the session.

A.4.3 Security Agreement (RFC 3329)

This IMS SIP extension specifies how to negotiate security capabilities for multiple types of endpoints. The evolution of security mechanisms often introduces new algorithms, or uncovers problems in existing ones, making negotiation of mechanisms a necessity. The purpose of this specification is to define negotiation functionality for the SIP between a UA and its first-hop SIP entity. Three new SIP header fields, namely Security-Client, Security-Server and Security-Verify, are defined.

A client wishing to use the security agreement adds a Security-Client header field to a request addressed to its first-hop proxy. This header field contains a list of all the security mechanisms that the client supports. A Require and Proxy-Require header fields are also added by the client with the value "sec-agree" to its request. The contents of the Security-Client header field may be used by the server to include any necessary information in its response. The server adds a Security-Server header field to this responselisting the security mechanisms that the server supports (independent on the contents of the client's list). Afterwards, it compares the list received in the Security-Client header field with the list to be sent in the Security-Server header field. When the client receives this response, it will choose the common security mechanism with the highest "q" value (The "q" value indicates a relative preference for the particular mechanism). Then, that particular security mechanism is initiated. All the subsequent SIP requests sent by the client to that server make use of the security mechanism initiated in this first step.

A.5 SDP Simple Capability Declaration (RFC 3407)

The SDP was not intended to provide capability negotiation, however, as the need for this has become increasingly important, work has begun on a "next generation SDP" (SDPng) that supports both session description and capability negotiation. Since SDPng is not anticipated to be backwards compatible with SDP and SIP and *Media Gateway Control Protocol* (MGCP) use SDP and are likely to continue doing so in the future, this document defines a set of SDP attributes that enables the provision of a minimal and backwards compatible capability declaration mechanism.

The SDP Simple Capability Declaration (simcap) is defined by a set of SDP attributes. Together, these attributes form a capability set which describes the complete media capabilities of the endpoint. Any previous capability sets issued by the endpoint for the session in question no longer apply. An endpoint receiving a capability set from another endpoint may use any of the media formats included in that capability set in a later attempt to negotiate media streams with the other endpoint.

The capability set consists of a sequence number and one or more capability descriptions

listing all media formats the endpoint is currently able and willing to support and may include one or more capability parameters to further define the capability. A session description must not contain more than one capability set, however the capability set can describe capabilities at both the session and media level.

A.6 BW Negotiation

The SDP shall include bandwidth information for each media stream and also for the session in total. The bandwidth information for each media stream and for the session is defined by the *Application Specific* (AS) bandwidth modifier as defined in RFC 4566 [8]:

b=<bwtype>:<bandwidth>

The <bwtype> is an alphanumeric modifier giving the meaning of the <bandwidth> figure. Two values are defined in this specification:

- CT : Proposed upper limit to the bandwidth used (the "conference total" bandwidth). The primary purpose of this is to give an approximate idea as to whether two or more sessions can coexist simultaneously. When using the CT modifier with RTP the conference total refers to total bandwidth of all RTP sessions.
- AS: The bandwidth is interpreted to be application specific (it will be the application's concept of maximum bandwidth). Normally, this will coincide with what is set on the application's "maximum bandwidth" control if applicable. For RTP-based applications, AS gives the RTP "session bandwidth".

CT gives a total bandwidth figure for all the media at all sites while AS gives a bandwidth figure for a single media at a single site, although there may be many sites sending simultaneously.

A.7 Warning Codes (RFC 3261)

Warning codes (RFC 3261 [8]) provide information supplemental to the status code in SIP response messages when the failure of the transaction results from a SDP problem. The first digit of warning codes beginning with "3" indicates warnings specific to SIP. Among them, Warnings 330 through 339 are warnings related to basic network services requested in the session escription and 370 through 379 are warnings related to quantitative QoS parameters requested in the session description, for example: 370 Insufficient bandwidth: The bandwidth specified in the session description or defined by the media exceeds that known to be available.

Preconditions

SDP parameters

current-status	"a=curr:"	precondition-type						SP status-type			SP direction-tag			
desired-status	"a=des:"	precondition-type		SP strength-tag				SP status-type			SP direction-tag			
confirm-status	"a=conf:"	precondition-type						SP status-type			SP direction-tag			
Values		"qos"	token	"mandatory"	"optional"	"none"	"failure"	"unknown"	"e2e"	"local"	"remote"	"none"	"send"	"recv"	"sendrecv"

Example

```
m=audio 20000 RTP/AVP 0
a=curr:qos e2e send
a=des:qos optional e2e send
a=des:qos mandatory e2e recv
m=audio 20002 RTP/AVP 0
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos optional local sendrecv
a=des:qos mandatory remote sendrecv
```

Rejection

- **New SIP status code:** Server-Error = "580" ;Precondition Failure
- In the offer/answer model, when an answerer wishes to reject a media stream, it sets its port to zero
- Unknown Precondition Type

SIP header fields associated:

Option tag "precondition" for use in:

- Require (offer contains one or more "mandatory" strength-tags)
- Supported (all the strength-tags in the description are "optional" or "none"

OPTIONS Request

- A **Contact** header field MAY be present in an OPTIONS.
- An **Accept** header field SHOULD be included -> type of message body the UAC wishes to receive in the response.
- All UAs MUST **support** the OPTIONS method.

Possible Answers: (the same that would have been chosen had the request been an INVITE)

- 1xx: Provisional: 180 (ringing).
- 2xx: Success: 200 (OK(; 202 (ACCEPTED)
- 3xx: Redirection.
- 4xx: Client Error: Request contains bad syntax or cannot be fulfilled at this server: 401 (Unauthorized), the client needs to provide credentials; 486 (Busy Here)
- 5xx: Server Error: (504 Timeout, if MTLS has not been configured between the home servers).
- 6xx: Global Failure: Request cannot be fulfilled at any server (new class defined for SIP)

200 (OK) Response -> would be returned if the UAS is ready to accept a call

- If the response to an OPTIONS is generated by a proxy server, the proxy returns a 200 (OK), listing the capabilities of the server. The response does not contain a message body.
- Allow, Accept, Accept-Encoding, Accept-Language, and Supported header fields SHOULD be present.
- If it is generated by a proxy, the Allow header field SHOULD be omitted (a proxy is method agnostic)
- May be present: Contact and Warning header fields. A message body MAY be sent

```
OPTIONS sip:carol@chicago.com SIP/2.0
Via: SIP/2.0/UDP
pc33.atlanta.com;branch=z9hG4bKhjhs8ass877
Max-Forwards: 70
To: <sip:carol@chicago.com>
From: Alice
<sip:alice@atlanta.com>;tag=1928301774
Call-ID: a84b4c76e66710
CSeq: 63104 OPTIONS
Contact: <sip:alice@pc33.atlanta.com>
Accept: application/sdp
      Content-Length: 0
```

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP
pc33.atlanta.com;branch=[…];received=192.0.2.4
To: <sip:carol@chicago.com>;tag=93810874
From: Alice <sip:…>;tag=…
Call-ID: a84b4c76e66710
CSeq: 63104 OPTIONS
Contact: <sip:carol@chicago.com>
Contact: <mailto:carol@chicago.com>
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Accept: application/sdp
Accept-Encoding: gzip
Accept-Language: en
Supported: foo
Content-Type: application/sdp
Content-Length: 274
(SDP not shown)
```

Negotiate Offer (Expired 2002)

(1) The NEGOTIATE/200 OK payload is in **XML**.
(2) Multiple NEGOTIATE payloads are permissible in a single NEGOTIATE, they are enclosed using **Multipart MIME**.
(3) Body contents are not changed/manipulated by intermediate proxies.
(4) Use of **S-MIME** for added security is permitted.

New header field:

"Key"|"y"	(":" key-format)			("=" key-param)
	"Header"	"Key32"	"Key8"	*(alphanum)
	key in clear text	32 bit Key	8 bit Key	MD5 hash of values pertaining to both the parties of a NEGOTIATE message: TO URL, FROM URL, Call-ID value and Via Branch (0 if it no present)

Example Header Field: Key:Header=ab45c6d2811e4681

Example key Generation:

- NEGOTIATE sip:broker@example.com SIP/2.0
  Via: SIP/2.0/UDP client.example.com:5060
  From: Endpoint <sip:user@client.example.com>;tag=88a7s
  To: sip:broker@example.com
  Call-ID: 3248543@client.example.com
  CSeq: 1 NEGOTIATE
  Content-Type: application/xpidf+xml
  Content-Length: 120
  ...
- Starter String fed in MD5:

  SIP:Broker@example.comENDPOINT<sip:user@client.exampl.com>;TAG=88A7S3248543@client.example.com0

- MD5 Key output (representation): **ab45c6d2811e4681f23513**

Answer: The offer (placed in a NEGOTIATE payload) may be:
- Accepted: a 200 OK.
- Rejected: 488 or 606 is returned.
- If the key value already exists-> rejected (with 400 "Bad Request" or 409 "Conflict").
- The use of a Key after the expiry of the time limit will result in a 4xx response (488 recommended)

Indicating UA Capabilities

- **Contact:** sip:u1@h.example.com;audio;video;methods="INVITE,BYE";q=0.2 [feature parameters after URI]
- The REGISTER request MAY contain a Require header field with the value "pref"

Caller Preferences

- **Header Field Definitions**
  - **Accept-Contact**
  - **Reject-Contact**
  - **Request-Disposition**
- The first step in proxy processing is to extract **explicit preferences**. To do that, it looks for the Accept-Contact and Reject-Contact header fields. **Features parameters**:
  - "audio"
  - "automata"
  - "class"
  - "duplex"
  - "data"
  - "control"
  - "mobility"
  - "description"
  - "events"
  - "priority"
  - "methods"
  - "extensions"
  - "schemes"
  - "application"
  - "video"
  - "language"
  - "type"
  - "isfocus"
  - "actor"
  - "text"
  - header fields parameters whose name begins with a plus (+)
- If the proxy did not find any explicit preferences in the request it extracts implicit preferences:
  - Methods: it is an implicit preference to have the request routed only to UAs that support the request method
  - Event header field: desire for the request to be routed only to a server that supports the given event package
- Example:

INVITE Header fields / Contact Header Field (dif URI)	Reject-Contact: *;actor="msg-taker";video	Accept-Contact: *;audio; require	Accept-Contact: *;video; explicit	Accept-Contact: *;methods="BYE"; class="business" ;q=1.0	**Q value**	Qa (2n order factor	Preference order
sip:u1@h.example.com;audio;video ;methods="INVITE,BYE";q=0.2		1	1	0,5	**0,2**	0,83	2
sip:u2@h.example.com;audio="FALSE"; methods="INVITE";actor="msg-taker";q=0.2		NOT MACH Discarted	----------	----------	**-----**	----	
sip:u3@h.example.com;audio;actor ="msg-taker"; methods="INVITE";video;q=0.3	Rejected	----------	----------	----------	**-----**	----	
sip:u4@h.example.com;audio; methods="INVITE,OPTIONS";q=0.2		1	NOT EXPLICIT 0	NOT MACH	**0,2**	0,5	3
sip:u5@h.example.com;q=0.5					**0,5**	1	1

P-Access-Network-Info

"P-Access-Network-Info" HCOLON	access-net-spec									
	access-type							*(SEMI access-info)		
Note: P-Access-Network-Info header SHOULD NOT be sent in any initial unauthenticated and unprotected requests (e.g., REGISTER)	"IEEE-802.11a"	"IEEE-802.11b"	"3GPP-GERAN"	"3GPP-UTRAN-FDD"	"3GPP-UTRAN-TDD"	"3GPP-CDMA2000"	token	"cgi-3gpp" EQUAL (token/quoted-string) Cell Global Identity obtained from lower layers of the UE. MCC, MNC, LAC and CI.	"utran-cell-id-3gpp" EQUAL (token/quoted-string) concatenation of the MCC, MNC, LAC and the UMTS Cell Identity, obtained from lower layers of the UE	extension-access-info
			GPRS access							

Example: P-Access-Network-Info: 3GPP-UTRAN-TDD; utran-cell-id-3gpp=234151D0FCE11

P-Media-Authorization

"**P-Media-Authorization**" HCOLON P-Media-Authorization-Token*(COMMA P-Media-Authorization-Token)
Where -> P-Media-Authorization-Token = 1*HEXDIG

- The P-Media-Authorization-Token is included for each dialog in all unreliable provisional responses (except 100), the first reliable 1xx or 2xx response, and all retransmissions of that reliable response for the dialog sent by the QoS enabled SIP proxy to the UAC.
- The User Agent Server receives the P-Media-Authorization-Token in an INVITE (or other) message from the QoS enabled SIP proxy.
- The P-Media-Authorization-Token contains, in binary format, binding information that identifies the SIP dialog that the token belongs to and the address of the PDF that generated the token

Example: P-Media-Authorization: 0020000100100101706466312e686f6d65312e6e6574000c02013331533134363231

Security Agreement

3 new header fields: (Syntax of ipsec-3gpp)

▪ **"Security-Client" HCOLON**	Sec-mechanism *(COMMA sec-mechanism)	
▪ **"Security-Server" HCOLON**	sec-mechanism *(COMMA sec-mechanism)	
▪ **"Security-Verify" HCOLON**	Sec-mechanism *(COMMA sec-mechanism)	
The communicating SIP entities need to know beforehand which keys to use.	**mechanism-name**	*(SEMI mech-parameters)
	ipsec-3gpp	**Mandatory Algorithm "alg"** EQUAL" (hmac-md5-96" / "hmac-sha-1-96")
		Protocol "prot" EQUAL ("ah" / "esp")
		mode "mod" EQUAL ("trans" / "tun")
		encrypt-algorithm "ealg" EQUAL ("des-ede3-cbc" / "null")
		Spi "spi" EQUAL (10DIGIT; 0 to 4294967295)
		port1 "port1" EQUAL (1*DIGIT)
		port2 "port2" EQUAL (1*DIGIT)

Example:

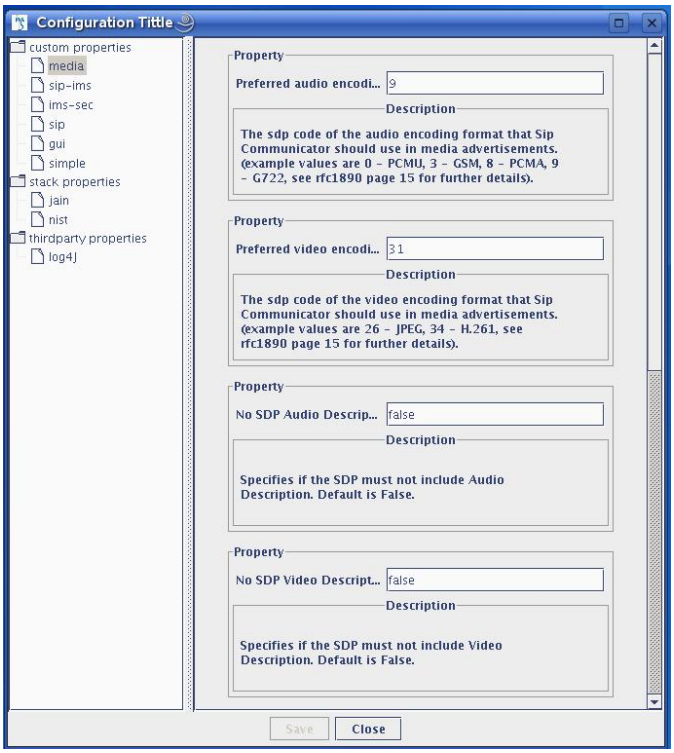
Security-Server: ipsec-3gpp; q=0.1; alg=hmac-sha-1-96;spi-c=98765432;spi-s=87654321;port-c=8642;port-s=7531

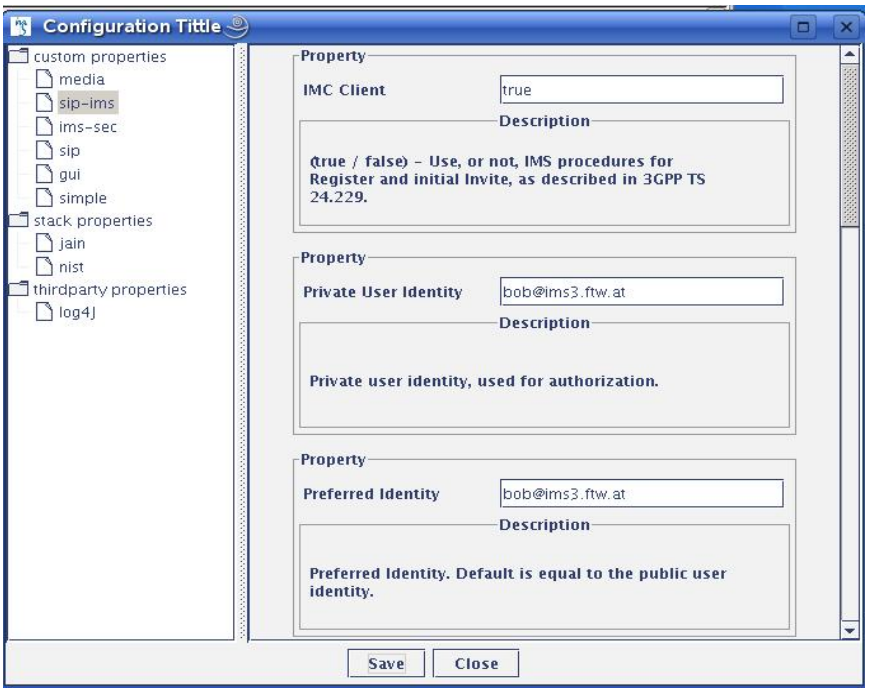
Require/Proxy- Require header fields:
Require: sec-agree
Proxy-Require: sec-agree

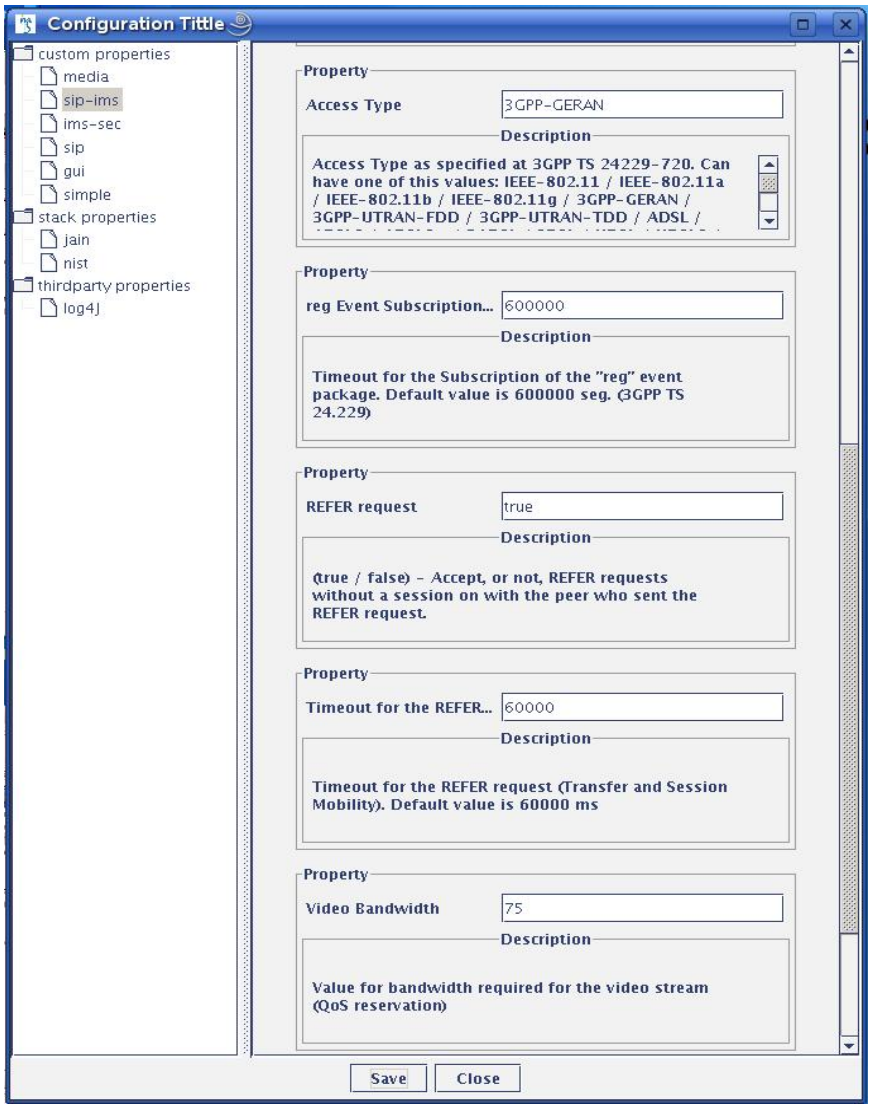
Appendix B

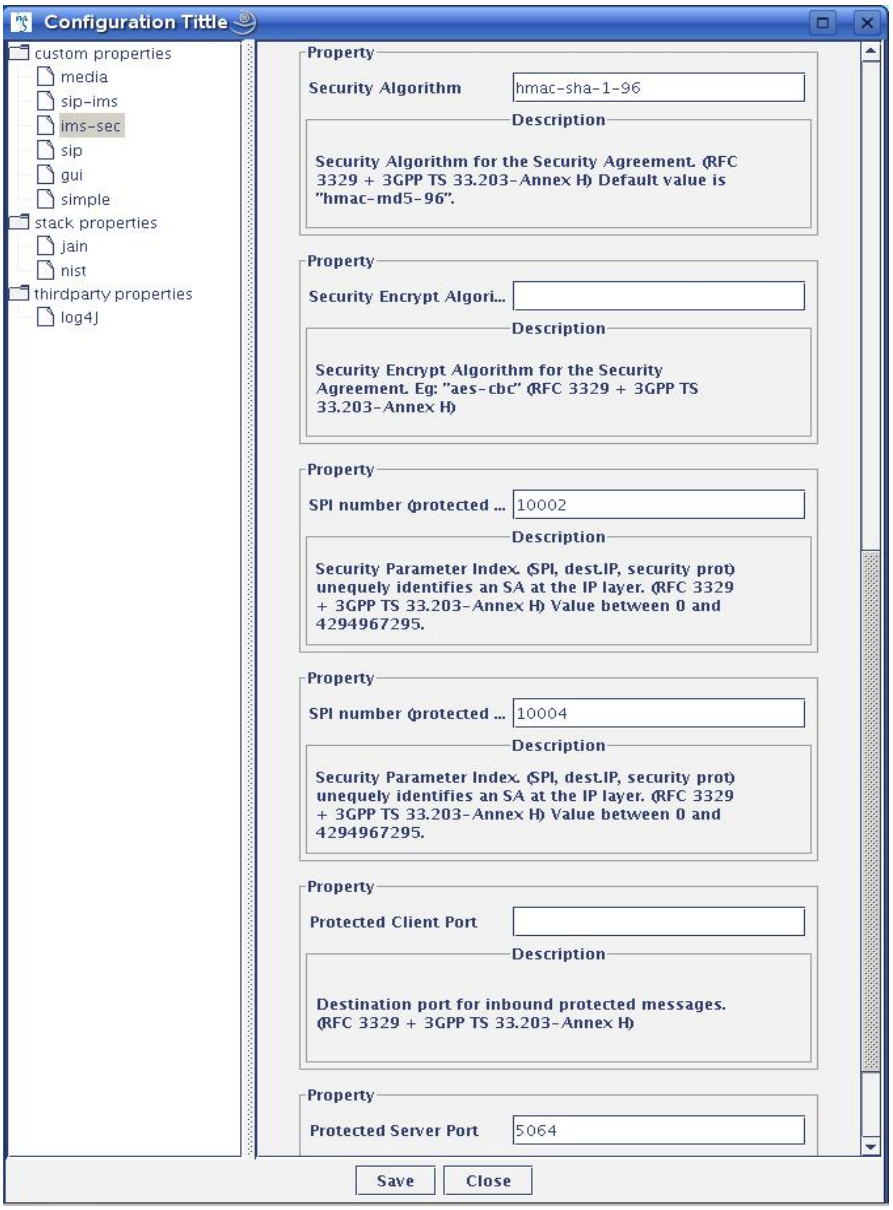
IMS-Communicator Configuration

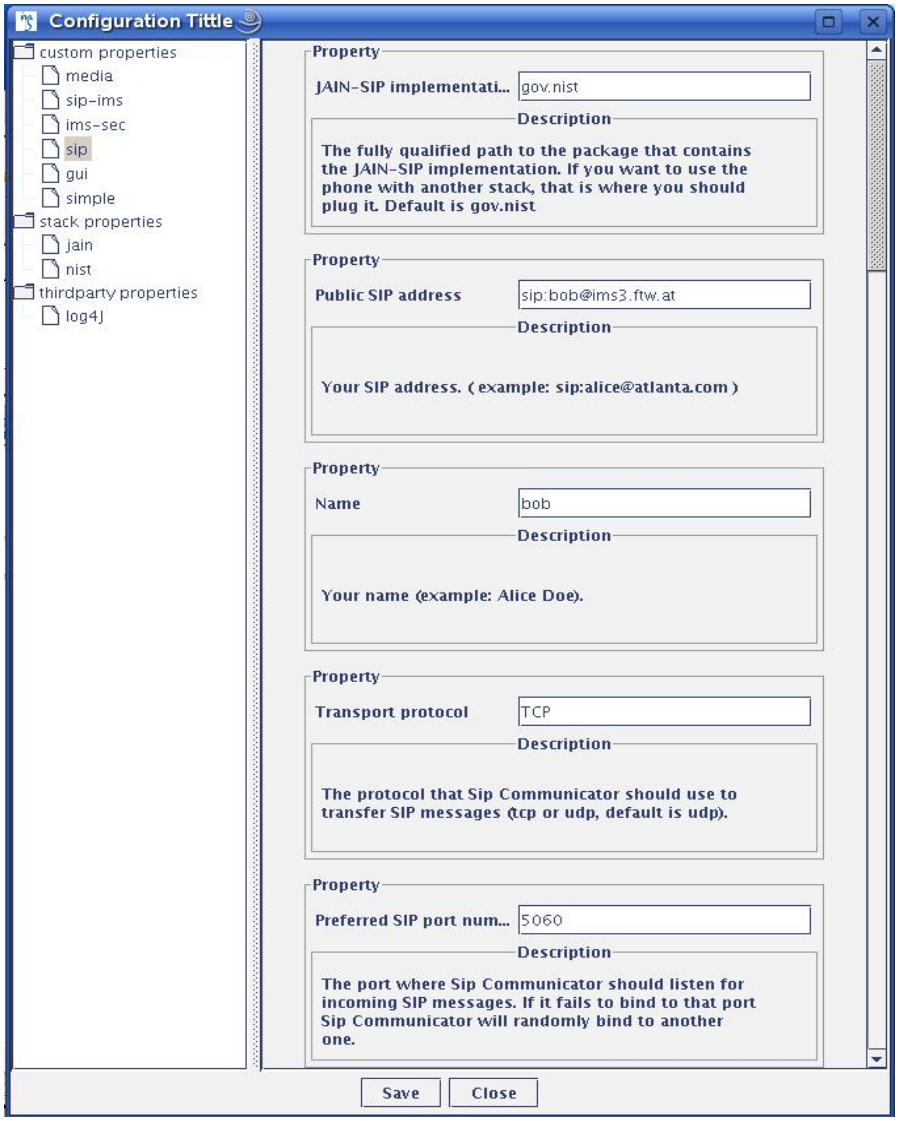
B.1 Settings > Configure

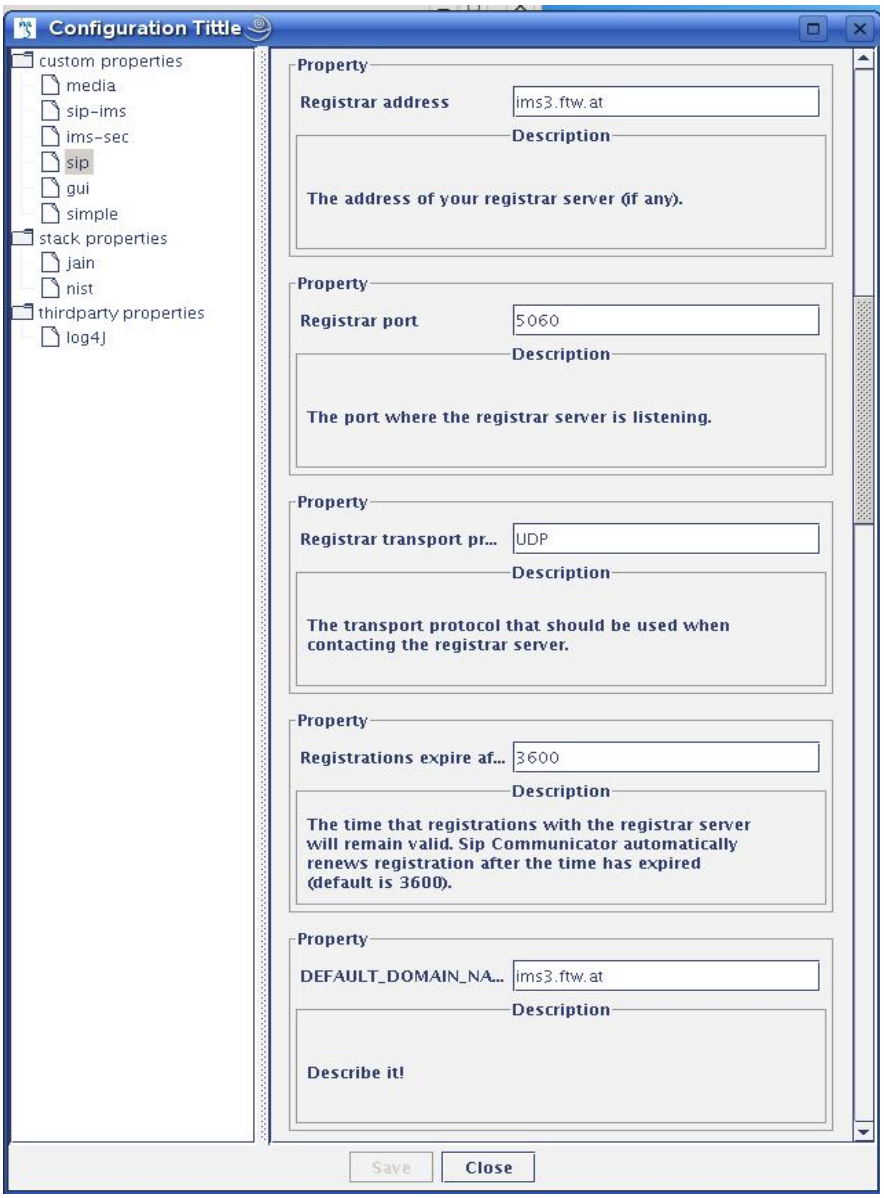


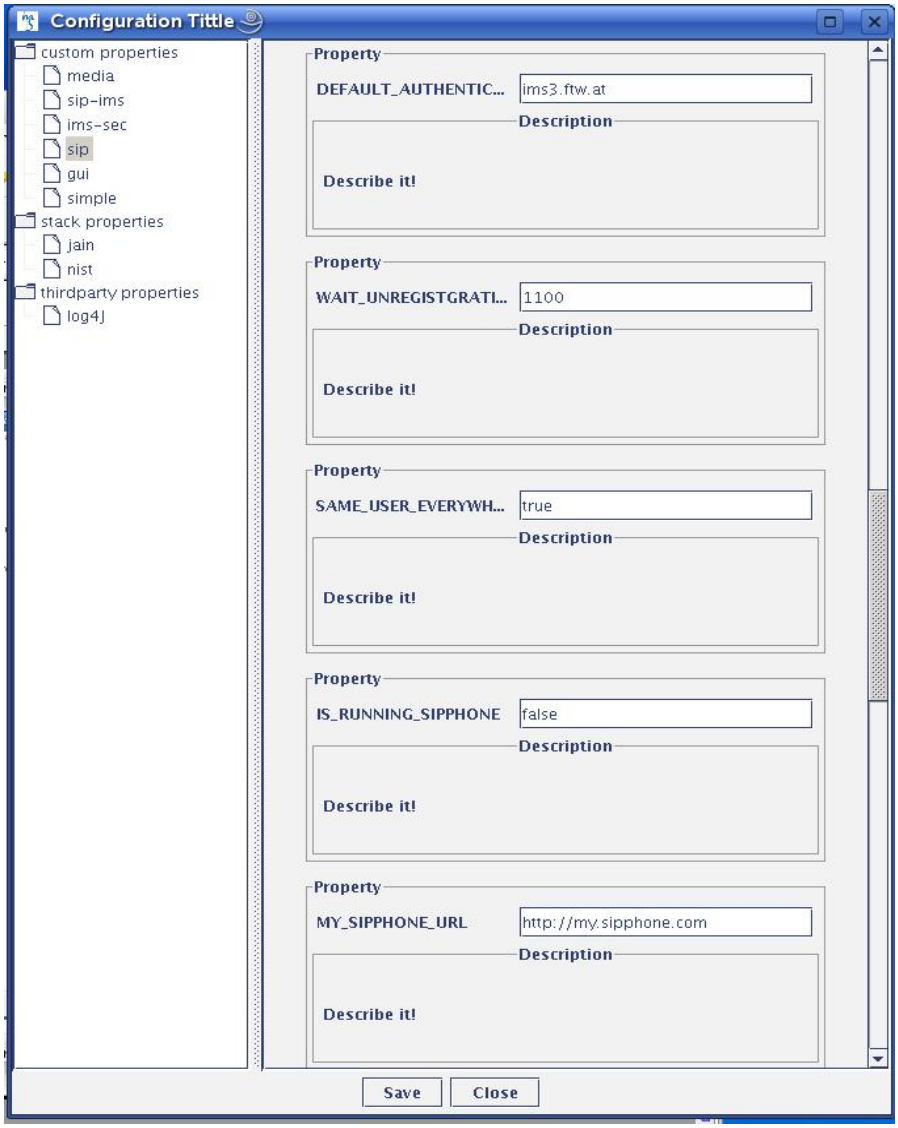

Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
IMC Client
true
Description
(true / false) - Use, or not, IMS procedures for Register and initial Invite, as described in 3GPP TS 24.229.
Property
Private User Identity
bob@ims3.ftw.at
Description
Private user identity, used for authorization.
Property
Preferred Identity
bob@ims3.ftw.at
Description
Preferred Identity. Default is equal to the public user identity.
Save
Close

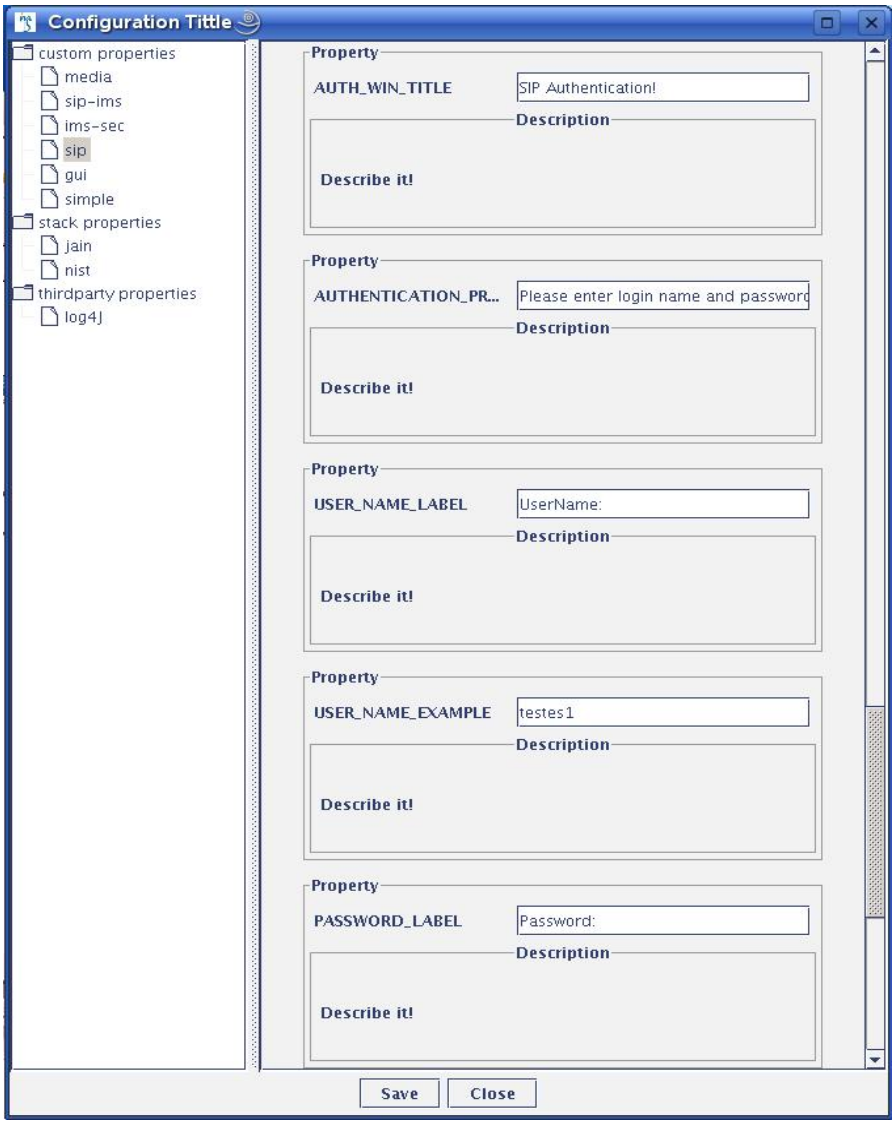

Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
Access Type
3GPP-GERAN
Description
Access Type as specified at 3GPP TS 24229-720. Can have one of this values: IEEE-802.11 / IEEE-802.11a / IEEE-802.11b / IEEE-802.11g / 3GPP-GERAN / 3GPP-UTRAN-FDD / 3GPP-UTRAN-TDD / ADSL /
Property
reg Event Subscription...
600000
Description
Timeout for the Subscription of the "reg" event package. Default value is 600000 seg. (3GPP TS 24.229)
Property
REFER request
true
Description
(true / false) - Accept, or not, REFER requests without a session on with the peer who sent the REFER request.
Property
Timeout for the REFER...
60000
Description
Timeout for the REFER request (Transfer and Session Mobility). Default value is 60000 ms
Property
Video Bandwidth
75
Description
Value for bandwidth required for the video stream (QoS reservation)
Save
Close

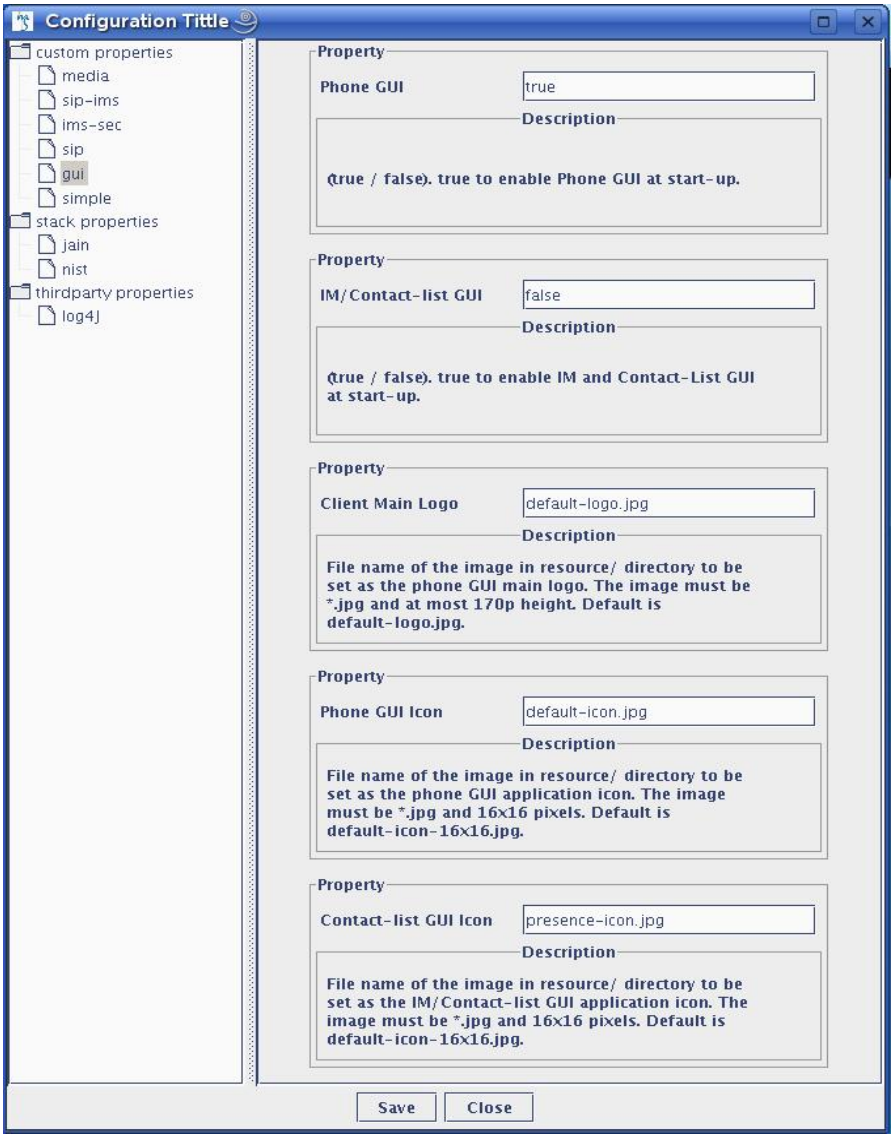

Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
Security Algorithm
hmac-sha-1-96
Description
Security Algorithm for the Security Agreement. (RFC 3329 + 3GPP TS 33.203-Annex H) Default value is "hmac-md5-96".
Property
Security Encrypt Algori...
Description
Security Encrypt Algorithm for the Security Agreement. Eg: "aes-cbc" (RFC 3329 + 3GPP TS 33.203-Annex H)
Property
SPI number (protected ...
10002
Description
Security Parameter Index. (SPI, dest.IP, security prot) unequely identifies an SA at the IP layer. (RFC 3329 + 3GPP TS 33.203-Annex H) Value between 0 and 4294967295.
Property
SPI number (protected ...
10004
Description
Security Parameter Index. (SPI, dest.IP, security prot) unequely identifies an SA at the IP layer. (RFC 3329 + 3GPP TS 33.203-Annex H) Value between 0 and 4294967295.
Property
Protected Client Port
Description
Destination port for inbound protected messages. (RFC 3329 + 3GPP TS 33.203-Annex H)
Property
Protected Server Port
5064
Save
Close

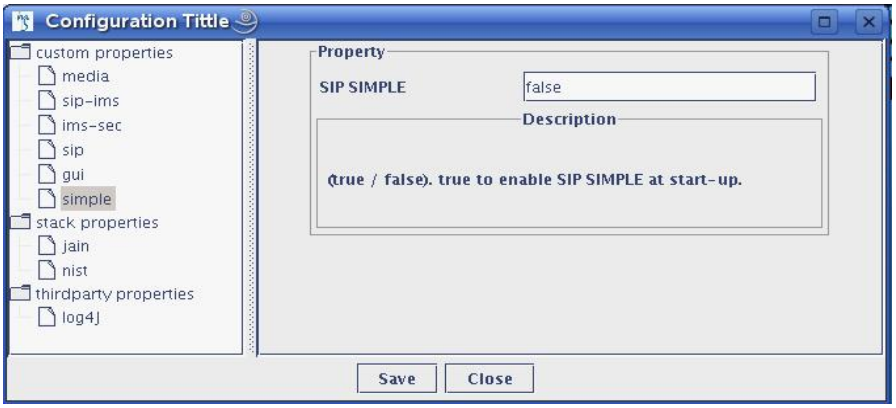

Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
JAIN-SIP implementati...
gov.nist
Description
The fully qualified path to the package that contains the JAIN-SIP implementation. If you want to use the phone with another stack, that is where you should plug it. Default is gov.nist
Property
Public SIP address
sip:bob@ims3.ftw.at
Description
Your SIP address. (example: sip:alice@atlanta.com)
Property
Name
bob
Description
Your name (example: Alice Doe).
Property
Transport protocol
TCP
Description
The protocol that Sip Communicator should use to transfer SIP messages (tcp or udp, default is udp).
Property
Preferred SIP port num...
5060
Description
The port where Sip Communicator should listen for incoming SIP messages. If it fails to bind to that port Sip Communicator will randomly bind to another one.
Save
Close

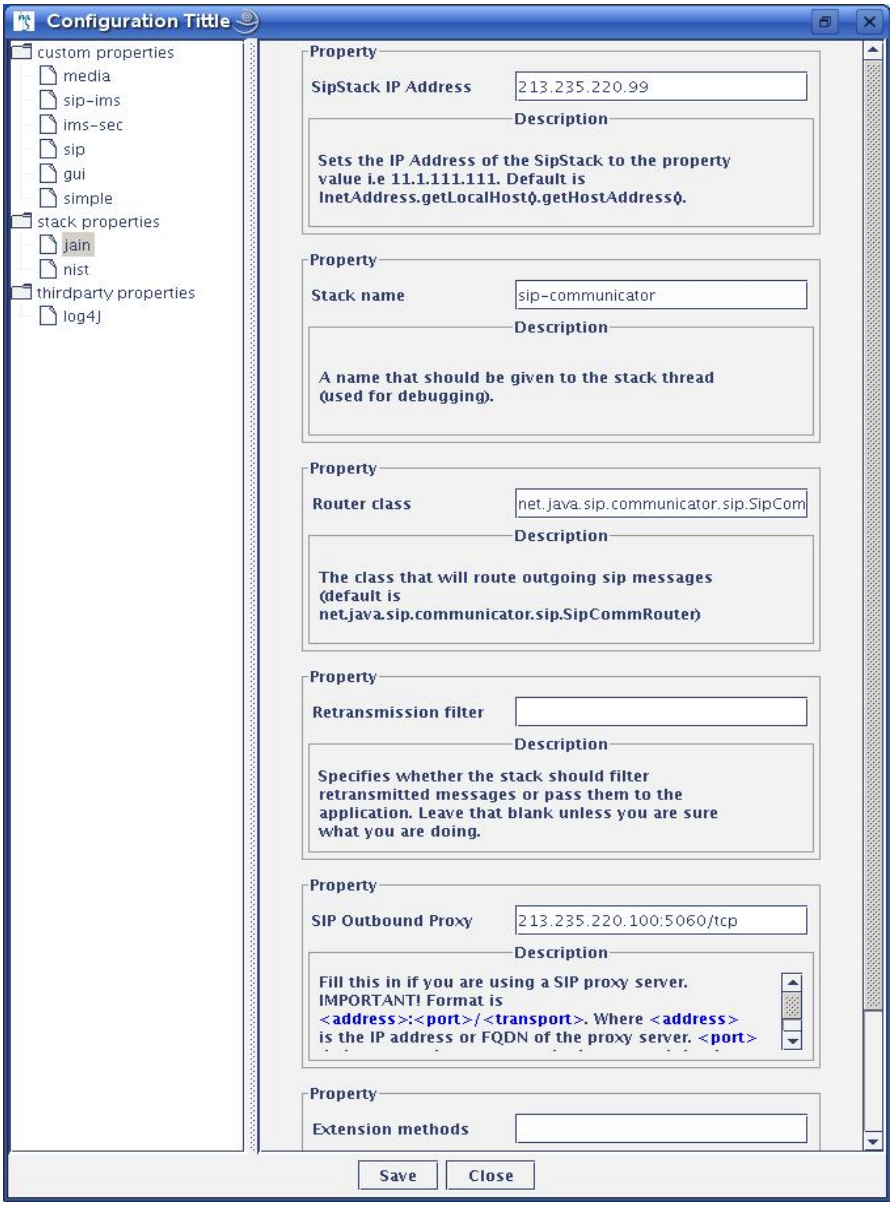

Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
Registrar address
ims3.ftw.at
Description
The address of your registrar server (if any).
Property
Registrar port
5060
Description
The port where the registrar server is listening.
Property
Registrar transport pr...
UDP
Description
The transport protocol that should be used when contacting the registrar server.
Property
Registrations expire af...
3600
Description
The time that registrations with the registrar server will remain valid. Sip Communicator automatically renews registration after the time has expired (default is 3600).
Property
DEFAULT_DOMAIN_NA...
ims3.ftw.at
Description
Describe it!
Save
Close

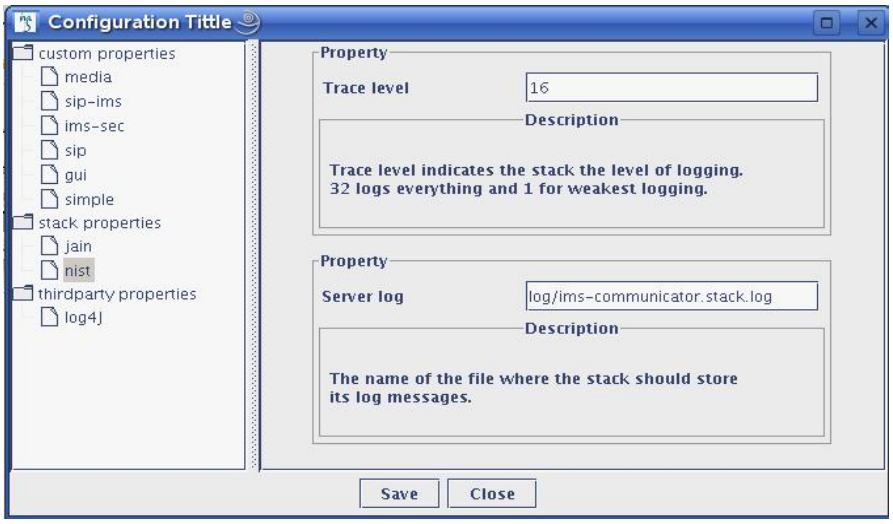

Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
DEFAULT_AUTHENTIC...
ims3.ftw.at
Description
Describe it!
Property
WAIT_UNREGISTGRATI...
1100
Description
Describe it!
Property
SAME_USER_EVERYWH...
true
Description
Describe it!
Property
IS_RUNNING_SIPPHONE
false
Description
Describe it!
Property
MY_SIPPHONE_URL
http://my.sipphone.com
Description
Describe it!
Save
Close

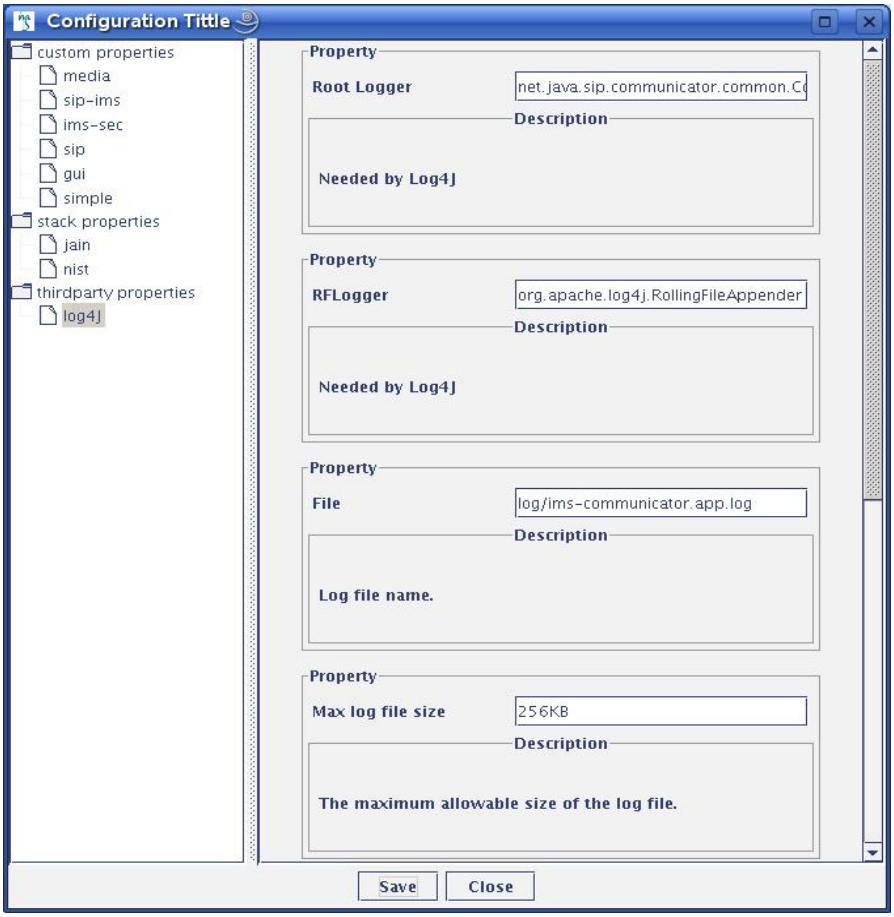

Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
AUTH_WIN_TITLE
SIP Authentication!
Description
Describe it!
Property
AUTHENTICATION_PR...
Please enter login name and password
Description
Describe it!
Property
USER_NAME_LABEL
UserName:
Description
Describe it!
Property
USER_NAME_EXAMPLE
testes1
Description
Describe it!
Property
PASSWORD_LABEL
Password:
Description
Describe it!
Save
Close

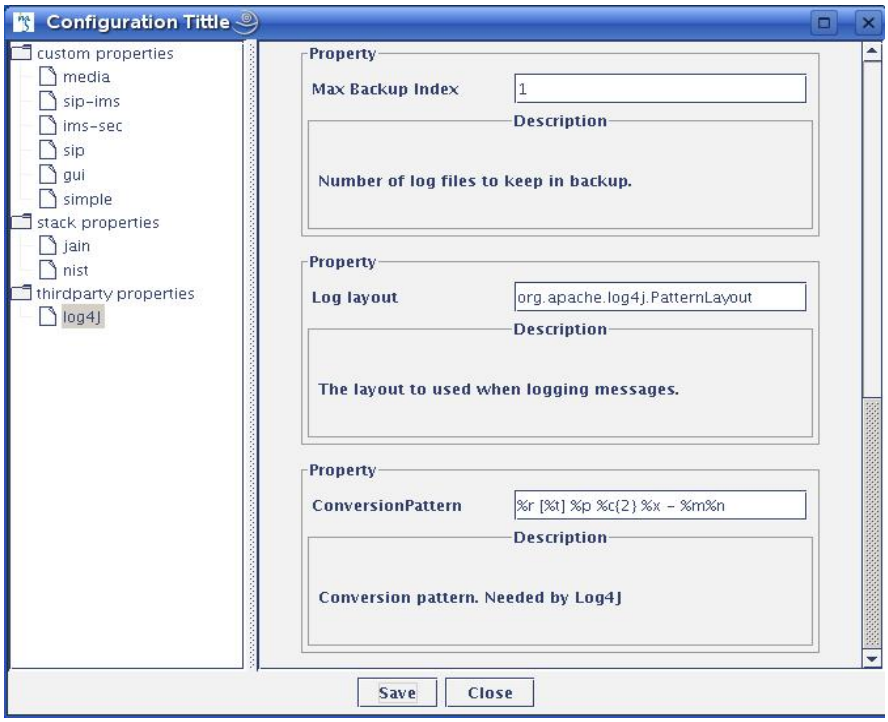

Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
Phone GUI
true
Description
(true / false). true to enable Phone GUI at start-up.
Property
IM/Contact-list GUI
false
Description
(true / false). true to enable IM and Contact-List GUI at start-up.
Property
Client Main Logo
default-logo.jpg
Description
File name of the image in resource/ directory to be set as the phone GUI main logo. The image must be *.jpg and at most 170p height. Default is default-logo.jpg.
Property
Phone GUI Icon
default-icon.jpg
Description
File name of the image in resource/ directory to be set as the phone GUI application icon. The image must be *.jpg and 16x16 pixels. Default is default-icon-16x16.jpg.
Property
Contact-list GUI Icon
presence-icon.jpg
Description
File name of the image in resource/ directory to be set as the IM/Contact-list GUI application icon. The image must be *.jpg and 16x16 pixels. Default is default-icon-16x16.jpg.
Save
Close


Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
SIP SIMPLE
false
Description
(true / false). true to enable SIP SIMPLE at start-up.
Save
Close


Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
SipStack IP Address
213.235.220.99
Description
Sets the IP Address of the SipStack to the property value i.e 11.1.111.111. Default is InetAddress.getLocalHost().getHostAddress().
Property
Stack name
sip-communicator
Description
A name that should be given to the stack thread (used for debugging).
Property
Router class
net.java.sip.communicator.sip.SipCom
Description
The class that will route outgoing sip messages (default is net.java.sip.communicator.sip.SipCommRouter)
Property
Retransmission filter
Description
Specifies whether the stack should filter retransmitted messages or pass them to the application. Leave that blank unless you are sure what you are doing.
Property
SIP Outbound Proxy
213.235.220.100:5060/tcp
Description
Fill this in if you are using a SIP proxy server. IMPORTANT! Format is <address>:<port>/<transport>. Where <address> is the IP address or FQDN of the proxy server. <port>
Property
Extension methods
Save
Close


Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
Trace level
16
Description
Trace level indicates the stack the level of logging. 32 logs everything and 1 for weakest logging.
Property
Server log
log/ims-communicator.stack.log
Description
The name of the file where the stack should store its log messages.
Save
Close


Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
Root Logger
net.java.sip.communicator.common.C
Description
Needed by Log4J
Property
RFLogger
org.apache.log4j.RollingFileAppender
Description
Needed by Log4J
Property
File
log/ims-communicator.app.log
Description
Log file name.
Property
Max log file size
256KB
Description
The maximum allowable size of the log file.
Save
Close


Configuration Tittle
custom properties
media
sip-ims
ims-sec
sip
gui
simple
stack properties
jain
nist
thirdparty properties
log4J
Property
Max Backup Index
1
Description
Number of log files to keep in backup.
Property
Log layout
org.apache.log4j.PatternLayout
Description
The layout to used when logging messages.
Property
ConversionPattern
%r [%t] %p %c{2} %x - %m%n
Description
Conversion pattern. Needed by Log4J
Save
Close

B.2 XML File

```
<?xml version="1.0" encoding="UTF-8" ?>
- <configuration>
- <log4j>
  <rootLogger value="net.java.sip.communicator.common.Console.TraceLevel,

   RFLogger" />
- <appender>
- <RFLogger value="org.apache.log4j.RollingFileAppender">
- <layout value="org.apache.log4j.PatternLayout">
  <ConversionPattern value="%r [%t] %p %c{2} %x - %m%n" />
  </layout>
  <MaxBackupIndex value="1" />
  <File value="log/ims-communicator.app.log" />
  <MaxFileSize value="256KB" />
  </RFLogger>
  </appender>
  </log4j>
- <net>
- <java>
- <sip>
- <communicator>
  <FIRST_LAUNCH value="false" />
  <ENABLE_SIMPLE value="false" />
- <media>

- <!--
-         <PREFERRED_AUDIO_ENCODING system="false" value=""/>
  -->
  <PREFERRED_AUDIO_ENCODING value="9" />
  <PREFERRED_VIDEO_ENCODING value="31" />
  <NO_AUDIO_DESCRIPTION_IN_SDP value="false" />
  <NO_VIDEO_DESCRIPTION_IN_SDP value="false" />
  <MEDIA_SOURCE value="" />
  <MEDIA_BUFFER_LENGTH value="100" />
  <IP_ADDRESS value="" />
  <AUDIO_PORT value="" />
  <VIDEO_PORT value="" />
  </media>
- <sip>
  <PUBLIC_ADDRESS value="sip:bob@ims3.ftw.at" />
```

```
  <TRANSPORT value="TCP" />
  <REGISTRAR_ADDRESS value="ims3.ftw.at" />
  <USER_NAME value="bob" />
  <STACK_PATH value="gov.nist" />
  <PREFERRED_LOCAL_PORT value="5060" />
  <DISPLAY_NAME value="bob" />
  <REGISTRAR_TRANSPORT value="UDP" />
  <REGISTRATIONS_EXPIRATION value="3600" />
  <REGISTRAR_PORT value="5060" />
  <FAIL_CALLS_ON_DEST_USER_MISMATCH value="false" />
  <DEFAULT_DOMAIN_NAME value="ims3.ftw.at" />
  <DEFAULT_AUTHENTICATION_REALM value="ims3.ftw.at" />
  <WAIT_UNREGISTGRATION_FOR value="1100" />
  <SAME_USER_EVERYWHERE value="true" />
  <ACCEPT_EARLY_MEDIA value="true" />
- <!--
 IMS Client
  -->
- <ims>
  <IMS_CLIENT value="true" />
  <PRIVATE_ADDRESS value="bob@ims3.ftw.at" />
  <PREFERRED_ADDRESS value="bob@ims3.ftw.at" />
  <PREFERRED_DISPLAY_NAME value="" />
  <ACCEPT_REFER_WITHOUT_SESSION_ON value="true" />
  <REFER_TIMEOUT value="60000" />
  <REG_EVENT_SUBSCRIPTION_TIMEOUT value="600000" />
  <ACCESS_TYPE value="3GPP-GERAN" />
  <DEFAULT_LOCAL_PRECONDITION value="sendrecv" />
  <VIDEO_BANDWIDTH value="75" />
  <AUDIO_BANDWIDTH value="25" />
- <!--
 testing feature
  -->
  <PCSCF_DYNAMIC_DISCOVERY value="false" />
- <sec>
  <PRIVACY value="" />
  <AUTH_ALGORITHM value="" />
- <!--
 operator ID : 16 bytes (32 hex char)
  -->
  <OPERATOR_ID value="00000000000000000000000000000000" />
- <!--
 if SECURITY_AGREEMENT value is empty, Security-Client header is
```

```
 not sent and IPSec not setup
  -->
- <!--
SECURITY_AGREEMENT value="ipsec-3gpp"/
  -->
  <SECURITY_AGREEMENT value="" />
  <SECURITY_ALGORITHM value="hmac-sha-1-96" />
  <SECURITY_ENCRYPT_ALG value="" />
  <SECURITY_SPI_C value="10002" />
  <SECURITY_SPI_S value="10004" />
- <!--
 if SECURITY_PORT_C empty, port-c parameter is filled with
 PREFERRED_LOCAL_PORT
  -->
  <SECURITY_PORT_C value="" />
  <SECURITY_PORT_S value="5064" />
  <REJECT_MALFORMED_NONCE value="false" />
  </sec>
  </ims>
- <!--
 end of IMS Client
  -->
- <simple>
  <CONTACT_LIST_FILE value="contact-list.xml" />
  <SUBSCRIPTION_EXP_TIME value="150" />
  <MIN_EXP_TIME value="60" />
  <LAST_SELECTED_OPEN_STATUS value="busy" />
  </simple>
  <EXCESSIVE_URI_CHARACTERS value="( )" />
  </sip>
- <gui>
  <AUTH_WIN_TITLE value="SIP Authentication!" />
  <AUTHENTICATION_PROMPT value="Please enter login name and password
  for the specified realm:" />
  <USER_NAME_LABEL value="UserName:" />
  <USER_NAME_EXAMPLE value="Example: testes1" />
  <PASSWORD_LABEL value="Password:" />
- <!--
GUI_MODE value="PhoneUiMode"/
  -->
  <PHONE_GUI_MODE value="true" />
  <IM_GUI_MODE value="false" />
- <imp>
```

```
  <CONTACT_LIST_X value="185" />
  <CONTACT_LIST_Y value="165" />
  <CONTACT_LIST_WIDTH value="157" />
  <CONTACT_LIST_HEIGHT value="433" />
  </imp>
- <logo>
  <MAIN_LOGO value="default-logo.jpg" />
  <PHONE_GUI_ICON value="default-icon.jpg" />
  <IM_GUI_ICON value="presence-icon.jpg" />
  </logo>
  </gui>
- <common>
  <PREFERRED_NETWORK_INTERFACE value="eth1" />
  <PREFERRED_NETWORK_ADDRESS value="213.235.220.99" />
  </common>
- <!--
    net.java.sip.communicator.sipphone.IS_RUNNING_SIPPHONE=false
    net.java.sip.communicator.sipphone.MY_SIPPHONE_URL=http://
    my.sipphone.com
  -->
- <sipphone>
  <IS_RUNNING_SIPPHONE value="false" />
  <MY_SIPPHONE_URL value="http://my.sipphone.com" />
  <USER_NAME_EXAMPLE value="testes1" />
  </sipphone>
- <!--
net.java.sip.communicator.gui.AUTH_WIN_TITLE=SIP Authentication!
net.java.sip.communicator.gui.AUTHENTICATION_PROMPT=Please enter
login name and password for the specified realm:
net.java.sip.communicator.gui.USER_NAME_LABEL=SIPphone Number:
net.java.sip.communicator.sipphone.USER_NAME_EXAMPLE=Example:
1-747-555-1212
net.java.sip.communicator.gui.PASSWORD_LABEL=Password:
  -->
- <!--
    net.java.sip.communicator.STUN_SERVER_ADDRESS=stun01.sipphone.com
    net.java.sip.communicator.STUN_SERVER_PORT=3478
    net.java.sip.communicator.VOICE_MAIL_ADDRESS=17475551212
  -->
- <!--
STUN_SERVER_ADDRESS value="stun01.sipphone.com"/
  -->
- <!--
```

```
STUN_SERVER_PORT value="3478"/
  -->
  <VOICE_MAIL_ADDRESS value="17475551212" />
  </communicator>
  </sip>
  </java>
  </net>
- <gov>
- <nist>
- <javax>
- <sip>
  <SERVER_LOG value="log/ims-communicator.stack.log" />
  <TRACE_LEVEL value="16" />
  </sip>
  </javax>
  </nist>
  </gov>
- <javax>
- <sip>
  <IP_ADDRESS value="213.235.220.99" />
  <STACK_NAME value="sip-communicator" />
  <ROUTER_PATH value="net.java.sip.communicator.sip.SipCommRouter" />
- <!--
 important: sip outbound proxy format is <address>:<port>/<transport>
  -->
  <OUTBOUND_PROXY value="213.235.220.100:5060/tcp" />
  <RETRANSMISSON_FILTER value="" />
  <EXTENSION_METHODS value="" />
  <RETRANSMISSION_FILTER value="true" />
  </sip>
  </javax>
- <java>
- <net>
  <preferIPv4Stack system="true" value="true" />
  <preferIPv6Addresses system="false" value="false" />
  </net>
  </java>
  </configuration>
```

Appendix C

IMS-SIP Clients

Client	Homepage	Features	Supported Platforms	Video support
x-lite, x-pro	http://www.xten.com/index.php?menu=download	Enhanced Quality of Service (QoS) for voice & video calls	Windows® 2000 / XP	video
MiniSIP	http://www.minisip.org/index.html.		Linux PC Linux familiar IPAQ PDA, Windows XP (soon Windows Mobile 2003 SE)	video
OpenWengo	http://www.openwengo.com/.			
Kphone	http://sourceforge.net/projects/kphone .		Linux	
Microsoft portrait	http://research.microsoft.com/~jiangli/portrait/.		Windows 98 Windows ME Windows 2000 Windows XP	video
Ubiquity User Agent	http://www.sipcenter.com/sip.nsf/html/User+Agent+Downloads		Windows	
EZ-Phone (Evaluation Version)	http://www.hssworld.com/voip/download.htm		Windows NT	
MySIP	http://www.mysip.ch/			
SJPhone	http://www.sjlabs.com/.	QOS support	MS Windows	video
Linphone	http://www.linphone.org/		Linux	video
Vovida	Vovida.org	Vocal: UA by Sipset	Linux	video
Siphon	http://siphon.sourceforge.net/index.html		Linux	
ActXPhone	http://www.pernau.at/kd/voip/ActXPhone/.		Windows XP. For Windows 2000	
sipXphone	sipfoundry.org.		Windows 98, 2000, NT 4 with Service Pack 4 and XP	
SIPPS	http://www.sippstar.com/.			
MSN Messenger	http://messenger.microsoft.com;		Windows	video
Shtoom	http://divmod.org/projects/shtoom.		Windows, Linux/Unix and Mac OS X.	no
Cornfed SIP-UA	http://www.cornfed.com/products/.		Linux	
PhoneGaim	http://www.phonegaim.com/.			
SFLphone	http://www.sflphone.org.		Linux	
ENUM softphone	http://www.enum.at/index.php?id=292.		windows	

Client	Homepage	Features	Supported Platforms	Video support
Phoner	http://www.phoner.de.		Microsoft Windows 95/98/ME/NT40/2000/XP	
Twinkle	http://www.twinklephone.com/.		Linux	
BOL SIP Phone	http://www.bol2000.com/download/sipphone/.			
sipXezPhone	http://www.sipfoundry.org/sipXezPhone/.		WIN32; however, sipXtapi can be built and used under Linux and MacOs X.	video
sillyAnt	http://www.sillyant.com/.	over mobile		
PJPHONE	http://www.pjsip.org/pjsua.htm		Windows	
PartiSIPation	http://developer.berlios.de/projects/partisipation/.		Linux	
Ekiga:	http://www.ekiga.org/.		Linux	video
eyeP Media	http://www.eyepmedia.com/.	softphone SDK	Windows	
Zfone:	http://www.philzimmermann.com/EN/zfone/index.html.	extension to any SIP phone for security provision	Windows Mac OS X Linux.	
Tapioca:	http://tapioca-voip.sourceforge.net/.			
Kapanga:	http://www.kapanga.net/.	Measure VoIP Quality of Service in real-time. Kapanga calculates the Mean Opinion Score (MOS) parameter in real-time. MOS is a very useful measure of quality of service that goes beyond having just hearing "noise" or "chopping". Service providers find MOS field measurements of great value to enhance customer service and increase network reliability.	Windows	video
ATL Softphone	http://www.opensourcesip.org:8080/jiveforums/sparkplugin.jsp	an open source Java instant messaging client	Windows	
YateClient:	http://voip.null.ro/pmwiki/index.php?n=Main.YateClient.		Linux, Windows	
Zoiper:	http://www.zoiper.com/.		Windows® 2000, XP and later; Mac OS; Linux	
SIP-communicator:	http://sip-communicator.org/.	SIP Communicator is an audio/video Internet phone and instant messenger	Linux, Windows	

Client	Homepage	Features	Supported Platforms	Video support
UTC IMS Client:	:http://uctimsclient.berlios.de/.	The UCT IMS Client is designed to be used in conjunction with the Fraunhofer FOKUS Open IMS Core The client supports AKA authentication, and tries to emulate IMS signaling as far as possible. The current version supports voice calls	Linux	
Lynxphone:	http://www.bitlynx.com/lynxphone.php.	QOS support (intserve, diffserve, 802.1p) Adaptive jitter buffer & packet-loss concealment	Microsoft Windows 98SE, ME, 2000, XP, 2003, Vista Apple OS X 10.3 or higher (PPC & Intel) Fedora Core 4 x86 NetBSD/i386 3.0 FreeBSD/i386 6.0	
ims-communicator	http://imscommunicator.berlios.de/	*Authentication* *a.- authentication algorithm AKAv1* *b.- subscription to the "reg" event package* *c.- Security Agreement mechanism* *IMS Session Establishment* *a.- Precondition Mechanism* *b.- Early Media* *c.- Call transfer*		
Frauhofer IMS client OpenIC	http://www.fokus.fraunhofer.de/ims/components/open_ic	Supported Features: AKA, MD5 authentication Session Initiated Protocol (SIP) based signaling for all media sessions Instant Message Audio Call Active Address book ISIM (IP multimedia Services Identity Module) profile simulation Call history list management	Windows XP Linux Windows Mobile	video

Appendix D

Captures & SIP Scenario Traces

D.1 Registration

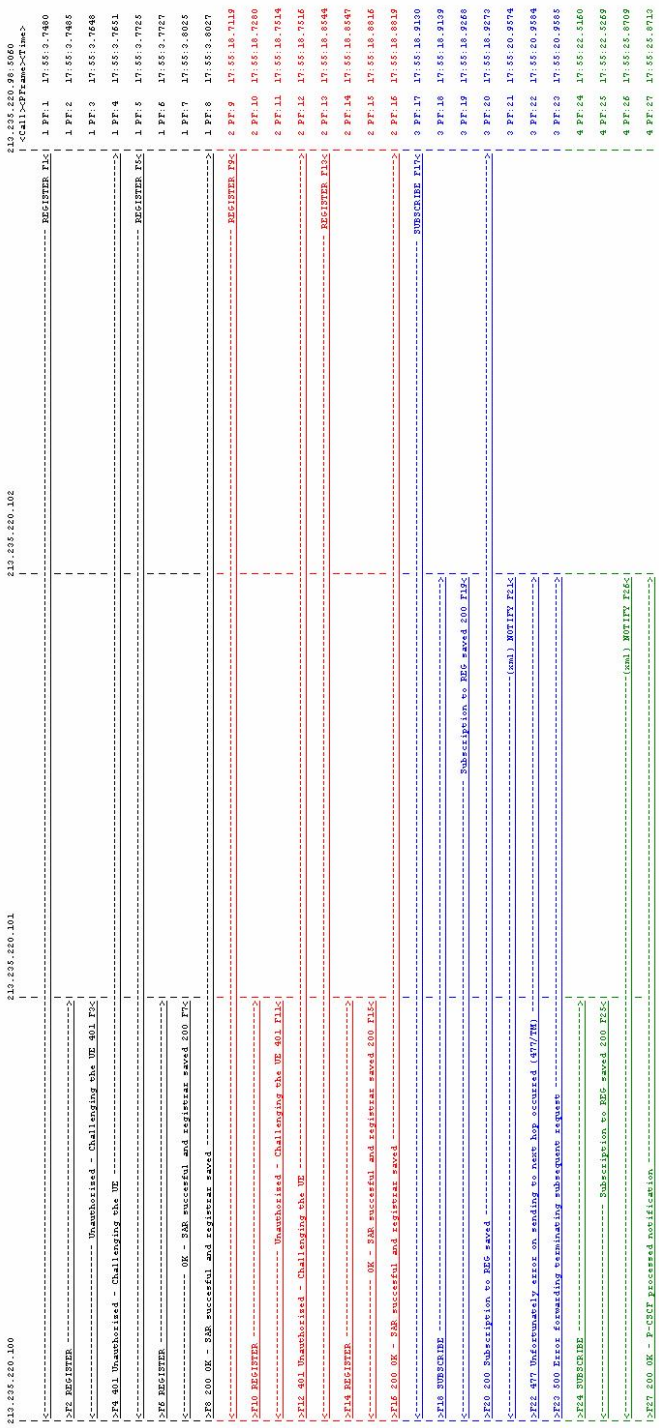


Figure D.1: Registration SIP flow

```
     SIP MESSAGE 1        213.235.220.98:5060() -> 213.235.220.100:5060()
     UDP Frame 1        21/Jul/07 17:55:3.7480 TimeFromPreviousSipFrame=0.0000 TimeFromStart=0.0000
REGISTER sip:ims3.ftw.at SIP/2.0
Call-ID: e87b4b35aa096fa180063a3cb0b87af8@213.235.220.98
Via: SIP/2.0/UDP 213.235.220.98:5060;branch=z9hG4bKa5a17d9bee68f796ef499c62384a8d80
Max-Forwards: 70
Expires: 0
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Supported: path
Contact: <sip:alice@213.235.220.98:5060>
Authorization: Digest
username="alice@ims3.ftw.at",realm="ims3.ftw.at",nonce="d42f5889ee4fd302b98524a2e5a48fe4",uri="sip:ims3.ftw.at
",response="1b75ecc247580d9c012c289460e6edd6",algorithm=MD5
From: "alice" <sip:alice@ims3.ftw.at>;tag=3916915
To: "alice" <sip:alice@ims3.ftw.at>
CSeq: 3 REGISTER
Content-Length: 0
```

```
     SIP MESSAGE 2        213.235.220.100:43281() -> 213.235.220.101:5060()
     TCP Frame 2        21/Jul/07 17:55:3.7485 TimeFromPreviousSipFrame=0.0006 TimeFromStart=0.0006
REGISTER sip:ims3.ftw.at SIP/2.0
Call-ID: e87b4b35aa096fa180063a3cb0b87af8@213.235.220.98
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa6b2.564baa4.0
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bKa5a17d9bee68f796ef499c62384a8d80
Max-Forwards: 16
Expires: 0
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Supported: path
Contact: <sip:alice@213.235.220.98:5060>
From: "alice" <sip:alice@ims3.ftw.at>;tag=3916915
To: "alice" <sip:alice@ims3.ftw.at>
CSeq: 3 REGISTER
Content-Length: 0
Path: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Require: path
P-Charging-Vector: icid-value="P-CSCFabcd46a22c57000015c7"; icid-generated-at="127.0.0.1"; orig-
ioi="ims3.ftw.at"
Authorization: Digest
username="alice@ims3.ftw.at",realm="ims3.ftw.at",nonce="d42f5889ee4fd302b98524a2e5a48fe4",uri="sip:ims3.ftw.at
",response="1b75ecc247580d9c012c289460e6edd6",algorithm=MD5, integrity-protected="no"
P-Visited-Network-ID: ims3.ftw.at
```

```
     SIP MESSAGE 3        213.235.220.101:5060() -> 213.235.220.100:43281()
     TCP Frame 3        21/Jul/07 17:55:3.7648 TimeFromPreviousSipFrame=0.0163 TimeFromStart=0.0168
SIP/2.0 401 Unauthorized - Challenging the UE
Call-ID: e87b4b35aa096fa180063a3cb0b87af8@213.235.220.98
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa6b2.564baa4.0
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bKa5a17d9bee68f796ef499c62384a8d80
From: "alice" <sip:alice@ims3.ftw.at>;tag=3916915
To: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-6a7a
CSeq: 3 REGISTER
WWW-Authenticate: Digest realm="ims3.ftw.at", nonce="f78b9498ea7c0730c64cd2ccac8c54e7", algorithm=MD5
Path: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Service-Route: <sip:orig@scscf.ims3.ftw.at:5060;lr>
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.102:5060 "Noisy feedback tells:  pid=23394 req_src_ip=213.235.220.101
req_src_port=39280 in_uri=sip:scscf.ims3.ftw.at:5060 out_uri=sip:scscf.ims3.ftw.at:5060 via_cnt==3"
```

```
     SIP MESSAGE 4        213.235.220.100:5060() -> 213.235.220.98:5060()
     UDP Frame 4        21/Jul/07 17:55:3.7651 TimeFromPreviousSipFrame=0.0003 TimeFromStart=0.0172
SIP/2.0 401 Unauthorized - Challenging the UE
Call-ID: e87b4b35aa096fa180063a3cb0b87af8@213.235.220.98
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bKa5a17d9bee68f796ef499c62384a8d80
From: "alice" <sip:alice@ims3.ftw.at>;tag=3916915
To: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-6a7a
CSeq: 3 REGISTER
Path: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Service-Route: <sip:orig@scscf.ims3.ftw.at:5060;lr>
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.102:5060 "Noisy feedback tells:  pid=23394 req_src_ip=213.235.220.101
req_src_port=39280 in_uri=sip:scscf.ims3.ftw.at:5060 out_uri=sip:scscf.ims3.ftw.at:5060 via_cnt==3"
```

```
WWW-Authenticate: Digest realm="ims3.ftw.at", nonce="f78b9498ea7c0730c64cd2ccac8c54e7", algorithm=MD5
```

```
     SIP MESSAGE 5        213.235.220.98:5060() -> 213.235.220.100:5060()
     UDP Frame 5        21/Jul/07 17:55:3.7725 TimeFromPreviousSipFrame=0.0073 TimeFromStart=0.0245
REGISTER sip:ims3.ftw.at SIP/2.0
Call-ID: e87b4b35aa096fa180063a3cb0b87af8@213.235.220.98
Via: SIP/2.0/UDP 213.235.220.98:5060;branch=z9hG4bKe7daa6f0e70e25e699242ceff1c805ae
Max-Forwards: 70
Expires: 0
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Supported: path
Contact: <sip:alice@213.235.220.98:5060>
CSeq: 4 REGISTER
Authorization: Digest
username="alice@ims3.ftw.at",realm="ims3.ftw.at",nonce="f78b9498ea7c0730c64cd2ccac8c54e7",uri="sip:ims3.ftw.at
",response="8f3235b9aa5c30641f9f3d1fa7f7ea6f",algorithm=MD5
From: "alice" <sip:alice@ims3.ftw.at>;tag=3916915
To: "alice" <sip:alice@ims3.ftw.at>
Content-Length: 0
```

```
     SIP MESSAGE 6        213.235.220.100:43281() -> 213.235.220.101:5060()
     TCP Frame 6        21/Jul/07 17:55:3.7727 TimeFromPreviousSipFrame=0.0003 TimeFromStart=0.0248
REGISTER sip:ims3.ftw.at SIP/2.0
Call-ID: e87b4b35aa096fa180063a3cb0b87af8@213.235.220.98
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK76b2.0ca62a4.0
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bKe7daa6f0e70e25e699242ceff1c805ae
Max-Forwards: 16
Expires: 0
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Supported: path
Contact: <sip:alice@213.235.220.98:5060>
CSeq: 4 REGISTER
From: "alice" <sip:alice@ims3.ftw.at>;tag=3916915
To: "alice" <sip:alice@ims3.ftw.at>
Content-Length: 0
Path: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Require: path
P-Charging-Vector: icid-value="P-CSCFabcd46a22c57000015c8"; icid-generated-at="127.0.0.1"; orig-
ioi="ims3.ftw.at"
Authorization: Digest
username="alice@ims3.ftw.at",realm="ims3.ftw.at",nonce="f78b9498ea7c0730c64cd2ccac8c54e7",uri="sip:ims3.ftw.at
",response="8f3235b9aa5c30641f9f3d1fa7f7ea6f",algorithm=MD5, integrity-protected="no"
P-Visited-Network-ID: ims3.ftw.at
```

```
     SIP MESSAGE 7        213.235.220.101:5060() -> 213.235.220.100:43281()
     TCP Frame 7        21/Jul/07 17:55:3.8025 TimeFromPreviousSipFrame=0.0297 TimeFromStart=0.0545
SIP/2.0 200 OK - SAR succesful and registrar saved
Call-ID: e87b4b35aa096fa180063a3cb0b87af8@213.235.220.98
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK76b2.0ca62a4.0
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bKe7daa6f0e70e25e699242ceff1c805ae
CSeq: 4 REGISTER
From: "alice" <sip:alice@ims3.ftw.at>;tag=3916915
To: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-cc16
Contact: <sip:alice@213.235.220.98:5060>;expires=0
Path: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Service-Route: <sip:orig@scscf.ims3.ftw.at:5060;lr>
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.102:5060 "Noisy feedback tells:  pid=23394 req_src_ip=213.235.220.101
req_src_port=39280 in_uri=sip:scscf.ims3.ftw.at:5060 out_uri=sip:scscf.ims3.ftw.at:5060 via_cnt==3"
```

```
     SIP MESSAGE 8        213.235.220.100:5060() -> 213.235.220.98:5060()
     UDP Frame 8        21/Jul/07 17:55:3.8027 TimeFromPreviousSipFrame=0.0002 TimeFromStart=0.0547
SIP/2.0 200 OK - SAR succesful and registrar saved
Call-ID: e87b4b35aa096fa180063a3cb0b87af8@213.235.220.98
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bKe7daa6f0e70e25e699242ceff1c805ae
CSeq: 4 REGISTER
From: "alice" <sip:alice@ims3.ftw.at>;tag=3916915
To: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-cc16
Contact: <sip:alice@213.235.220.98:5060>;expires=0
Path: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Service-Route: <sip:orig@scscf.ims3.ftw.at:5060;lr>
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
```

```
Content-Length: 0
Warning: 392 213.235.220.102:5060 "Noisy feedback tells:  pid=23394 req_src_ip=213.235.220.101
req_src_port=39280 in_uri=sip:scscf.ims3.ftw.at:5060 out_uri=sip:scscf.ims3.ftw.at:5060 via_cnt==3"
```

```
     SIP MESSAGE 9        213.235.220.98:5060() -> 213.235.220.100:5060()
     UDP Frame 9        21/Jul/07 17:55:18.7119 TimeFromPreviousSipFrame=14.9093 TimeFromStart=14.9640
REGISTER sip:ims3.ftw.at SIP/2.0
Call-ID: 6d46bdc4cee56ab23a7f4a8c4a002f0e@213.235.220.98
CSeq: 1 REGISTER
From: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
To: "alice" <sip:alice@ims3.ftw.at>
Via: SIP/2.0/UDP 213.235.220.98:5060;branch=z9hG4bK742d4127ff22ec88706a87b71300c939
Max-Forwards: 70
Expires: 3600
P-Access-Network-Info: IEEE-802.11
Authorization: Digest username="alice@ims3.ftw.at",realm="ims3.ftw.at",nonce="",uri="ims3.ftw.at",response=""
User-Agent: IMS-Communicator 070605
Supported: path
Contact: <sip:alice@213.235.220.98:5060>
Content-Length: 0
```

```
     SIP MESSAGE 10       213.235.220.100:43281() -> 213.235.220.101:5060()
     TCP Frame 10       21/Jul/07 17:55:18.7280 TimeFromPreviousSipFrame=0.0161 TimeFromStart=14.9800
REGISTER sip:ims3.ftw.at SIP/2.0
Call-ID: 6d46bdc4cee56ab23a7f4a8c4a002f0e@213.235.220.98
CSeq: 1 REGISTER
From: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
To: "alice" <sip:alice@ims3.ftw.at>
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKcd38.c9b414a7.0
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bK742d4127ff22ec88706a87b71300c939
Max-Forwards: 16
Expires: 3600
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Supported: path
Contact: <sip:alice@213.235.220.98:5060>
Content-Length: 0
Path: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Require: path
P-Charging-Vector: icid-value="P-CSCFabcd46a22c66000015c9"; icid-generated-at="127.0.0.1"; orig-
ioi="ims3.ftw.at"
Authorization: Digest username="alice@ims3.ftw.at",realm="ims3.ftw.at",nonce="",uri="ims3.ftw.at",response="",
integrity-protected="no"
P-Visited-Network-ID: ims3.ftw.at
```

```
     SIP MESSAGE 11       213.235.220.101:5060() -> 213.235.220.100:43281()
     TCP Frame 11       21/Jul/07 17:55:18.7514 TimeFromPreviousSipFrame=0.0235 TimeFromStart=15.0035
SIP/2.0 401 Unauthorized - Challenging the UE
Call-ID: 6d46bdc4cee56ab23a7f4a8c4a002f0e@213.235.220.98
CSeq: 1 REGISTER
From: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
To: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-e6d2
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKcd38.c9b414a7.0
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bK742d4127ff22ec88706a87b71300c939
WWW-Authenticate: Digest realm="ims3.ftw.at", nonce="ae498470a2e7c8fc80036d0fb903580a", algorithm=MD5
Path: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Service-Route: <sip:orig@scscf.ims3.ftw.at:5060;lr>
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.102:5060 "Noisy feedback tells:  pid=23395 req_src_ip=213.235.220.101
req_src_port=39280 in_uri=sip:scscf.ims3.ftw.at:5060 out_uri=sip:scscf.ims3.ftw.at:5060 via_cnt==3"
```

```
     SIP MESSAGE 12       213.235.220.100:5060() -> 213.235.220.98:5060()
     UDP Frame 12       21/Jul/07 17:55:18.7516 TimeFromPreviousSipFrame=0.0002 TimeFromStart=15.0037
SIP/2.0 401 Unauthorized - Challenging the UE
Call-ID: 6d46bdc4cee56ab23a7f4a8c4a002f0e@213.235.220.98
CSeq: 1 REGISTER
From: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
To: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-e6d2
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bK742d4127ff22ec88706a87b71300c939
Path: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Service-Route: <sip:orig@scscf.ims3.ftw.at:5060;lr>
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.102:5060 "Noisy feedback tells:  pid=23395 req_src_ip=213.235.220.101
```

```
req_src_port=39280 in_uri=sip:scscf.ims3.ftw.at:5060 out_uri=sip:scscf.ims3.ftw.at:5060 via_cnt==3"
WWW-Authenticate: Digest realm="ims3.ftw.at", nonce="ae498470a2e7c8fc80036d0fb903580a", algorithm=MD5
```

```
     SIP MESSAGE 13       213.235.220.98:5060() -> 213.235.220.100:5060()
     UDP Frame 13       21/Jul/07 17:55:18.8544 TimeFromPreviousSipFrame=0.1027 TimeFromStart=15.1064
REGISTER sip:ims3.ftw.at SIP/2.0
Call-ID: 6d46bdc4cee56ab23a7f4a8c4a002f0e@213.235.220.98
CSeq: 2 REGISTER
Via: SIP/2.0/UDP 213.235.220.98:5060;branch=z9hG4bK768d6cdcdf3637ab9e46ab654ba32469
Max-Forwards: 70
Expires: 3600
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Supported: path
Contact: <sip:alice@213.235.220.98:5060>
Authorization: Digest
username="alice@ims3.ftw.at",realm="ims3.ftw.at",nonce="ae498470a2e7c8fc80036d0fb903580a",uri="sip:ims3.ftw.at
",response="93480cdfd644c72b88905807d693329d",algorithm=MD5
From: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
To: "alice" <sip:alice@ims3.ftw.at>
Content-Length: 0
```

```
     SIP MESSAGE 14       213.235.220.100:43281() -> 213.235.220.101:5060()
     TCP Frame 14       21/Jul/07 17:55:18.8547 TimeFromPreviousSipFrame=0.0004 TimeFromStart=15.1068
REGISTER sip:ims3.ftw.at SIP/2.0
Call-ID: 6d46bdc4cee56ab23a7f4a8c4a002f0e@213.235.220.98
CSeq: 2 REGISTER
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK9d38.bbb80723.0
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bK768d6cdcdf3637ab9e46ab654ba32469
Max-Forwards: 16
Expires: 3600
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Supported: path
Contact: <sip:alice@213.235.220.98:5060>
From: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
To: "alice" <sip:alice@ims3.ftw.at>
Content-Length: 0
Path: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Require: path
P-Charging-Vector: icid-value="P-CSCFabcd46a22c66000015ca"; icid-generated-at="127.0.0.1"; orig-
ioi="ims3.ftw.at"
Authorization: Digest
username="alice@ims3.ftw.at",realm="ims3.ftw.at",nonce="ae498470a2e7c8fc80036d0fb903580a",uri="sip:ims3.ftw.at
",response="93480cdfd644c72b88905807d693329d",algorithm=MD5, integrity-protected="no"
P-Visited-Network-ID: ims3.ftw.at
```

```
     SIP MESSAGE 15       213.235.220.101:5060() -> 213.235.220.100:43281()
     TCP Frame 15       21/Jul/07 17:55:18.8816 TimeFromPreviousSipFrame=0.0269 TimeFromStart=15.1336
SIP/2.0 200 OK - SAR succesful and registrar saved
Call-ID: 6d46bdc4cee56ab23a7f4a8c4a002f0e@213.235.220.98
CSeq: 2 REGISTER
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK9d38.bbb80723.0
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bK768d6cdcdf3637ab9e46ab654ba32469
From: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
To: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-60b4
P-Associated-URI: <sip:alice@ims3.ftw.at>
Contact: <sip:alice@213.235.220.98:5060>;expires=3600
Path: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Service-Route: <sip:orig@scscf.ims3.ftw.at:5060;lr>
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.102:5060 "Noisy feedback tells:  pid=23395 req_src_ip=213.235.220.101
req_src_port=39280 in_uri=sip:scscf.ims3.ftw.at:5060 out_uri=sip:scscf.ims3.ftw.at:5060 via_cnt==3"
```

```
     SIP MESSAGE 16       213.235.220.100:5060() -> 213.235.220.98:5060()
     UDP Frame 16       21/Jul/07 17:55:18.8819 TimeFromPreviousSipFrame=0.0003 TimeFromStart=15.1339
SIP/2.0 200 OK - SAR succesful and registrar saved
Call-ID: 6d46bdc4cee56ab23a7f4a8c4a002f0e@213.235.220.98
CSeq: 2 REGISTER
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bK768d6cdcdf3637ab9e46ab654ba32469
From: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
To: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-60b4
P-Associated-URI: <sip:alice@ims3.ftw.at>
Contact: <sip:alice@213.235.220.98:5060>;expires=3600
```

```
Path: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Service-Route: <sip:orig@scscf.ims3.ftw.at:5060;lr>
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.102:5060 "Noisy feedback tells:  pid=23395 req_src_ip=213.235.220.101
req_src_port=39280 in_uri=sip:scscf.ims3.ftw.at:5060 out_uri=sip:scscf.ims3.ftw.at:5060 via_cnt==3"
```

```
     SIP MESSAGE 17       213.235.220.98:5060() -> 213.235.220.100:5060()
     UDP Frame 17       21/Jul/07 17:55:18.9130 TimeFromPreviousSipFrame=0.0312 TimeFromStart=15.1651
SUBSCRIBE sip:alice@ims3.ftw.at SIP/2.0
Call-ID: c7f8c9b5a625cbf30e196b51da6334a2@213.235.220.98
CSeq: 1 SUBSCRIBE
From: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
To: "alice" <sip:alice@ims3.ftw.at>
Via: SIP/2.0/UDP 213.235.220.98:5060;branch=z9hG4bK4814f6d90b2957f43ca9d6b1a054e6fd
Max-Forwards: 70
Contact: <sip:alice@213.235.220.98:5060>
Event:  reg
Expires: 600000
P-Access-Network-Info: IEEE-802.11
Route: <sip:213.235.220.100:5060;transport=udp>,<sip:orig@scscf.ims3.ftw.at:5060;lr>
P-Preferred-Identity: <sip:alice@ims3.ftw.at>
Require: sec-agree
Proxy-Require: sec-agree
Accept: application/reginfo+xml
User-Agent: IMS-Communicator 070605
Content-Length: 0
```

```
     SIP MESSAGE 18       213.235.220.100:43282() -> 213.235.220.102:5060()
     TCP Frame 18       21/Jul/07 17:55:18.9139 TimeFromPreviousSipFrame=0.0009 TimeFromStart=15.1660
SUBSCRIBE sip:alice@ims3.ftw.at SIP/2.0
Record-Route: <sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Call-ID: c7f8c9b5a625cbf30e196b51da6334a2@213.235.220.98
CSeq: 1 SUBSCRIBE
From: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
To: "alice" <sip:alice@ims3.ftw.at>
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKb137.8191fb64.0
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bK4814f6d90b2957f43ca9d6b1a054e6fd
Max-Forwards: 16
Contact: <sip:alice@213.235.220.98:5060>
Event:  reg
Expires: 600000
P-Access-Network-Info: IEEE-802.11
Route: <sip:orig@scscf.ims3.ftw.at:5060;lr>
Require: sec-agree
Proxy-Require: sec-agree
Accept: application/reginfo+xml
User-Agent: IMS-Communicator 070605
Content-Length: 0
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
P-Charging-Vector: icid-value="P-CSCFabcd46a22c66000015cb"; icid-generated-at="127.0.0.1"; orig-
ioi="ims3.ftw.at"
```

```
     SIP MESSAGE 19       213.235.220.102:5060() -> 213.235.220.100:43282()
     TCP Frame 19       21/Jul/07 17:55:18.9268 TimeFromPreviousSipFrame=0.0129 TimeFromStart=15.1789
SIP/2.0 200 Subscription to REG saved
Record-Route: <sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Call-ID: c7f8c9b5a625cbf30e196b51da6334a2@213.235.220.98
CSeq: 1 SUBSCRIBE
From: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
To: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-8733
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKb137.8191fb64.0
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bK4814f6d90b2957f43ca9d6b1a054e6fd
Expires: 600000
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.102:5060 "Noisy feedback tells:  pid=23396 req_src_ip=213.235.220.100
req_src_port=43282 in_uri=sip:alice@ims3.ftw.at out_uri=sip:alice@ims3.ftw.at via_cnt==2"
```

```
     SIP MESSAGE 20       213.235.220.100:5060() -> 213.235.220.98:5060()
     UDP Frame 20       21/Jul/07 17:55:18.9273 TimeFromPreviousSipFrame=0.0004 TimeFromStart=15.1793
SIP/2.0 200 Subscription to REG saved
Record-Route: <sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Call-ID: c7f8c9b5a625cbf30e196b51da6334a2@213.235.220.98
CSeq: 1 SUBSCRIBE
From: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
```

```
To: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-8733
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bK4814f6d90b2957f43ca9d6b1a054e6fd
Expires: 600000
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.102:5060 "Noisy feedback tells:  pid=23396 req_src_ip=213.235.220.100
req_src_port=43282 in_uri=sip:alice@ims3.ftw.at out_uri=sip:alice@ims3.ftw.at via_cnt==2"
```

```
     SIP MESSAGE 21       213.235.220.102:37430() -> 213.235.220.100:5060()
     TCP Frame 21       21/Jul/07 17:55:20.9574 TimeFromPreviousSipFrame=2.0301 TimeFromStart=17.2094
NOTIFY sip:alice@213.235.220.98:5060 SIP/2.0
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK10aa.d07511c5.0
To: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
From: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-8733
CSeq: 10 NOTIFY
Call-ID: c7f8c9b5a625cbf30e196b51da6334a2@213.235.220.98
Route: <sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Content-Length: 333
User-Agent: Sip EXpress router(2.1.0-dev1 OpenIMSCore (i386/linux))
Contact: <sip:scscf.ims3.ftw.at:5060>
Event: reg
Max-Forwards: 70
Subscription-State: active;expires=3630
Content-Type: application/reginfo+xml

<?xml version="1.0"?>
<reginfo xmlns="urn:ietf:params:xml:ns:reginfo" version="0" state="full">
[09]<registration aor="sip:alice@ims3.ftw.at" id="0x4060ce94" state="active">
[09][09]<contact id="0x4064dd88" state="active" event="registered" expires="3600">
[09][09][09]<uri>sip:alice@213.235.220.98:5060</uri>
[09][09]</contact>
[09]</registration>
</reginfo>
```

```
     SIP MESSAGE 22       213.235.220.100:5060() -> 213.235.220.102:37430()
     TCP Frame 22       21/Jul/07 17:55:20.9584 TimeFromPreviousSipFrame=0.0011 TimeFromStart=17.2105
SIP/2.0 477 Unfortunately error on sending to next hop occurred (477/TM)
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK10aa.d07511c5.0;rport=37430
To: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
From: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-8733
CSeq: 10 NOTIFY
Call-ID: c7f8c9b5a625cbf30e196b51da6334a2@213.235.220.98
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.100:5060 "Noisy feedback tells:  pid=32669 req_src_ip=213.235.220.102
req_src_port=37430 in_uri=sip:alice@213.235.220.98:5060 out_uri=sip:alice@213.235.220.98:5060 via_cnt==1"
```

```
     SIP MESSAGE 23       213.235.220.100:5060() -> 213.235.220.102:37430()
     TCP Frame 23       21/Jul/07 17:55:20.9585 TimeFromPreviousSipFrame=0.0001 TimeFromStart=17.2105
SIP/2.0 500 Error forwarding terminating subsequent request
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK10aa.d07511c5.0;rport=37430
To: "alice" <sip:alice@ims3.ftw.at>;tag=2719739
From: "alice" <sip:alice@ims3.ftw.at>;tag=6958d7b48b96070a68c4b4c728bebbcf-8733
CSeq: 10 NOTIFY
Call-ID: c7f8c9b5a625cbf30e196b51da6334a2@213.235.220.98
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.100:5060 "Noisy feedback tells:  pid=32669 req_src_ip=213.235.220.102
req_src_port=37430 in_uri=sip:alice@213.235.220.98:5060 out_uri=sip:alice@213.235.220.98:5060 via_cnt==1"
```

```
     SIP MESSAGE 24       213.235.220.100:43281() -> 213.235.220.101:5060()
     TCP Frame 24       21/Jul/07 17:55:22.5160 TimeFromPreviousSipFrame=1.5575 TimeFromStart=18.7680
SUBSCRIBE sip:alice@ims3.ftw.at SIP/2.0
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK7d32.c1dd0453.0
To: sip:alice@ims3.ftw.at;tag=6958d7b48b96070a68c4b4c728bebbcf-2862
From: sip:p-cscf.ims3.ftw.at:5060;tag=aa7e34480ba2e4b2b69e2d91e559d4c3-d5f2
CSeq: 5 SUBSCRIBE
Call-ID: 6e7fff3f-32667@213.235.220.100
User-Agent: Sip EXpress router(2.1.0-dev1 OpenIMSCore (i386/linux))
Event: reg
Accept: application/reginfo+xml
Content-Length: 0
Max-Forwards: 10
Expires: 3630
Contact: <sip:p-cscf.ims3.ftw.at:5060>
P-Asserted-Identity: <sip:term@p-cscf.ims3.ftw.at:5060>
```

```
     SIP MESSAGE 25       213.235.220.101:5060() -> 213.235.220.100:43281()
     TCP Frame 25       21/Jul/07 17:55:22.5269 TimeFromPreviousSipFrame=0.0110 TimeFromStart=18.7790
SIP/2.0 200 Subscription to REG saved
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK7d32.c1dd0453.0
To: sip:alice@ims3.ftw.at;tag=6958d7b48b96070a68c4b4c728bebbcf-2862
From: sip:p-cscf.ims3.ftw.at:5060;tag=aa7e34480ba2e4b2b69e2d91e559d4c3-d5f2
CSeq: 5 SUBSCRIBE
Call-ID: 6e7fff3f-32667@213.235.220.100
Expires: 3630
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.102:5060 "Noisy feedback tells:  pid=23395 req_src_ip=213.235.220.101
req_src_port=39280 in_uri=sip:alice@ims3.ftw.at out_uri=sip:alice@ims3.ftw.at via_cnt==2"
```

```
     SIP MESSAGE 26       213.235.220.102:37430() -> 213.235.220.100:5060()
     TCP Frame 26       21/Jul/07 17:55:25.8709 TimeFromPreviousSipFrame=3.3440 TimeFromStart=22.1229
NOTIFY sip:p-cscf.ims3.ftw.at:5060 SIP/2.0
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK3ca5.6bd16714.0
To: sip:p-cscf.ims3.ftw.at:5060;tag=aa7e34480ba2e4b2b69e2d91e559d4c3-d5f2
From: sip:alice@ims3.ftw.at;tag=6958d7b48b96070a68c4b4c728bebbcf-3f71
CSeq: 10 NOTIFY
Call-ID: 6e7fff3f-32667@213.235.220.100
Content-Length: 333
User-Agent: Sip EXpress router(2.1.0-dev1 OpenIMSCore (i386/linux))
Contact: <sip:scscf.ims3.ftw.at:5060>
Event: reg
Max-Forwards: 70
Subscription-State: active;expires=3626
Content-Type: application/reginfo+xml

<?xml version="1.0"?>
<reginfo xmlns="urn:ietf:params:xml:ns:reginfo" version="0" state="full">
[09]<registration aor="sip:alice@ims3.ftw.at" id="0x4060ce94" state="active">
[09][09]<contact id="0x4064dd88" state="active" event="registered" expires="3596">
[09][09][09]<uri>sip:alice@213.235.220.98:5060</uri>
[09][09]</contact>
[09]</registration>
</reginfo>
```

```
     SIP MESSAGE 27       213.235.220.100:5060() -> 213.235.220.102:37430()
     TCP Frame 27       21/Jul/07 17:55:25.8713 TimeFromPreviousSipFrame=0.0005 TimeFromStart=22.1234
SIP/2.0 200 OK - P-CSCF processed notification
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK3ca5.6bd16714.0;rport=37430
To: sip:p-cscf.ims3.ftw.at:5060;tag=aa7e34480ba2e4b2b69e2d91e559d4c3-d5f2
From: sip:alice@ims3.ftw.at;tag=6958d7b48b96070a68c4b4c728bebbcf-3f71
CSeq: 10 NOTIFY
Call-ID: 6e7fff3f-32667@213.235.220.100
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.100:5060 "Noisy feedback tells:  pid=32669 req_src_ip=213.235.220.102
req_src_port=37430 in_uri=sip:p-cscf.ims3.ftw.at:5060 out_uri=sip:p-cscf.ims3.ftw.at:5060 via_cnt==1"
```

D.2 477 Error Message

```
     SIP MESSAGE 1        213.235.220.98:5060() -> 213.235.220.100:5060()
     UDP Frame 1        16/Jul/07 17:09:52.1690 TimeFromPreviousSipFrame=0.0000
TimeFromStart=0.0000
INVITE sip:bob@ims3.ftw.at SIP/2.0
Call-ID: 60e0a34fdca07eec9c484f3a69857535@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=30308427
To: <sip:bob@ims3.ftw.at>
Via: SIP/2.0/UDP 213.235.220.98:5060;branch=z9hG4bK1e03c0cf90f1cdab97705a1a2b622348
Max-Forwards: 70
Contact: <sip:alice@213.235.220.98:5060>
Route: <sip:213.235.220.100:5060;transport=udp>,<sip:orig@scscf.ims3.ftw.at:5060;lr>
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
P-Preferred-Identity: <sip:alice@ims3.ftw.at>
Supported: 100rel,precondition,early-session
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Content-Type: application/sdp
Content-Length: 670

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
m=video 22222 RTP/AVP 34 26 31
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
```

```
     SIP MESSAGE 2        213.235.220.100:5060() -> 213.235.220.98:5060()
     UDP Frame 2        16/Jul/07 17:09:52.1848 TimeFromPreviousSipFrame=0.0158
TimeFromStart=0.0158
SIP/2.0 100 trying -- your call is important to us
Call-ID: 60e0a34fdca07eec9c484f3a69857535@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=30308427
To: <sip:bob@ims3.ftw.at>
Via: SIP/2.0/UDP 213.235.220.98:5060;branch=z9hG4bK1e03c0cf90f1cdab97705a1a2b622348;rport=5060
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.100:5060 "Noisy feedback tells:  pid=32664 req_src_ip=213.235.220.98
req_src_port=5060 in_uri=sip:bob@ims3.ftw.at out_uri=sip:bob@ims3.ftw.at via_cnt==1"
```

```
     SIP MESSAGE 3        213.235.220.100:5060() -> 213.235.220.98:5060()
     UDP Frame 3        16/Jul/07 17:09:52.1961 TimeFromPreviousSipFrame=0.0113
TimeFromStart=0.0272
SIP/2.0 477 Unfortunately error on sending to next hop occurred (477/TM)
Call-ID: 60e0a34fdca07eec9c484f3a69857535@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=30308427
```

```
To: <sip:bob@ims3.ftw.at>;tag=f5719eda0d682cbb2a3ef71b7d042d72-bf9e
Via: SIP/2.0/UDP
213.235.220.98:5060;received=213.235.220.98;rport=5060;branch=z9hG4bK1e03c0cf90f1cdab97705a1a2b6223
48
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.100:5060 "Noisy feedback tells:  pid=32670 req_src_ip=213.235.220.102
req_src_port=37182 in_uri=sip:bob@213.235.220.99:5060 out_uri=sip:bob@213.235.220.99:5060
via_cnt==4"
```

```
     SIP MESSAGE 4        213.235.220.98:5060() -> 213.235.220.100:5060()
     UDP Frame 4        16/Jul/07 17:09:52.1990 TimeFromPreviousSipFrame=0.0029
TimeFromStart=0.0301
ACK sip:bob@ims3.ftw.at SIP/2.0
Call-ID: 60e0a34fdca07eec9c484f3a69857535@213.235.220.98
CSeq: 1 ACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=30308427
To: <sip:bob@ims3.ftw.at>;tag=f5719eda0d682cbb2a3ef71b7d042d72-bf9e
Via: SIP/2.0/UDP 213.235.220.98:5060;branch=z9hG4bK1e03c0cf90f1cdab97705a1a2b622348
Max-Forwards: 70
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
P-Preferred-Identity: <sip:alice@ims3.ftw.at>
Supported: 100rel,precondition,early-session
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Content-Length: 0
```

```
213.235.220.100:5060                                                    213.235.220.98:5060
|                                                                       | <Call><PFrame><Time>
|                                                                       |
|<-------------------------------------------------------(sdp) INVITE F1<|  1 PF:1   17:09:52.1690
|                                                                       |
|>F2 100 trying -- your call is important to us ----------------------->|  1 PF:2   17:09:52.1848
|                                                                       |
|>F3 477 Unfortunately error on sending to next hop occurred (477/TM) ->|  1 PF:3   17:09:52.1961
|                                                                       |
|<------------------------------------------------------------- ACK F4<|  1 PF:4   17:09:52.1990
```

Figure D.2: Registration SIP flow

D.3 Basic Call: Alice − > Bob

```
213.235.220.102                          213.235.220.100                          213.235.220.99                           213.235.220.98
|                                        |                                        |                                        | <Call><PFrame><Time>
|                                        |                                        |                                        |
|                                        |<---------------------------------------------------------------------------(sdp) INVITE F1<|  1 PF:69  14:40:33.3948
|                                        |                                        |
|                                        |>F2 100 trying -- your call is important to us ---------------------------------------------->|  1 PF:70  14:40:33.4068
|                                        |                                        |                                        |
|<--------------------------------(sdp) INVITE F3<|                                |                                        |  1 PF:72  14:40:33.4069
|                                        |                                        |                                        |
|>F4 100 trying -- your call is important to us ->|                                |                                        |  1 PF:85  14:40:33.4078
|                                        |                                        |                                        |
|>F5 INVITE (sdp)------------------------------->|                                |                                        |  1 PF:108 14:40:33.4087
|                                        |                                        |                                        |
|<- trying -- your call is important to us 100 F6<|                               |                                        |  1 PF:112 14:40:33.4097
|                                        |                                        |                                        |
|                                        |>F7 INVITE (sdp)------------------------------->|                                |  1 PF:114 14:40:33.4099
|                                        |                                        |                                        |
|                                        |<------------------(sdp) Session progress 183 F8<|                               |  1 PF:122 14:40:33.4354
|                                        |                                        |                                        |
|<------------------(sdp) Session progress 183 F9<|                               |                                        |  1 PF:125 14:40:33.4360
|                                        |                                        |                                        |
|>F10 183 Session progress (sdp)------------------>|                               |                                        |  1 PF:136 14:40:33.4369
|                                        |                                        |                                        |
|                                        |>F11 183 Session progress (sdp)-------------------------------------------------------------->|  1 PF:138 14:40:33.4374
|                                        |                                        |                                        |
|                                        |<---------------------------------------------------------------------------(sdp) PRACK F12<|  1 PF:142 14:40:33.4625
|                                        |                                        |                                        |
|<--------------------------------(sdp) PRACK F13<|                               |                                        |  1 PF:145 14:40:33.4634
|                                        |                                        |                                        |
|>F14 PRACK (sdp)------------------------------->|                                |                                        |  1 PF:152 14:40:33.4643
|                                        |                                        |                                        |
|                                        |>F15 PRACK (sdp)------------------------------->|                                |  1 PF:155 14:40:33.4650
|                                        |                                        |                                        |
|                                        |<------------------------------(sdp) OK 200 F16<|                                |  1 PF:158 14:40:33.4740
|                                        |                                        |                                        |
|<------------------------------(sdp) OK 200 F17<|                                |                                        |  1 PF:160 14:40:33.4745
|                                        |                                        |                                        |
|>F18 200 OK (sdp)------------------------------>|                                |                                        |  1 PF:166 14:40:33.4753
|                                        |                                        |                                        |
|                                        |>F19 200 OK (sdp)---------------------------------------------------------------------------->|  1 PF:167 14:40:33.4759
|                                        |                                        |
|                                        |<---------------------------------------------------------------------------(sdp) UPDATE F20<|  1 PF:169 14:40:33.4870
|                                        |                                        |                                        |
|<------------------------------(sdp) UPDATE F21<|                                |                                        |  1 PF:172 14:40:33.4880
|                                        |                                        |                                        |
|>F22 UPDATE (sdp)------------------------------>|                                |                                        |  1 PF:179 14:40:33.4889
|                                        |                                        |                                        |
|                                        |>F23 UPDATE (sdp)------------------------------>|                                |  1 PF:182 14:40:33.4897
|                                        |                                        |                                        |
|                                        |<------------------------------(sdp) OK 200 F24<|                                |  1 PF:184 14:40:33.4986
|                                        |                                        |                                        |
|<------------------------------(sdp) OK 200 F25<|                                |                                        |  1 PF:185 14:40:33.4991
|                                        |                                        |                                        |
|>F26 200 OK (sdp)------------------------------>|                                |                                        |  1 PF:192 14:40:33.4999
|                                        |                                        |                                        |
|                                        |>F27 200 OK (sdp)---------------------------------------------------------------------------->|  1 PF:193 14:40:33.5005
|                                        |                                        |
|                                        |<------------------------------ Ringing 180 F28<|                                |  1 PF:195 14:40:33.5391
|                                        |                                        |                                        |
|<------------------------------ Ringing 180 F29<|                                |                                        |  1 PF:199 14:40:33.5398
|                                        |                                        |                                        |
|>F30 180 Ringing ------------------------------>|                                |                                        |  1 PF:206 14:40:33.5404
|                                        |                                        |                                        |
|                                        |>F31 180 Ringing ----------------------------------------------------------------------------->|  1 PF:208 14:40:33.5409
|                                        |                                        |
|                                        |<---------------------------------------------------------------------------- PRACK F32<|  1 PF:210 14:40:33.5512
|                                        |                                        |                                        |
|<------------------------------------ PRACK F33<|                                |                                        |  1 PF:211 14:40:33.5520
|                                        |                                        |                                        |
|>F34 PRACK ------------------------------------>|                                |                                        |  1 PF:216 14:40:33.5526
|                                        |                                        |                                        |
|                                        |>F35 PRACK ------------------------------------>|                                |  1 PF:217 14:40:33.5533
|                                        |                                        |                                        |
|                                        |<------------------------------------ OK 200 F36<|                                |  1 PF:218 14:40:33.5594
|                                        |                                        |                                        |
|<------------------------------------ OK 200 F37<|                                |                                        |  1 PF:219 14:40:33.5599
|                                        |                                        |                                        |
|>F38 200 OK ------------------------------------>|                                |                                        |  1 PF:224 14:40:33.5604
|                                        |                                        |                                        |
|                                        |>F39 200 OK ---------------------------------------------------------------------------------->|  1 PF:226 14:40:33.5610
|                                        |                                        |
|                                        |<------------------------------------ OK 200 F40<|                                |  1 PF:273 14:40:40.2155
|                                        |                                        |                                        |
|<------------------------------------ OK 200 F41<|                                |                                        |  1 PF:275 14:40:40.2162
|                                        |                                        |                                        |
|>F42 200 OK ------------------------------------>|                                |                                        |  1 PF:283 14:40:40.2170
|                                        |                                        |                                        |
|                                        |>F43 200 OK ---------------------------------------------------------------------------------->|  1 PF:287 14:40:40.2176
|                                        |                                        |
|                                        |<---------------------------------------------------------------------------------- ACK F44<|  1 PF:289 14:40:40.2229
|                                        |                                        |                                        |
|<------------------------------------ ACK F45<|                                |                                        |  1 PF:290 14:40:40.2235
|                                        |                                        |                                        |
|>F46 ACK --------------------------------------->|                                |                                        |  1 PF:295 14:40:40.2241
|                                        |                                        |                                        |
|                                        |>F47 ACK --------------------------------------->|                                |  1 PF:296 14:40:40.2246
|                                        |                                        |
|                                        |<---------------------------------------------------------------------------------- BYE F48<|  1 PF:373 14:40:51.7925
|                                        |                                        |                                        |
|<------------------------------------ BYE F49<|                                |                                        |  1 PF:375 14:40:51.7933
|                                        |                                        |                                        |
|>F50 BYE --------------------------------------->|                                |                                        |  1 PF:381 14:40:51.7941
|                                        |                                        |                                        |
|                                        |>F51 BYE --------------------------------------->|                                |  1 PF:383 14:40:51.7947
|                                        |                                        |
|                                        |<------------------------------------ OK 200 F52<|                                |  1 PF:385 14:40:52.7989
|                                        |                                        |                                        |
|<------------------------------------ OK 200 F53<|                                |                                        |  1 PF:386 14:40:52.8001
|                                        |                                        |                                        |
|>F54 200 OK ------------------------------------>|                                |                                        |  1 PF:395 14:40:52.8007
|                                        |                                        |                                        |
|                                        |>F55 200 OK ---------------------------------------------------------------------------------->|  1 PF:396 14:40:52.8012
```

Figure D.3: Basic Call SIP flow

```
     SIP MESSAGE 1        213.235.220.98:47858() -> 213.235.220.100:5060()
     TCP Frame 69       17/Jul/07 14:40:33.3948 TimeFromPreviousSipFrame=12.5152 TimeFromStart=12.5152
INVITE sip:bob@ims3.ftw.at SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>
Via: SIP/2.0/TCP 213.235.220.98:5060;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Max-Forwards: 70
Contact: <sip:alice@213.235.220.98:5060>
Route: <sip:213.235.220.100:5060;transport=tcp>,<sip:orig@scscf.ims3.ftw.at:5060;lr>
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
P-Preferred-Identity: <sip:alice@ims3.ftw.at>
Supported: 100rel,precondition,early-session
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Content-Type: application/sdp
Content-Length: 671

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
```

```
     SIP MESSAGE 2        213.235.220.100:5060() -> 213.235.220.98:47858()
     TCP Frame 70       17/Jul/07 14:40:33.4068 TimeFromPreviousSipFrame=0.0119 TimeFromStart=12.5271
SIP/2.0 100 trying -- your call is important to us
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>
Via: SIP/2.0/TCP 213.235.220.98:5060;branch=z9hG4bK8e0920833a7217e41c49584a84178424;rport=47858
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.100:5060 "Noisy feedback tells:  pid=32671 req_src_ip=213.235.220.98
req_src_port=47858 in_uri=sip:bob@ims3.ftw.at out_uri=sip:bob@ims3.ftw.at via_cnt==1"
```

```
     SIP MESSAGE 3        213.235.220.100:42962() -> 213.235.220.102:5060()
     TCP Frame 72       17/Jul/07 14:40:33.4069 TimeFromPreviousSipFrame=0.0001 TimeFromStart=12.5273
Extra Information: Packet was continued from Frame=71

INVITE sip:bob@ims3.ftw.at SIP/2.0
Record-Route: <sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Max-Forwards: 16
Contact: <sip:alice@213.235.220.98:5060>
Route: <sip:orig@scscf.ims3.ftw.at:5060;lr>
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Supported: 100rel,precondition,early-session
```

```
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Content-Type: application/sdp
Content-Length: 671
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
P-Charging-Vector: icid-value="P-CSCFabcd469cb8c100001198"; icid-generated-at="127.0.0.1"; orig-
ioi="ims3.ftw.at"

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
```

```
     SIP MESSAGE 4        213.235.220.102:5060() -> 213.235.220.100:42962()
     TCP Frame 85       17/Jul/07 14:40:33.4078 TimeFromPreviousSipFrame=0.0008 TimeFromStart=12.5281
SIP/2.0 100 trying -- your call is important to us
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.102:5060 "Noisy feedback tells:  pid=23395 req_src_ip=213.235.220.100
req_src_port=42962 in_uri=sip:bob@ims3.ftw.at out_uri=sip:bob@ims3.ftw.at via_cnt==2"
```

```
     SIP MESSAGE 5        213.235.220.102:37243() -> 213.235.220.100:5060()
     TCP Frame 108      17/Jul/07 14:40:33.4087 TimeFromPreviousSipFrame=0.0009 TimeFromStart=12.5290
Extra Information: Packet was continued from Frame=107

INVITE sip:bob@213.235.220.99:5060 SIP/2.0
Record-Route: <sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>
Route: <sip:term@p-cscf.ims3.ftw.at:5060;lr>
Record-Route: <sip:mo@scscf.ims3.ftw.at:5060;transport=tcp;lr>
Record-Route: <sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bKa34f.c897b743.0;i=aa
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bKa34f.b897b743.0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Max-Forwards: 14
Contact: <sip:alice@213.235.220.98:5060>
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Supported: 100rel,precondition,early-session
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Content-Type: application/sdp
Content-Length: 671
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
```

```
P-Charging-Vector: icid-value="P-CSCFabcd469cb8c100001198"; icid-generated-at="127.0.0.1"; orig-
ioi="ims3.ftw.at"
P-Called-Party-ID: <sip:bob@ims3.ftw.at>

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
```

```
     SIP MESSAGE 6        213.235.220.100:5060() -> 213.235.220.102:37243()
     TCP Frame 112      17/Jul/07 14:40:33.4097 TimeFromPreviousSipFrame=0.0010 TimeFromStart=12.5301
SIP/2.0 100 trying -- your call is important to us
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bKa34f.c897b743.0;i=aa;rport=37243
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bKa34f.b897b743.0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Server: Sip EXpress router (2.1.0-dev1 OpenIMSCore (i386/linux))
Content-Length: 0
Warning: 392 213.235.220.100:5060 "Noisy feedback tells:  pid=32672 req_src_ip=213.235.220.102
req_src_port=37243 in_uri=sip:bob@213.235.220.99:5060 out_uri=sip:bob@213.235.220.99:5060 via_cnt==4"
```

```
     SIP MESSAGE 7        213.235.220.100:5060() -> 213.235.220.99:35791()
     TCP Frame 114      17/Jul/07 14:40:33.4099 TimeFromPreviousSipFrame=0.0001 TimeFromStart=12.5302
Extra Information: Packet was continued from Frame=113

INVITE sip:bob@213.235.220.99:5060 SIP/2.0
Record-Route: <sip:mt@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Record-Route: <sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>
Record-Route: <sip:mo@scscf.ims3.ftw.at:5060;transport=tcp;lr>
Record-Route: <sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa34f.fa21eeb2.0;i=5a01
Via: SIP/2.0/TCP 213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bKa34f.c897b743.0;i=aa
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bKa34f.b897b743.0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Max-Forwards: 13
Contact: <sip:alice@213.235.220.98:5060>
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Supported: 100rel,precondition,early-session
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Content-Type: application/sdp
Content-Length: 671
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
P-Charging-Vector: icid-value="P-CSCFabcd469cb8c100001198"; icid-generated-at="127.0.0.1"; orig-
```

```
ioi="ims3.ftw.at"
P-Called-Party-ID: <sip:bob@ims3.ftw.at>

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
```

```
     SIP MESSAGE 8        213.235.220.99:35791() -> 213.235.220.100:5060()
     TCP Frame 122      17/Jul/07 14:40:33.4354 TimeFromPreviousSipFrame=0.0255 TimeFromStart=12.5557
Extra Information: Packet was continued from Frame=121

SIP/2.0 183 Session progress
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa34f.fa21eeb2.0;i=5a01;rport=5060,SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bKa34f.c897b743.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bKa34f.b897b743.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Record-Route: <sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:bob@213.235.220.99:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Require: 100rel, precondition
RSeq: 502978345
Content-Type: application/sdp
Content-Length: 734

v=0
o=bob 0 0 IN IP4 213.235.220.99
s=-
c=IN IP4 213.235.220.99
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=recvonly
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
m=video 22222 RTP/AVP 34 26 31
```

```
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
```

```
     SIP MESSAGE 9        213.235.220.100:5060() -> 213.235.220.102:37243()
     TCP Frame 125      17/Jul/07 14:40:33.4360 TimeFromPreviousSipFrame=0.0006 TimeFromStart=12.5563
Extra Information: Packet was continued from Frame=124

SIP/2.0 183 Session progress
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bKa34f.c897b743.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bKa34f.b897b743.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Record-Route: <sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:bob@213.235.220.99:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Require: 100rel, precondition
RSeq: 502978345
Content-Type: application/sdp
Content-Length: 734
P-Asserted-Identity: <sip:bob@ims3.ftw.at>

v=0
o=bob 0 0 IN IP4 213.235.220.99
s=-
c=IN IP4 213.235.220.99
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=recvonly
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
m=video 22222 RTP/AVP 34 26 31
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
```

```
     SIP MESSAGE 10       213.235.220.102:5060() -> 213.235.220.100:42962()
     TCP Frame 136      17/Jul/07 14:40:33.4369 TimeFromPreviousSipFrame=0.0009 TimeFromStart=12.5572
Extra Information: Packet was continued from Frame=134

SIP/2.0 183 Session progress
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
```

```
Record-Route: <sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:bob@213.235.220.99:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Require: 100rel, precondition
RSeq: 502978345
Content-Type: application/sdp
Content-Length: 734
P-Asserted-Identity: <sip:bob@ims3.ftw.at>

v=0
o=bob 0 0 IN IP4 213.235.220.99
s=-
c=IN IP4 213.235.220.99
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=recvonly
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
m=video 22222 RTP/AVP 34 26 31
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
```

```
     SIP MESSAGE 11       213.235.220.100:5060() -> 213.235.220.98:47858()
     TCP Frame 138      17/Jul/07 14:40:33.4374 TimeFromPreviousSipFrame=0.0006 TimeFromStart=12.5578
Extra Information: Packet was continued from Frame=137

SIP/2.0 183 Session progress
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Record-Route: <sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:bob@213.235.220.99:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Require: 100rel, precondition
RSeq: 502978345
Content-Type: application/sdp
Content-Length: 734
P-Asserted-Identity: <sip:bob@ims3.ftw.at>

v=0
o=bob 0 0 IN IP4 213.235.220.99
s=-
c=IN IP4 213.235.220.99
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=recvonly
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
```

```
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
m=video 22222 RTP/AVP 34 26 31
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
```

```
     SIP MESSAGE 12       213.235.220.98:47858() -> 213.235.220.100:5060()
     TCP Frame 142      17/Jul/07 14:40:33.4625 TimeFromPreviousSipFrame=0.0251 TimeFromStart=12.5829
Extra Information: Packet was continued from Frame=141

PRACK sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 2 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.98:5060;branch=z9hG4bK9db7ac72a4c75eacd9e41632d22167bc
Max-Forwards: 70
Route: <sip:mo@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mt@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:alice@213.235.220.98:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
RAck: 502978345 1 INVITE
Content-Type: application/sdp
Require: precondition,sec-agree
Proxy-Require: sec-agree
Content-Length: 681

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

```
     SIP MESSAGE 13       213.235.220.100:42962() -> 213.235.220.102:5060()
     TCP Frame 145      17/Jul/07 14:40:33.4634 TimeFromPreviousSipFrame=0.0009 TimeFromStart=12.5838
Extra Information: Packet was continued from Frame=144

PRACK sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 2 PRACK
```

```
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK734f.3c0af29.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK9db7ac72a4c75eacd9e41632d22167bc
Max-Forwards: 16
Route:
<sip:mo@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:alice@213.235.220.98:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
RAck: 502978345 1 INVITE
Content-Type: application/sdp
Require: precondition,sec-agree
Proxy-Require: sec-agree
Content-Length: 681
P-Asserted-Identity: <sip:alice@ims3.ftw.at>

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

```
     SIP MESSAGE 14       213.235.220.102:37243() -> 213.235.220.100:5060()
     TCP Frame 152      17/Jul/07 14:40:33.4643 TimeFromPreviousSipFrame=0.0009 TimeFromStart=12.5847
Extra Information: Packet was continued from Frame=151

PRACK sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 2 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK734f.c871e1f2.0;i=aa
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK734f.b871e1f2.0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK734f.3c0af29.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK9db7ac72a4c75eacd9e41632d22167bc
Max-Forwards: 14
Route: <sip:mt@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:alice@213.235.220.98:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
RAck: 502978345 1 INVITE
Content-Type: application/sdp
Require: precondition,sec-agree
Proxy-Require: sec-agree
Content-Length: 681
P-Asserted-Identity: <sip:alice@ims3.ftw.at>

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
```

```
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

```
     SIP MESSAGE 15       213.235.220.100:5060() -> 213.235.220.99:35791()
     TCP Frame 155      17/Jul/07 14:40:33.4650 TimeFromPreviousSipFrame=0.0007 TimeFromStart=12.5854
Extra Information: Packet was continued from Frame=154

PRACK sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 2 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK734f.4c0af29.0;i=5a01
Via: SIP/2.0/TCP 213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bK734f.c871e1f2.0;i=aa
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK734f.b871e1f2.0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK734f.3c0af29.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK9db7ac72a4c75eacd9e41632d22167bc
Max-Forwards: 13
Contact: <sip:alice@213.235.220.98:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
RAck: 502978345 1 INVITE
Content-Type: application/sdp
Require: precondition,sec-agree
Proxy-Require: sec-agree
Content-Length: 681
P-Asserted-Identity: <sip:alice@ims3.ftw.at>

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

```
     SIP MESSAGE 16       213.235.220.99:35791() -> 213.235.220.100:5060()
     TCP Frame 158      17/Jul/07 14:40:33.4740 TimeFromPreviousSipFrame=0.0089 TimeFromStart=12.5943
Extra Information: Packet was continued from Frame=157

SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 2 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK734f.4c0af29.0;i=5a01;rport=5060,SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bK734f.c871e1f2.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bK734f.b871e1f2.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bK734f.3c0af29.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK9db7ac72a4c75eacd9e41632d22167bc
P-Access-Network-Info: IEEE-802.11
Require: precondition
Content-Type: application/sdp
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 734

v=0
o=bob 0 0 IN IP4 213.235.220.99
s=-
c=IN IP4 213.235.220.99
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=recvonly
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
m=video 22222 RTP/AVP 34 26 31
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
```

```
     SIP MESSAGE 17       213.235.220.100:5060() -> 213.235.220.102:37243()
     TCP Frame 160      17/Jul/07 14:40:33.4745 TimeFromPreviousSipFrame=0.0005 TimeFromStart=12.5949
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 2 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bK734f.c871e1f2.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bK734f.b871e1f2.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bK734f.3c0af29.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK9db7ac72a4c75eacd9e41632d22167bc
P-Access-Network-Info: IEEE-802.11
Require: precondition
Content-Type: application/sdp
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 734

v=0
o=bob 0 0 IN IP4 213.235.220.99
s=-
c=IN IP4 213.235.220.99
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=recvonly
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
```

```
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
m=video 22222 RTP/AVP 34 26 31
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
```

```
     SIP MESSAGE 18       213.235.220.102:5060() -> 213.235.220.100:42962()
     TCP Frame 166      17/Jul/07 14:40:33.4753 TimeFromPreviousSipFrame=0.0008 TimeFromStart=12.5957
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 2 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK734f.3c0af29.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK9db7ac72a4c75eacd9e41632d22167bc
P-Access-Network-Info: IEEE-802.11
Require: precondition
Content-Type: application/sdp
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 734

v=0
o=bob 0 0 IN IP4 213.235.220.99
s=-
c=IN IP4 213.235.220.99
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=recvonly
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
m=video 22222 RTP/AVP 34 26 31
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
```

```
     SIP MESSAGE 19       213.235.220.100:5060() -> 213.235.220.98:47858()
     TCP Frame 167      17/Jul/07 14:40:33.4759 TimeFromPreviousSipFrame=0.0005 TimeFromStart=12.5962
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 2 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK9db7ac72a4c75eacd9e41632d22167bc
P-Access-Network-Info: IEEE-802.11
Require: precondition
Content-Type: application/sdp
```

```
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 734

v=0
o=bob 0 0 IN IP4 213.235.220.99
s=-
c=IN IP4 213.235.220.99
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=recvonly
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
m=video 22222 RTP/AVP 34 26 31
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote sendrecv
```

```
     SIP MESSAGE 20       213.235.220.98:47858() -> 213.235.220.100:5060()
     TCP Frame 169      17/Jul/07 14:40:33.4870 TimeFromPreviousSipFrame=0.0112 TimeFromStart=12.6074
Extra Information: Packet was continued from Frame=168

UPDATE sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 3 UPDATE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.98:5060;branch=z9hG4bKfa6ee0be36a78cb7661af44f86b48e09
Max-Forwards: 70
Route: <sip:mo@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mt@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:alice@213.235.220.98:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Require: sec-agree
Proxy-Require: sec-agree
Content-Type: application/sdp
Content-Length: 689

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
```

```
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

```
     SIP MESSAGE 21       213.235.220.100:42962() -> 213.235.220.102:5060()
     TCP Frame 172      17/Jul/07 14:40:33.4880 TimeFromPreviousSipFrame=0.0009 TimeFromStart=12.6083
Extra Information: Packet was continued from Frame=171

UPDATE sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 3 UPDATE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK834f.687b803.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKfa6ee0be36a78cb7661af44f86b48e09
Max-Forwards: 16
Route:
<sip:mo@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:alice@213.235.220.98:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Require: sec-agree
Proxy-Require: sec-agree
Content-Type: application/sdp
Content-Length: 689
P-Asserted-Identity: <sip:alice@ims3.ftw.at>

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

```
     SIP MESSAGE 22       213.235.220.102:37243() -> 213.235.220.100:5060()
     TCP Frame 179      17/Jul/07 14:40:33.4889 TimeFromPreviousSipFrame=0.0009 TimeFromStart=12.6092
Extra Information: Packet was continued from Frame=178

UPDATE sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 3 UPDATE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK834f.c9c12ce3.0;i=aa
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK834f.b9c12ce3.0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK834f.687b803.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKfa6ee0be36a78cb7661af44f86b48e09
Max-Forwards: 14
Route: <sip:mt@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:alice@213.235.220.98:5060>
```

```
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Require: sec-agree
Proxy-Require: sec-agree
Content-Type: application/sdp
Content-Length: 689
P-Asserted-Identity: <sip:alice@ims3.ftw.at>

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

```
     SIP MESSAGE 23       213.235.220.100:5060() -> 213.235.220.99:35791()
     TCP Frame 182      17/Jul/07 14:40:33.4897 TimeFromPreviousSipFrame=0.0009 TimeFromStart=12.6101
Extra Information: Packet was continued from Frame=181

UPDATE sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 3 UPDATE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK834f.787b803.0;i=5a01
Via: SIP/2.0/TCP 213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bK834f.c9c12ce3.0;i=aa
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK834f.b9c12ce3.0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK834f.687b803.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKfa6ee0be36a78cb7661af44f86b48e09
Max-Forwards: 13
Contact: <sip:alice@213.235.220.98:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Require: sec-agree
Proxy-Require: sec-agree
Content-Type: application/sdp
Content-Length: 689
P-Asserted-Identity: <sip:alice@ims3.ftw.at>

v=0
o=alice 0 0 IN IP4 213.235.220.98
s=-
c=IN IP4 213.235.220.98
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=sendrecv
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
```

```
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 26 34 31
b=AS:100
a=sendrecv
a=rtpmap:26 jpeg/90000
a=rtpmap:34 h263/90000
a=rtpmap:31 h261/90000
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

```
     SIP MESSAGE 24       213.235.220.99:35791() -> 213.235.220.100:5060()
     TCP Frame 184      17/Jul/07 14:40:33.4986 TimeFromPreviousSipFrame=0.0088 TimeFromStart=12.6189
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 3 UPDATE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK834f.787b803.0;i=5a01;rport=5060,SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bK834f.c9c12ce3.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bK834f.b9c12ce3.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bK834f.687b803.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKfa6ee0be36a78cb7661af44f86b48e09
P-Access-Network-Info: IEEE-802.11
Content-Type: application/sdp
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 694

v=0
o=bob 0 0 IN IP4 213.235.220.99
s=-
c=IN IP4 213.235.220.99
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=recvonly
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 34 26 31
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

```
     SIP MESSAGE 25       213.235.220.100:5060() -> 213.235.220.102:37243()
     TCP Frame 185      17/Jul/07 14:40:33.4991 TimeFromPreviousSipFrame=0.0005 TimeFromStart=12.6195
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 3 UPDATE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bK834f.c9c12ce3.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bK834f.b9c12ce3.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bK834f.687b803.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKfa6ee0be36a78cb7661af44f86b48e09
P-Access-Network-Info: IEEE-802.11
Content-Type: application/sdp
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 694

v=0
o=bob 0 0 IN IP4 213.235.220.99
```

```
s=-
c=IN IP4 213.235.220.99
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=recvonly
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 34 26 31
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

```
     SIP MESSAGE 26       213.235.220.102:5060() -> 213.235.220.100:42962()
     TCP Frame 192      17/Jul/07 14:40:33.4999 TimeFromPreviousSipFrame=0.0008 TimeFromStart=12.6202
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 3 UPDATE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK834f.687b803.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKfa6ee0be36a78cb7661af44f86b48e09
P-Access-Network-Info: IEEE-802.11
Content-Type: application/sdp
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 694

v=0
o=bob 0 0 IN IP4 213.235.220.99
s=-
c=IN IP4 213.235.220.99
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=recvonly
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 34 26 31
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

```
     SIP MESSAGE 27       213.235.220.100:5060() -> 213.235.220.98:47858()
     TCP Frame 193      17/Jul/07 14:40:33.5005 TimeFromPreviousSipFrame=0.0006 TimeFromStart=12.6209
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 3 UPDATE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
```

```
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKfa6ee0be36a78cb7661af44f86b48e09
P-Access-Network-Info: IEEE-802.11
Content-Type: application/sdp
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 694

v=0
o=bob 0 0 IN IP4 213.235.220.99
s=-
c=IN IP4 213.235.220.99
t=0 0
m=audio 22224 RTP/AVP 0 3 4 5 6 8 15 18
b=AS:25
a=recvonly
a=rtpmap:0 ULAW/8000
a=rtpmap:3 gsm/8000
a=rtpmap:4 g723/8000
a=rtpmap:5 dvi/8000
a=rtpmap:6 dvi/16000
a=rtpmap:8 alaw/8000
a=rtpmap:15 g728/8000
a=rtpmap:18 g729/8000
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 22222 RTP/AVP 34 26 31
b=AS:75
a=sendrecv
a=rtpmap:34 h263/90000
a=rtpmap:26 jpeg/90000
a=rtpmap:31 h261/90000
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

```
     SIP MESSAGE 28       213.235.220.99:35791() -> 213.235.220.100:5060()
     TCP Frame 195      17/Jul/07 14:40:33.5391 TimeFromPreviousSipFrame=0.0386 TimeFromStart=12.6595
SIP/2.0 180 Ringing
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa34f.fa21eeb2.0;i=5a01;rport=5060,SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bKa34f.c897b743.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bKa34f.b897b743.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Record-Route: <sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:bob@213.235.220.99:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Require: 100rel
RSeq: 502978346
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Content-Length: 0
```

```
     SIP MESSAGE 29       213.235.220.100:5060() -> 213.235.220.102:37243()
     TCP Frame 199      17/Jul/07 14:40:33.5398 TimeFromPreviousSipFrame=0.0007 TimeFromStart=12.6601
SIP/2.0 180 Ringing
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bKa34f.c897b743.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bKa34f.b897b743.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Record-Route: <sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:bob@213.235.220.99:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Require: 100rel
RSeq: 502978346
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
```

```
Content-Length: 0
P-Asserted-Identity: <sip:bob@ims3.ftw.at>
```

```
     SIP MESSAGE 30       213.235.220.102:5060() -> 213.235.220.100:42962()
     TCP Frame 206      17/Jul/07 14:40:33.5404 TimeFromPreviousSipFrame=0.0007 TimeFromStart=12.6608
SIP/2.0 180 Ringing
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Record-Route: <sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:bob@213.235.220.99:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Require: 100rel
RSeq: 502978346
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Content-Length: 0
P-Asserted-Identity: <sip:bob@ims3.ftw.at>
```

```
     SIP MESSAGE 31       213.235.220.100:5060() -> 213.235.220.98:47858()
     TCP Frame 208      17/Jul/07 14:40:33.5409 TimeFromPreviousSipFrame=0.0005 TimeFromStart=12.6613
SIP/2.0 180 Ringing
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
Record-Route: <sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:bob@213.235.220.99:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
Require: 100rel
RSeq: 502978346
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Content-Length: 0
P-Asserted-Identity: <sip:bob@ims3.ftw.at>
```

```
     SIP MESSAGE 32       213.235.220.98:47858() -> 213.235.220.100:5060()
     TCP Frame 210      17/Jul/07 14:40:33.5512 TimeFromPreviousSipFrame=0.0103 TimeFromStart=12.6716
PRACK sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 4 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.98:5060;branch=z9hG4bKddcb59a7f81feb94b72a1a1f81b7d20b
Max-Forwards: 70
Route: <sip:mo@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mt@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:alice@213.235.220.98:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
RAck: 502978346 1 INVITE
Require: sec-agree
Proxy-Require: sec-agree
Content-Length: 0
```

```
     SIP MESSAGE 33       213.235.220.100:42962() -> 213.235.220.102:5060()
     TCP Frame 211      17/Jul/07 14:40:33.5520 TimeFromPreviousSipFrame=0.0007 TimeFromStart=12.6723
PRACK sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 4 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK534f.eee47092.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKddcb59a7f81feb94b72a1a1f81b7d20b
Max-Forwards: 16
Route:
<sip:mo@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@p-
```

```
cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:alice@213.235.220.98:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
RAck: 502978346 1 INVITE
Require: sec-agree
Proxy-Require: sec-agree
Content-Length: 0
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
```

```
     SIP MESSAGE 34       213.235.220.102:37243() -> 213.235.220.100:5060()
     TCP Frame 216      17/Jul/07 14:40:33.5526 TimeFromPreviousSipFrame=0.0007 TimeFromStart=12.6730
PRACK sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 4 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK534f.96754d06.0;i=aa
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK534f.86754d06.0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK534f.eee47092.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKddcb59a7f81feb94b72a1a1f81b7d20b
Max-Forwards: 14
Route: <sip:mt@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Contact: <sip:alice@213.235.220.98:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
RAck: 502978346 1 INVITE
Require: sec-agree
Proxy-Require: sec-agree
Content-Length: 0
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
```

```
     SIP MESSAGE 35       213.235.220.100:5060() -> 213.235.220.99:35791()
     TCP Frame 217      17/Jul/07 14:40:33.5533 TimeFromPreviousSipFrame=0.0007 TimeFromStart=12.6737
PRACK sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 4 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK534f.fee47092.0;i=5a01
Via: SIP/2.0/TCP 213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bK534f.96754d06.0;i=aa
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK534f.86754d06.0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK534f.eee47092.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKddcb59a7f81feb94b72a1a1f81b7d20b
Max-Forwards: 13
Contact: <sip:alice@213.235.220.98:5060>
P-Access-Network-Info: IEEE-802.11
User-Agent: IMS-Communicator 070605
RAck: 502978346 1 INVITE
Require: sec-agree
Proxy-Require: sec-agree
Content-Length: 0
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
```

```
     SIP MESSAGE 36       213.235.220.99:35791() -> 213.235.220.100:5060()
     TCP Frame 218      17/Jul/07 14:40:33.5594 TimeFromPreviousSipFrame=0.0061 TimeFromStart=12.6798
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 4 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK534f.fee47092.0;i=5a01;rport=5060,SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bK534f.96754d06.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bK534f.86754d06.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bK534f.eee47092.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKddcb59a7f81feb94b72a1a1f81b7d20b
P-Access-Network-Info: IEEE-802.11
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 0
```

```
     SIP MESSAGE 37       213.235.220.100:5060() -> 213.235.220.102:37243()
     TCP Frame 219      17/Jul/07 14:40:33.5599 TimeFromPreviousSipFrame=0.0005 TimeFromStart=12.6802
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 4 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
```

```
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bK534f.96754d06.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bK534f.86754d06.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bK534f.eee47092.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKddcb59a7f81feb94b72a1a1f81b7d20b
P-Access-Network-Info: IEEE-802.11
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 0
```

```
     SIP MESSAGE 38       213.235.220.102:5060() -> 213.235.220.100:42962()
     TCP Frame 224      17/Jul/07 14:40:33.5604 TimeFromPreviousSipFrame=0.0005 TimeFromStart=12.6808
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 4 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK534f.eee47092.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKddcb59a7f81feb94b72a1a1f81b7d20b
P-Access-Network-Info: IEEE-802.11
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 0
```

```
     SIP MESSAGE 39       213.235.220.100:5060() -> 213.235.220.98:47858()
     TCP Frame 226      17/Jul/07 14:40:33.5610 TimeFromPreviousSipFrame=0.0006 TimeFromStart=12.6813
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 4 PRACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKddcb59a7f81feb94b72a1a1f81b7d20b
P-Access-Network-Info: IEEE-802.11
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 0
```

```
     SIP MESSAGE 40       213.235.220.99:35791() -> 213.235.220.100:5060()
     TCP Frame 273      17/Jul/07 14:40:40.2155 TimeFromPreviousSipFrame=6.6545 TimeFromStart=19.3358
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa34f.fa21eeb2.0;i=5a01;rport=5060,SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bKa34f.c897b743.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bKa34f.b897b743.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
User-Agent: IMS-Communicator 070605
Record-Route: <sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 0
```

```
     SIP MESSAGE 41       213.235.220.100:5060() -> 213.235.220.102:37243()
     TCP Frame 275      17/Jul/07 14:40:40.2162 TimeFromPreviousSipFrame=0.0007 TimeFromStart=19.3365
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bKa34f.c897b743.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bKa34f.b897b743.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
User-Agent: IMS-Communicator 070605
Record-Route: <sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 0
P-Asserted-Identity: <sip:bob@ims3.ftw.at>
```

```
     SIP MESSAGE 42       213.235.220.102:5060() -> 213.235.220.100:42962()
     TCP Frame 283      17/Jul/07 14:40:40.2170 TimeFromPreviousSipFrame=0.0008 TimeFromStart=19.3373
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bKa34f.ea21eeb2.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
User-Agent: IMS-Communicator 070605
Record-Route: <sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 0
P-Asserted-Identity: <sip:bob@ims3.ftw.at>
```

```
     SIP MESSAGE 43       213.235.220.100:5060() -> 213.235.220.98:47858()
     TCP Frame 287      17/Jul/07 14:40:40.2176 TimeFromPreviousSipFrame=0.0007 TimeFromStart=19.3380
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 INVITE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bK8e0920833a7217e41c49584a84178424
User-Agent: IMS-Communicator 070605
Record-Route: <sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mo@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Allow: INVITE,ACK,CANCEL,BYE,MESSAGE,PRACK,UPDATE
Contact: <sip:bob@213.235.220.99:5060>
Content-Length: 0
P-Asserted-Identity: <sip:bob@ims3.ftw.at>
```

```
     SIP MESSAGE 44       213.235.220.98:47858() -> 213.235.220.100:5060()
     TCP Frame 289      17/Jul/07 14:40:40.2229 TimeFromPreviousSipFrame=0.0053 TimeFromStart=19.3432
ACK sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 ACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.98:5060;branch=z9hG4bKda426199
Max-Forwards: 70
Route: <sip:mo@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mt@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Content-Length: 0
```

```
     SIP MESSAGE 45       213.235.220.100:42962() -> 213.235.220.102:5060()
     TCP Frame 290      17/Jul/07 14:40:40.2235 TimeFromPreviousSipFrame=0.0006 TimeFromStart=19.3439
ACK sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 ACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=0;i=f901
Via: SIP/2.0/TCP 213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKda426199
Max-Forwards: 16
Route:
<sip:mo@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>
Content-Length: 0
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
```

```
     SIP MESSAGE 46       213.235.220.102:37243() -> 213.235.220.100:5060()
     TCP Frame 295      17/Jul/07 14:40:40.2241 TimeFromPreviousSipFrame=0.0006 TimeFromStart=19.3444
ACK sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 ACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.102;branch=0;i=aa
Via: SIP/2.0/TCP 213.235.220.102;branch=0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=0;i=f901
Via: SIP/2.0/TCP 213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKda426199
Max-Forwards: 14
```

```
Route: <sip:mt@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Content-Length: 0
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
```

```
     SIP MESSAGE 47       213.235.220.100:5060() -> 213.235.220.99:35791()
     TCP Frame 296      17/Jul/07 14:40:40.2246 TimeFromPreviousSipFrame=0.0005 TimeFromStart=19.3450
ACK sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 1 ACK
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=0;i=5a01
Via: SIP/2.0/TCP 213.235.220.102;received=213.235.220.102;rport=37243;branch=0;i=aa
Via: SIP/2.0/TCP 213.235.220.102;branch=0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=0;i=f901
Via: SIP/2.0/TCP 213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKda426199
Max-Forwards: 13
Content-Length: 0
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
```

```
     SIP MESSAGE 48       213.235.220.98:47858() -> 213.235.220.100:5060()
     TCP Frame 373      17/Jul/07 14:40:51.7925 TimeFromPreviousSipFrame=11.5679 TimeFromStart=30.9129
BYE sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 5 BYE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.98:5060;branch=z9hG4bKb347e268975e7433aee33c93f2adac42
Max-Forwards: 70
User-Agent: IMS-Communicator 070605
Route: <sip:mo@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mo@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ft
w.at:5060;transport=tcp;lr>,<sip:mt@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Content-Length: 0
```

```
     SIP MESSAGE 49       213.235.220.100:42962() -> 213.235.220.102:5060()
     TCP Frame 375      17/Jul/07 14:40:51.7933 TimeFromPreviousSipFrame=0.0008 TimeFromStart=30.9136
BYE sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 5 BYE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK634f.dca68635.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKb347e268975e7433aee33c93f2adac42
Max-Forwards: 16
User-Agent: IMS-Communicator 070605
Route:
<sip:mo@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@scscf.ims3.ftw.at:5060;transport=tcp;lr>,<sip:mt@p-
cscf.ims3.ftw.at:5060;transport=tcp;lr>
Content-Length: 0
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
```

```
     SIP MESSAGE 50       213.235.220.102:37243() -> 213.235.220.100:5060()
     TCP Frame 381      17/Jul/07 14:40:51.7941 TimeFromPreviousSipFrame=0.0008 TimeFromStart=30.9144
BYE sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 5 BYE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK634f.6c1c3c81.0;i=aa
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK634f.5c1c3c81.0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK634f.dca68635.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKb347e268975e7433aee33c93f2adac42
Max-Forwards: 14
User-Agent: IMS-Communicator 070605
Route: <sip:mt@p-cscf.ims3.ftw.at:5060;transport=tcp;lr>
Content-Length: 0
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
```

```
     SIP MESSAGE 51       213.235.220.100:5060() -> 213.235.220.99:35791()
     TCP Frame 383      17/Jul/07 14:40:51.7947 TimeFromPreviousSipFrame=0.0006 TimeFromStart=30.9151
BYE sip:bob@213.235.220.99:5060 SIP/2.0
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 5 BYE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
```

```
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK634f.eca68635.0;i=5a01
Via: SIP/2.0/TCP 213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bK634f.6c1c3c81.0;i=aa
Via: SIP/2.0/TCP 213.235.220.102;branch=z9hG4bK634f.5c1c3c81.0;i=8a
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK634f.dca68635.0;i=f901
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKb347e268975e7433aee33c93f2adac42
Max-Forwards: 13
User-Agent: IMS-Communicator 070605
Content-Length: 0
P-Asserted-Identity: <sip:alice@ims3.ftw.at>
```

```
     SIP MESSAGE 52       213.235.220.99:35791() -> 213.235.220.100:5060()
     TCP Frame 385      17/Jul/07 14:40:52.7989 TimeFromPreviousSipFrame=1.0042 TimeFromStart=31.9192
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 5 BYE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK634f.eca68635.0;i=5a01;rport=5060,SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bK634f.6c1c3c81.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bK634f.5c1c3c81.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bK634f.dca68635.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKb347e268975e7433aee33c93f2adac42
Content-Length: 0
```

```
     SIP MESSAGE 53       213.235.220.100:5060() -> 213.235.220.102:37243()
     TCP Frame 386      17/Jul/07 14:40:52.8001 TimeFromPreviousSipFrame=0.0012 TimeFromStart=31.9204
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 5 BYE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.102;received=213.235.220.102;rport=37243;branch=z9hG4bK634f.6c1c3c81.0;i=aa,SIP/2.0/TCP
213.235.220.102;branch=z9hG4bK634f.5c1c3c81.0;i=8a,SIP/2.0/TCP
213.235.220.100;branch=z9hG4bK634f.dca68635.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKb347e268975e7433aee33c93f2adac42
Content-Length: 0
```

```
     SIP MESSAGE 54       213.235.220.102:5060() -> 213.235.220.100:42962()
     TCP Frame 395      17/Jul/07 14:40:52.8007 TimeFromPreviousSipFrame=0.0007 TimeFromStart=31.9211
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 5 BYE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP 213.235.220.100;branch=z9hG4bK634f.dca68635.0;i=f901,SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKb347e268975e7433aee33c93f2adac42
Content-Length: 0
```

```
     SIP MESSAGE 55       213.235.220.100:5060() -> 213.235.220.98:47858()
     TCP Frame 396      17/Jul/07 14:40:52.8012 TimeFromPreviousSipFrame=0.0005 TimeFromStart=31.9216
SIP/2.0 200 OK
Call-ID: 537cbd2556e62ed3651a2d2982bb7d7c@213.235.220.98
CSeq: 5 BYE
From: "alice" <sip:alice@ims3.ftw.at>;tag=32820206
To: <sip:bob@ims3.ftw.at>;tag=11697208
Via: SIP/2.0/TCP
213.235.220.98:5060;received=213.235.220.98;rport=47858;branch=z9hG4bKb347e268975e7433aee33c93f2adac42
Content-Length: 0
```

Abbreviations

3G	3rd Generation
3GPP	3rd Generation Partnership Project
3GPP TR	3GPP Technical Report
3GPP TS	3GPP Technical Specification
AAA	Authentication, Authorization and Accounting
AF	Application Function
AKA	Authorization Key Agreement
AMR	Adaptive Multi-Rate
AMR-NB	Adaptive Multi-Rate - NarrowBand
AMR-WB	Adaptive Multi-Rate - WideBand
APN	Access Point Name
AS	Application Server
AS	Application Specific (SDP Attribute modifier)
ATM	Asynchronous Transfer Mode
AVC	Advanced Video Coding
AVP	Audio-Visual Profile
AVPF	Audio-Visual Profile with Feedback
BGCF	Breakout Gateway Control Function
BLER	Block Error Rate
BS	Bearer Service
CAC	Connection Admission Control
CCM	Codec Control Messages
CIF	Common Intermediate Format
CK	Cipher Key
CN (IM CN)	Core Network (IP Multimedia Core Network)
COPS	Common Open Policy Service
CS	Circuit Switched
CSCF	Call Session Control Function
CTM	Cellular Text telephone Modem
DCCP	Datagram Congestion Control Protocol
DCT	Discrete Cosine Transform

DHCP	Dynamic Host Configuration Protocol
DiffServ	Differentiated Services
DNS	Domain Name System
DOCSIS	Data Over Cable Service Interface Specification
DTD	Document Type Definition
DTX	Discontinuous Transmission
Eo/No	Energy (Per Bit) to Noise Ratio
ETSI	European Telecommunications Standards Institute
FDDI	Fiber Distributed Data Interface
FER	Frame Error Rate
FHoSS	FOKUS HSS
GGSN	Gateway GPRS Support Node
GPRS	General Packet Radio Service
GUI	Graphical User Interface
HLR	Home Location Register
HSS	Home Subscriber Server
HTTP	HyperText Transfer Protocol
I-CSCF	Interrogating CSCF
IETF	Internet Engineering Task Force
IK	Integrity Key
IKE	Internet Key Management
IMS	IP Multimedia Subsystem
IntServ	Integrated Services
IP	Internet Protocol
IP-CAN	IP-Connectivity Access Network
IPsec	IP security
IPv4	Internet Protocol version 4
IPv6	Internet Protocol version 6
ISIM	IM Subscriber Identity Module
ISO/IEC	International Organization for Standardization / International Electrotechnical Commission
ITU-T	International Telecommunication Union (ITU) Telecommunication Standardization Sector
ITU-T VCEG	ITU-T Video Coding Experts Group
KPI	Key Performance Indicators
LIA	Location-Information-Answer
LIR	Location-Information-Request
LS	lip Synchronization
MD5	Message-Digest Algorithm 5
MGCF	Media Gateway Control Function

MGW	Media Gateway
MIME	Multipurpose Internet Mail Extensions
MPEG	Moving Picture Experts Group
MT	Multimedia Terminal
MTLS	Multicast Transport Layer Security
MTSI	Multimedia Telephony Service for IMS
MTU	Maximum Transfer Unit
NAT	Network Address Translation
NDS	Network Domain Security
NSAPI	Network Sublayer Access Point Id
NTP	Network Time Protocol
OBMC	Overlapped Block Motion Compensation
OSIMS	Open Source IMS (Fraunhofer Institute FOKUS)
P-CSCF	Proxy CSCF
PDF	Policy Decision Function
PDG	Packet Data Gateway
PDP	Packet Data Protocol
PDP	Policy Decision Point
PDU	Protocol Data Unit
PEP	Policy Enforcement Point
PIB	Policy Information Base
PLI	Picture Loss Indication
PLMN	Public Land Mobile Network
PPP	Point to Point Protocol
PS	Packet Switched
PSTN	Public Switched Telephone Network
QCIF	Quarter Common Intermediate Format
QoE	Quality of Experience
QoS	Quality of Service
RAB	Radio Access Bearer
RAN	Radio Access Network
RB	Radio Bearer
RFC	Request For Comments
RR	Receiver Report
RR	RTCP Bandwidth for Receivers (SDP attribute modifier)
RS	RTCP Bandwidth for Senders (SDP attribute modifier)
RSCP	Received Signal Code Power.
RSVP	Resource ReSerVation Protocol
RTCP	Real-time Transport Control Protocol
RTP	Real-time Transport Protocol

RTSP	Real Time Streaming Protocol
SAP	Service Access Point
SAP	Session Announcement Protocol
SBLP	Service Based Local Policy
S-CSCF	Serving CSCF
SDP	Session Description Protocol
SDU	Service Data Unit
SER	SIP Express Router (Fraunhofer Institute FOKUS)
SGSN	Serving GPRS Support Node
SIP	Session Initiation Protocol
SLF	Subscriber Local Function
SMTP	Simple Mail Transfer Protocol
SNR	Signal to Noise Ratio
SOAP	Simple Object Access Protocol
SPI	security parameter index
SQCIF	Sub - Common Intermediate Format
SR	Sender Report
SRF	Single Reservation Flow
SS7	Signaling System 7
SSP	Service Switching Point
TDM	Time-Division Multiplexing
TE	Terminal Equipment
THIG	Topology Hiding Internetwork Gateway
TI	Transaction Identifier
TLS	Transport Layer Security
TMMBN	Temporary Maximum Media Bit-rate Notification
TMMBR	Temporary Maximum Media Bit-rate Request
UA	User Agent
UAC	User Agent Client
UAR	User Authorization Request
UAS	User Agent Server
UDP	User Datagram Protocol
UE	User Equipment
UMTS	Universal Mobile Telecommunications System
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
UVM	Unrestricted Motion Vector
VLC	Variable Length Codes
XML	eXtensible Markup Language

Bibliography

[1] 3GPP. *Technical Specification Group Services and System Aspects, Service aspects; Services and service capabilities*, release 8 edition. TS 22.105.

[2] H. Montes, G. Gomez, R. Cuny, and J. F. Paris. *Deployment of IP Multimedia Streaming Services in Third-Generation Mobile Networks*, October 2002.

[3] G. Camarillo and M.A. Garcia Martin. *The 3G IP Multimedia Subsystem; Merging the Internet and the Cellular worlds*, 2004.

[4] J. Fabini, I. Gojmerac, F. Hammer, O. Nemethova, and M. Ries. *Configuration, Architecture, Migration, Performance Analysis and Requirements of IMS; Deliverable D2.1: IP Multimedia Subsystem QoS.* N3 [CAMPARI]; State-of-the-Art Document.

[5] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. *SIP: Session Initiation Protocol*, June 2002. RFC 3261.

[6] J. Klensin. *Simple Mail Transfer Protocol*, April 2001. RFC 2821.

[7] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. *Hypertext Transfer Protocol – HTTP/1.1.* RFC 2616.

[8] M. Handley, V. Jacobson, and C. Perkins. *SDP: Session Description Protocol*, July 2006. RFC 4566.

[9] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. *RTP: A Transport Protocol for Real-Time Applications*, July 2003. RFC 3550.

[10] S. Casner. *Session Description Protocol (SDP) Bandwidth Modifiers for RTP Control Protocol (RTCP) Bandwidth*, July 2003. RFC 3556.

[11] 3GPP. *Technical Specification Group Services and Systems Aspects IP Multimedia Subsystem (IMS); Multimedia Telephony;Media handling and interaction*, March 2007. (Release 7), TS 26.114 v2.0.0.

[12] ITU-T. *Video coding for low bit rate communication*, January 2005. Recommendation H.263.

[13] R. Koenen. *MPEG-4 Overview*. MPEG-4 WG11. Version 21, ISO-IEC JTC1/SC29/WG.

[14] Thomas Wiegand, Gary J. Sullivan, Gisle Bjntegaard, and Ajay Luthra. *Overview of the H.264/AVC Video Coding Standard*, July 2003. IEEE Transactions on circuits and systems for video technology, VOL. 13, NO. 7.

[15] A. Munje, N. Naqvi, T. Plestid, K. Liang, J.P. Cormier, and M-UI. Hassan. *Methods and apparatus for efficiently establishing and maintaining a data connection between a mobile station and a wireless network*, 2006.

[16] R. Braden, L. Zhang, S. Berson, S. Herzog, and S. Jamin. *Resource ReSerVation Protocol (RSVP)*, September 1997. RFC 2205.

[17] 3GPP. *Technical Specification Group Services and Systems Aspects; Arquitectural enhancements for end-to-end Quality of Service (QoS)*. (Release 7), TR 23.802.

[18] 3GPP. *Technical Specification Group Services and Systems Aspects; IP Multimedia Subsystem (IMS); Stage 2*. (Release 8), TS 23.228.

[19] 3GPP. *Technical Specification Group Services and Systems Aspects; Quality of Service (QoS) concept and architecture*. (Release 6), TS 23.107.

[20] D. Soldani, Man Li, and R. Cuny. *QoS and QoE Management in UMTS Cellular Systems*.

[21] D. Soldani. *Introduction to QoS and QoE and service performance*. Special Course on Networking Technology; Helsinki University of Technology.

[22] 3GPP. *Technical Specification Group Core Network and Terminals; Signalling Flows for the session setup in the IP Multimedia core network Subsystem (IMS) based on the Session Initiation Protocol (SIP) And Session Description Protocol (SDP); Stage 3*. (Release 8), TR 24.930.

[23] G. Camarillo, Ed., W. Marshall, Ed., and J. Rosenberg. *Integration of Resource Management and Session Initiation Protocol (SIP)*, October 2002. RFC 3312.

[24] 3GPP. *General Packet Radio Service (GPRS); Service description; Stage 2*.

[25] R. Ferrús, A. Gelonch, F. Casadevall, J. Pérez, O. Sallent, R. Agustí, N. Nafisi, L. Wang, M. Dohler, H. Aghvami, and P. Karlsson. *End-to-End QoSArchitecture for Multi-Domain and Wireless Heterogeneous Access Networks: the EVEREST approach*, March 2004.

[26] R. Yavatkar, D. Pendarakis, and R. Guerin. *A Framework for Policy-based Admission Control*, January 2000. RFC 2753.

[27] J. Kim, S. Kim, B. Lee, and J. Choi. *Policy-based QoS Control for Open Services in BcN*, February 2005.

[28] W. Zhuang, Y. Sze Gan, K. Jeng Loh, and K. Chaing Chua. *Policy-Based QoS Architecture in the IP Multimedia Subsystem of UMTS*. IEEE Network, May/June 2003.

[29] 3GPP. *Technical Specification Group Core Network; Policy Control over Go interface*. (Release 5), TS 29.207.

[30] W. Marshall and Ed. *Private Session Initiation Protocol (SIP) Extensions for Media Authorization*, January 2003. RFC 3313.

[31] 3GPP. *Technical Specification Group Services and System Aspects; End-to-end Quality of Service (QoS) concept and architecture*. (Release 5), TS 23.207.

[32] 3GPP. *Technical Specification Group Core Network; Mobile radio interface layer 3 specification; Core Network Protocols - Stage 3*. (Release 1999), TS 24.008.

[33] T. Borosa, B. Marsić, and C. Pocuca. *QoS Support in IP Multimedia Subsystem Using DiffServ*, June 2003.

[34] Adi Paz. *IMS SIP: The Right Solution for Widespread Next Generation Networks*, February 2007.

[35] 3GPP. *Technical Specification Group Core Network and Terminals; IP multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3*. (Release 7), TS 24.229.

[36] Dorgham Sisalem. *Multimedia Communication in the Internet; SIP: Next Generation Networks*.

[37] 3GPP. *Technical Specification Group Core Network; End-to-end Quality of Service (QoS) signalling flows*. (Release 5), TS 29.208.

[38] L-N. Hamer, B. Gage, and H. Shieh. *Framework for Session Set-up with Media Authorization*, April 2003. RFC 3521.

[39] Computer Science at Columbia University. *SIP RFCs and Drafts*. http://www1.cs.columbia.edu/sip/drafts.html.

[40] J. Rosenberg. *The Session Initiation Protocol (SIP) UPDATE Method*, September 2002. RFC 3311.

[41] R. Price, C. Bormann, J. Christoffersson, H. Hannu, Z. Liu, and J. Rosenberg. *Signaling Compression (SigComp)*, January 2003. RFC 3320.

[42] M. Garcia-Martin, E. Henrikson, and D. Mills. *Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the 3rd-Generation Partnership Project (3GPP)*, January 2003. RFC 3455.

[43] C. Jennings, J. Peterson, and M. Watson. *Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks*, November 2002. RFC 3325.

[44] J. Arkko, V. Torvinen, G. Camarillo, A. Niemi, and T. Haukka. *Security Mechanism Agreement for the Session Initiation Protocol (SIP)*, January 2003. RFC 3329.

[45] A. Niemi, J. Arkko, and V. Torvinen. *Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA)*, September 2002. RFC 3310.

[46] S. Kent and R. Atkinson. *Security Architecture for the Internet Protocol*, November 1998. RFC 2401.

[47] D. Willis and B. Hoeneisen. *Session Initiation Protocol (SIP) Extension Header Field for Registering Non-Adjacent Contacts*, December 2002. RFC 3327.

[48] D. Willis and B. Hoeneisen. *Session Initiation Protocol (SIP) Extension Header Field for Service Route Discovery During Registration*, October 2003. RFC 3608.

[49] S. Deering and R. Hinden. *Internet Protocol, Version 6 (IPv6) Specification*, December 1998. RFC 2460.

[50] J. Ott, S. Wenger, N. Sato, C. Burmeister, and J. Rey. *Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)*, July 2006. RFC 4585.

[51] G. Camarillo, J. Holler, and H. Schulzrinne. *Grouping of Media Lines in the Session Description Protocol (SDP)*, December 2002. RFC 3388.

[52] Sriram Parameswar and Brian Stucker. *The SIP Negotiate Method*, August 2001. <draft-spbs-sip-negotiate-00.txt>.

[53] J. Rosenberg, H. Schulzrinne, and P. Kyzivat. *Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)*, August 2004. RFC 3840.

[54] J. Rosenberg, H. Schulzrinne, and P. Kyzivat. *Caller Preferences for the Session Initiation Protocol (SIP)*, August 2004. RFC 3841.

[55] W. Kellerer, M. Wagner W.T. Balke, and H. Schulzrinne. *TPreference-based Session Management for IP-based Mobile Multimedia Signaling*, October 2003.