

# ◆ Advanced IMS Client Supporting Secure Signaling

Ramana Isukapalli, Steven Benno, Candace Park,  
and Peretz M. Feder

*With recent advances in core and access networks and the availability of increased bandwidth and sophisticated devices for end users, there is an increased demand for client applications running on mobile devices, such as laptops and handheld devices, to support real time applications like Voice over Internet Protocol (VoIP) and streaming video, apart from traditional applications like web browsing. This paper presents a prototype IP Multimedia Subsystem (IMS) client, which serves as a VoIP client to set up calls between Internet Protocol (IP) devices and interworks with circuit-switched networks to deliver calls to public switched telephone network (PSTN) phones. It implements supplementary services (including call waiting, call transfer, and call forwarding); supports multimedia ringing, short message service/multimedia messaging service (SMS/MMS), audio/video conferencing, and peer-to-peer video; and it can deliver a call to a user (as opposed to a device) by simultaneously ringing multiple devices registered by the user. Further, to address various security concerns, the client supports Hypertext Transfer Protocol (HTTP) digest authentication using Message Digest 5 (MD5) cryptographic function authentication and key agreement (AKA) and can create secure tunnels to the core network using IP security (IPsec). © 2008 Alcatel-Lucent.*

## Introduction

There have been many advances in access and core networks. The bandwidth available to the terminals of the end user has increased significantly. Today, using wireless access technologies like evolution data optimized (EV-DO) [13], Universal Mobile Telecommunications System (UMTS) [1], and Worldwide Interoperability for Microwave Access (WiMAX) [9], the end user has access to enough bandwidth for many real time applications like streaming video and multimedia services. The IP Multimedia Subsystem (IMS) [2] provides an all

Internet Protocol (IP) core network that can work with any of the wireless access networks, Ethernet, or even the traditional circuit-switched public switched telephone networks (PSTN). With these networks, service providers can now enable voice calls among different users using the less expensive option of transmitting Voice over Internet Protocol (VoIP), instead of the traditional way of transmitting voice over circuit-switched networks.

In addition to the networks, end user mobile client devices have become quite sophisticated in

### Panel 1. Abbreviations, Acronyms, and Terms

3GPP—3rd Generation Partnership Project  
AAA—Authentication, authorization and accounting  
AH—Authorization header  
AKA—Authentication and key agreement  
ANM—Answer message  
AUTN—Authentication token  
AVP—Audio video profile  
BREW—Binary runtime environment for Windows  
CDMA—Code division multiple access  
CFB—Call forwarding busy  
CFD—Call forwarding default  
CFNA—Call forwarding not available  
CK—Cipher key  
CSCF—Call session control function  
eMRS—enhanced Media Resource Server  
ESP—Encapsulating security payload  
EV-DO—Evolution data optimized  
EVRC—Enhanced variable rate codec  
HSS—Home subscriber server  
HTTP—Hypertext Transfer Protocol  
IAM—Initial address message  
I-CSCF—Interrogating CSCF  
IETF—Internet Engineering Task Force  
IK—Integrity key  
IMPI—IMS private identity  
IMS—IP Multimedia Subsystem  
IP—Internet Protocol  
IPsec—Internet Protocol Security  
ISDN—Integrated services digital network  
ISUP—ISDN user part

J2ME—Java 2 Micro Edition  
MD5—Message Digest 5  
MMS—Multimedia messaging service  
MMSc—Multimedia message service center  
P-CSCF—Proxy CSCF  
PDA—Personal digital assistant  
PRACK—Provisional acknowledgement  
PSTN—Public switched telephone network  
RAND—Random challenge  
REL—Release message  
RLC—Release complete  
RTP—Real Time Transport Protocol  
SA—Security association  
S-CSCF—Serving CSCF  
SCTP—Stream Control Transmission Protocol  
SDP—Session Description Protocol  
SIP—Session Initiation Protocol  
SMS—Short message service  
SMSc—Short Message Service Center  
SMV—Selectable mode vocoder  
SPI—Security parameter index  
SS7—Signaling System 7  
TCP—Transmission Control Protocol  
UCP—Universal Computer Protocol  
UDP—User Datagram Protocol  
UMTS—Universal Mobile Telecommunications System  
URI—Uniform resource identifier  
URL—Uniform resource locator  
VoIP—Voice over Internet Protocol  
WiMAX—Worldwide Interoperability for Microwave Access

terms of the features and functionality they support and the amount of data they can store. Today, most of the mobile handheld devices are IP enabled, can play audio and video files, have good screen resolution, and can also provide many programming tools for novel applications. As a result, there is an increased demand on the client devices to support not just voice calls, but a variety of other applications like VoIP, multimedia ringback, short message service (SMS), multimedia messaging service (MMS), and peer-to-peer video. Security can be a big issue and a concern for any client that supports these features. Although a variety of VoIP clients are available today, there is no single client that supports such a rich set of features and addresses all security concerns.

This paper presents an Alcatel-Lucent IMS client that supports the features above using Session Initiation Protocol (SIP) [3, 4]; the client supports both IMS SIP [3, 4] and Internet Engineering Task Force (IETF) SIP [12]. It can also deliver voice calls to (or receive voice calls from) traditional PSTN phones using a PSTN↔IP gateway. It supports supplementary services (like call waiting, call transfer, call forwarding), SMS, MMS, buddy lists, multimedia ringback, and peer-to-peer video. The Alcatel-Lucent IMS client is implemented on Windows\* XP, Windows 2000, and Windows CE (Pocket PC). To address security concerns, these clients have Hypertext Transfer Protocol (HTTP) digest authentication using Message Digest 5 (MD5) cryptographic function [11].

Additionally, the client on Windows XP and Windows 2000 also supports authentication and key agreement (AKA) [5]. Further, it can use the keys generated by AKA to set up secure Internet Protocol security (IPsec) tunnels with the proxy call session control function (P-CSCF) of the IMS core network. Hence, all the messages between the client and P-CSCF go through the secure IPsec tunnels. The client is a very useful tool in many of Alcatel-Lucent's product portfolios, including EV-DO, WiMAX, as well as IMS. However, it is a product of research activity and is not sold as a commercial product. In this paper, we will first present an overview of SIP and describe the various features of the Alcatel-Lucent IMS client with call flows and other details.

## Overview of SIP

SIP is a signaling protocol that runs at the application layer of the IP stack; it facilitates communications between different users by providing the means to find and contact one another and then negotiate the various parameters to set up a media session. The media session path is separate from the SIP signaling path. SIP allows users to be attached to the network at different access points and still be found by other users and applications. It is designed to be transport layer agnostic and it runs on Transmission Control Protocol (TCP), User Datagram Protocol (UDP), and Stream Control Transmission Protocol (SCTP).

SIP has a client server architecture in which the SIP clients (or user agents) are the endpoints. The SIP proxy server keeps track of the end location (contact IP address) of the clients. Users are identified by their user uniform resource identifier (URI). By registering with SIP registration servers, the user URI gets bound to the IP addresses of the devices the user has attached to the network. So, using a proxy server, registered users can be contacted by their URI, even if their contact address is not known. Any user can thus find the contact address of other users using the SIP proxy server and then establish a media (e.g., voice) session with them.

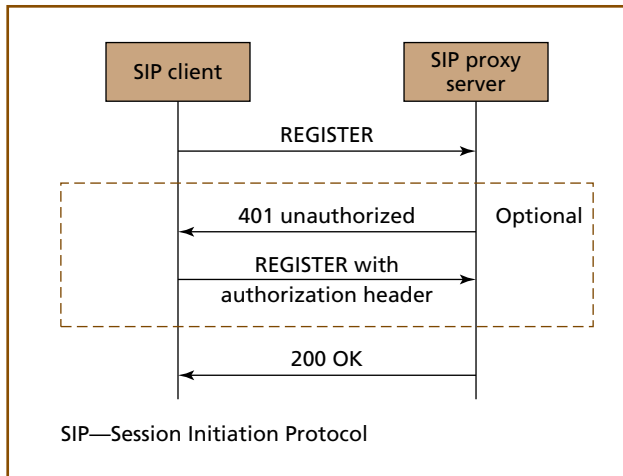
Session Description Protocol (SDP) messages carried in SIP messages enable session parameters to be negotiated between users. SDP is defined in [8]. A typical trace of an SDP message included in a SIP message is given below:

```
v = 0
o = SIPClient 0 0 IN IP4 135.5.6.7
s = Our conversation
c = IN IP4 135.5.163.227
t = 0 0
m = audio 7078 RTP/AVP 4 18 98 0 8
a = sendrecv
```

Here the SDP fields *v*, *o*, *s*, *c*, *t*, *m*, and *a* refer to the protocol version (*v*), originator of session (*o*) (along with the username (SIPClient), session id (0), version (0), network type (IN, i.e., Internet), address type (IPv4) and address, 135.5.6.7 in the above example), session identifier (*s*), connection information (*c*), time the session is active (*t*), media name (audio, Real Time Transport Protocol (RTP) port = 7078 and the supported codecs), transport attributes (*m*) and attribute lines (*a*), respectively. Two clients that wish to establish a media session exchange their SDPs (as shown above) in SIP messages. This will enable them to negotiate common attributes (like supported codecs), find out about each other's RTP ports, and such other parameters needed to establish a media session. Please refer to [8] for details.

## IETF SIP

The core IETF SIP protocol is described in [6, 7]. In IETF SIP, to set up a session (e.g., VoIP session) in its simplest form, all that is required are two end users with knowledge of how to contact each other. IETF SIP networks can also include registration servers, proxy servers, and application servers; these extend the functionality of SIP but are not required to set up a session. When a user attaches to a new access point on a network, they can register their new location with a SIP proxy server (or registrar) using the REGISTER message, shown in **Figure 1**. In the figure, REGISTER is the request sent by the user. It contains the user URI, also known as the user's "well-known" address, and the IP address of the device the user is currently using. The proxy server binds the user URI to the device's IP address and stores it in its database. 200 OK is the response sent by the server indicating a successful registration. It confirms the request information and adds an expiration time to the registration. The user must re-register before the expiration time is up to maintain the registration. The figure also shows two additional messages (401 Unauthorized

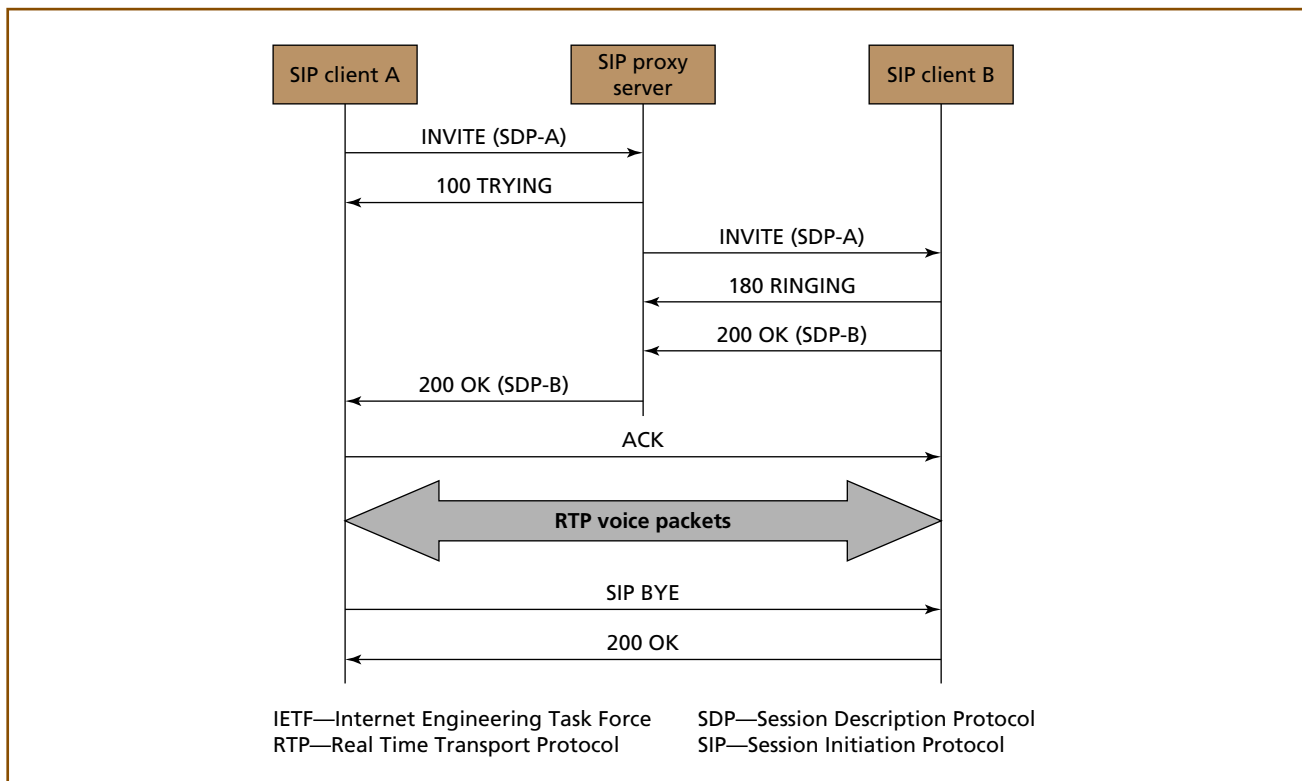


**Figure 1.**  
**SIP registration.**

and REGISTER with authorization header) that are used during authentication. These messages are optional and are used only if the proxy server requires authentication. The authorization header is computed using an authentication algorithm (and any required parameters) like MD5.

**Figure 2** shows how a VoIP session is established between two clients. In the figure, client A sends a SIP INVITE message to the proxy server indicating the intention to set up a VoIP call (or some other session, like a peer-to-peer video session) with client B.

The SIP message also includes an SDP message (similar to the one shown above) that lists the various parameters like audio codecs supported by the client and RTP port. The proxy server sends an INVITE message to client B including the SDP parameters as sent by client A. Client B accepts the call (assuming it is registered and is reachable) and eventually sends a SIP 200 OK message (with its SDP message in it) to the proxy server. The proxy server in turn sends a SIP 200 OK message to client A indicating a successful call setup. The two endpoints thus negotiate the relevant parameters, set up a VoIP session, and communicate with each other using voice packets on their RTP ports. The call flow to set up other media sessions is the same, except that the SDP parameters used to set up the session are different



**Figure 2.**  
**Call delivery using IETF SIP.**

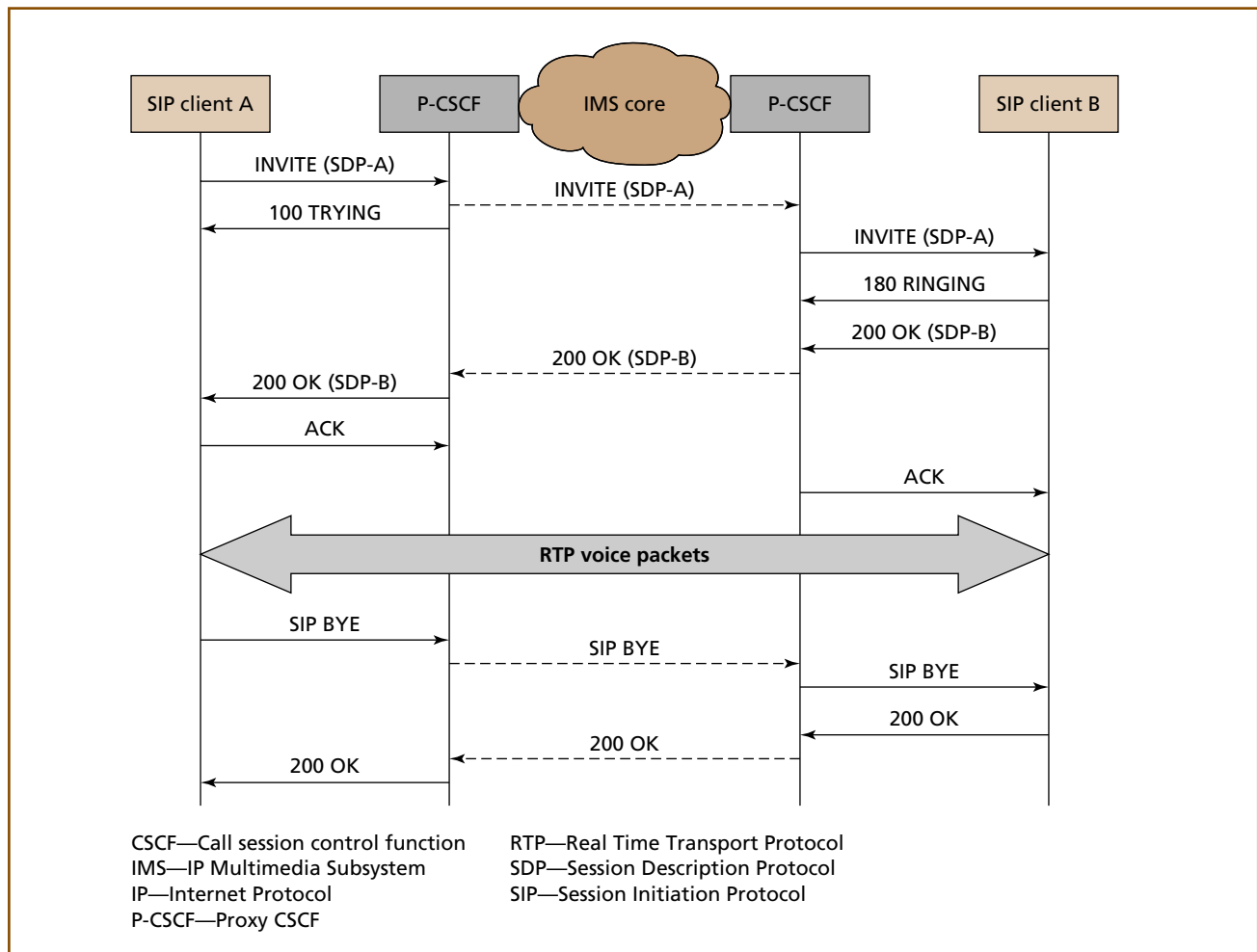
and are specific to the media session that needs to be set up.

### IMS SIP

IMS SIP is explained in [3], and the 3rd Generation Partnership Project (3GPP\*) IMS profile for SIP is defined in [4]. **Figure 3** shows how a VoIP call is set up using IMS SIP. It is similar to the call setup shown in Figure 2. Here, P-CSCF acts as the proxy server. Moreover, all the SIP signals between the client device and P-CSCF go through secure tunnels.

As the name suggests, IMS provides enhanced multimedia services to end user devices of various access networks (both wireline and wireless) through call control using an IP-based core network. It uses the required media servers and gateways to

set up calls between the PSTN and IP devices. IMS requires the use of proxy and registration servers. P-CSCF acts as the first contact point of the IMS core network; it communicates directly with a client device. It assists in authentication, setting up IPsec tunnels with client devices, and routing SIP messages to the correct SIP servers, and other devices. Serving CSCF (S-CSCF) downloads and caches the user profile from the home subscriber server (HSS), terminates registration requests, and acts as a registration server for IMS clients. It can authenticate a client and can help in creating an IPsec tunnel between the client and P-CSCF. The interrogating CSCF (I-CSCF) assists in finding the correct S-CSCF. IMS mandates extensions to the IETF SIP to enable it to work in IMS networks.



**Figure 3.**  
Call delivery using IMS SIP.

IMS SIP has more access security mechanisms; these are defined in [6]. User authentication is required in IMS: i.e., the registration procedure shown in Figure 1 always uses authentication. The two optional messages (401 Unauthorized and REGISTER with authorization header) are no longer optional; they are mandatory. Further, IMS SIP mandates the use of secure tunnels between the client device and P-CSCF. IPsec is the only security association currently defined over the Gm interface, but the plan is to allow alternate security association mechanisms. IPsec implementation on the client is explained later in this paper. To support the creation of security associations, “security-client,” “security-server,” and “security-verify” headers are used in IMS SIP. Their syntax is defined in [7] and in Annex H of [6].

### Alcatel-Lucent IMS Client

This section explains the various features—VoIP, supplementary services, conference calling, SMS and MMS, multimedia ringing, and peer-to-peer video, which are supported by the Alcatel-Lucent IMS client.

#### Voice Over Internet Protocol

The call flow to deliver a call using IETF SIP is given in Figure 2 and the call flow using IMS SIP is shown in Figure 3. Using IETF SIP, the Alcatel-Lucent IMS client can set up calls using a proxy server, as shown in Figure 2, or without using a proxy server, if it knows the contact IP address of the terminating side SIP client. IMS SIP mandates the use of IMS core network in call setup as shown in Figure 3.

#### Calls to/from PSTN Phones

The Alcatel-Lucent IMS client can deliver calls to (and also receive calls from) PSTN phones using an IP↔PSTN gateway. This gateway is used for two reasons. First, it is used to translate SIP messages to integrated services digital network user part (ISUP) messages, and vice versa, during call setup. Second, it acts as a media gateway to translate voice (IP) packets to voice streams that can be used on a circuit-switched network, and vice versa. Note that only one gateway is used here for these two functions; however, it can use two separate gateways.

The call flow for delivering a call from the Alcatel-Lucent IMS client to a PSTN phone is shown in

**Figure 4.** At a high level, the client sends an INVITE message to the SIP proxy server. The proxy server notes that the terminating side is a PSTN phone and sends a SIP INVITE message to the IP↔PSTN gateway. The gateway sends an ISUP initial address message (IAM) to the terminating PSTN switch to set up a call with the PSTN phone. The switch responds with an ISUP answer message (ANM) which is translated to a SIP 200 OK by the IP↔PSTN gateway that is then sent to the client. The client responds with a SIP ACK message and sets up media (voice packets) through its RTP port. The call is thus successfully set up. The IP↔PSTN gateway converts the voice packets to voice streams for delivery to the PSTN phone through the PSTN switch.

Either side can end a call. The IMS client ends a call by sending a SIP BYE signal that is translated to an ISUP release message (REL) by the IP↔PSTN gateway and is sent to the PSTN switch, which in turn hangs up the call and responds with an ISUP release complete (RLC) message.

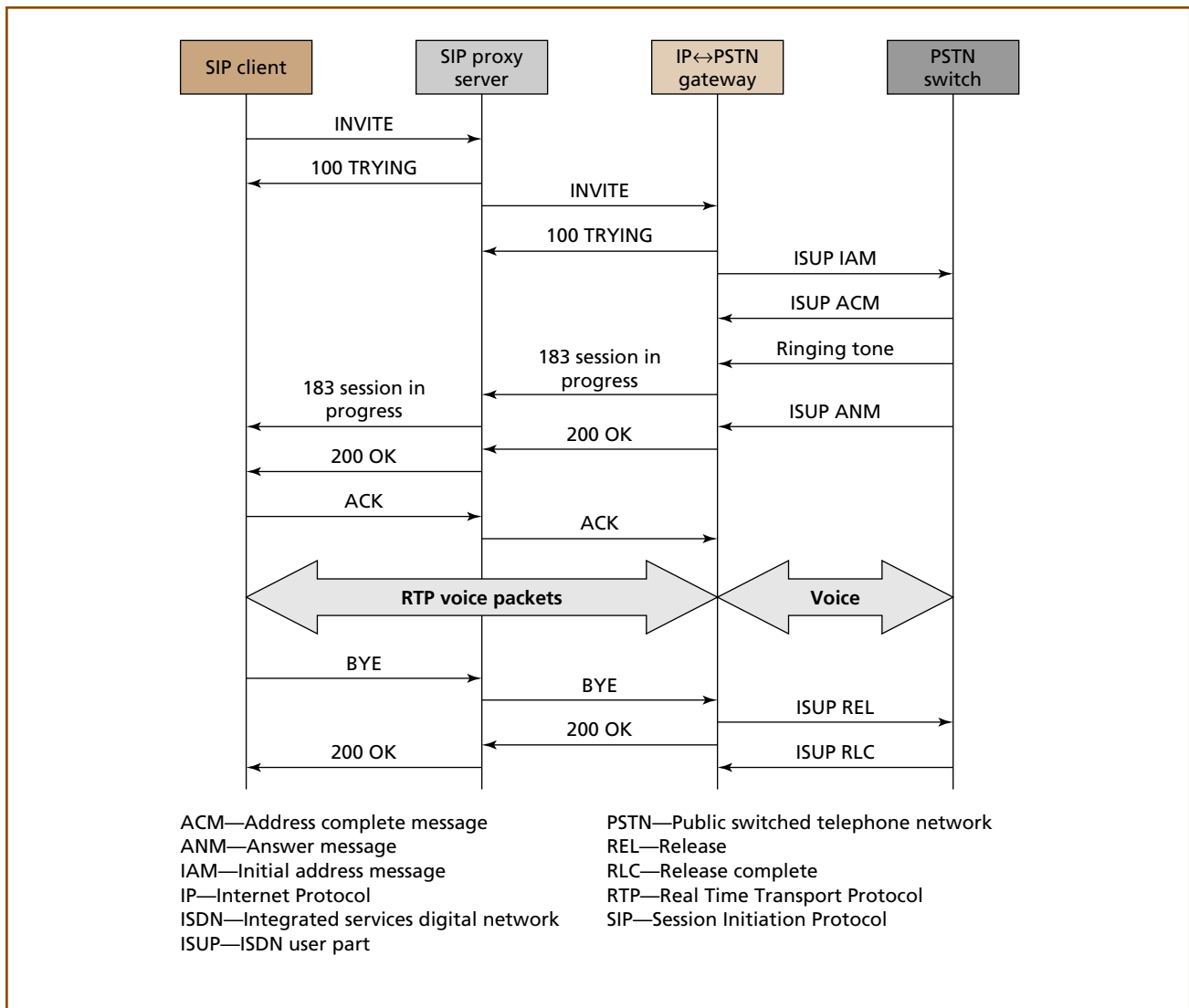
Call delivery to cell phones in wireless networks, like UMTS and code division multiple access (CDMA), is similar to the one shown in Figure 4. SIP signaling messages are converted to the protocol (UMTS or CDMA) specific signaling messages, instead of the PSTN messages, using a gateway. Voice (IP) packets are converted to voice streams by a media gateway before they are delivered to the device on the terminating side of the call.

#### Supplementary Services

The Alcatel-Lucent IMS client supports many of the supplementary services that a typical traditional PSTN phone supports. Some of these features are listed below with relevant details.

**Caller ID.** The IMS client accepts incoming calls (from either other VoIP devices or PSTN phones) using the SIP INVITE message. The client extracts the user ID of the incoming call from the “From” field of the incoming SIP INVITE message and displays it to the user receiving the call.

**Call waiting.** The IMS client can put on hold any active call by suspending the RTP (voice) packets temporarily, but not closing the RTP session. The call can be made active again by reactivating the RTP session.



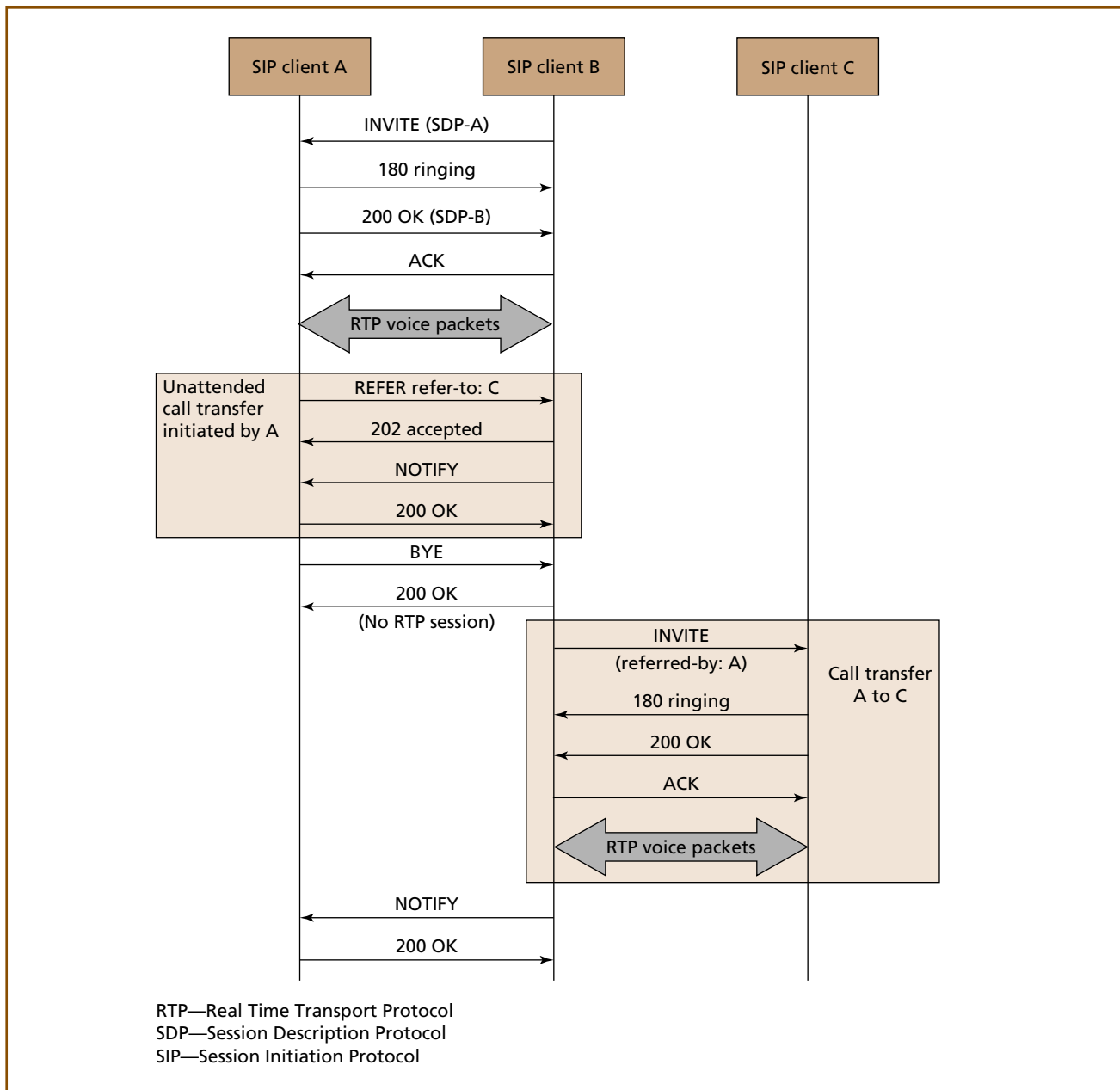
**Figure 4.**  
Call delivery to a PSTN phone.

The IMS client uses this feature to implement call waiting; i.e., it places on hold any active call by temporarily suspending their RTP sessions. It can then set up a new session with any other VoIP client and can switch between the two calls as needed.

**Call transfer.** Figure 5 shows the call flow for call transfer. In the figure, user B (denoted by SIP client B) calls user A (denoted by SIP client A). Then, user A transfers user B to user C (denoted by SIP client C). The call transfer is accomplished via a SIP REFER message.

The first four messages in the figure denote the call setup between user A and user B (the proxy

server is not shown for simplicity). To transfer the call to user C, user A sends a SIP REFER message with the address of user C in the refer-to header. To initiate a call with user C, user B now sends an INVITE message to user C with the “referred-by” header set to user A’s address. User C responds with a SIP “180 ringing” message followed by a SIP “200 OK” message to set up the call. The call transfer is unattended; i.e., the voice stream between user A and user B is closed before the call is transferred. However, the dialog between user A and user B exists until user A receives a NOTIFY message from user B. That is when the subscription created by the REFER message



**Figure 5.**  
**Call transfer call flow.**

terminates and the dialog is closed. User A sends a 200 OK message to user B in response to the NOTIFY message.

**Call forwarding.** Call forwarding has different varieties, including call forwarding busy (CFB), call forwarding not available (CFNA), and call forwarding default (CFD). The IMS client uses an application server in the IMS core network called “feature server

5000” that supports many supplementary services for VoIP to implement call forwarding. That is, any user can set up any of these call forwarding features in the feature server. For any incoming call, the feature server checks whether there are any matching call forwarding features active for the user and forwards the call appropriately. For example, if a user activates call forwarding default to his voicemail in the feature

server, any incoming call to the user is forwarded to his voicemail by the feature server.

**White listing and black listing.** The client can list specific users from whom he will accept (or reject) any incoming call, based on the identity of the user making the call. This list is stored in the feature server (end-to-end proxy). The list of users whose calls are accepted forms the white list, while the list of users whose calls are not accepted forms the black list. By default all the users are in the white list—no one goes to the black list unless explicitly specified. Any incoming call goes through the feature server and it delivers the call only if the user originating the call is not in the black list of the user terminating the call.

### Conference Calling

The Alcatel-Lucent IMS client implements conference calling using a “conference server,” as shown in **Figure 6**. A conference server is a special application server (not a SIP registrar) that is used to set up conference calls. As can be seen from the figure, any user (represented here by SIP client 1) can conference -in several other users (shown by SIP client 2, . . . , SIP client n) using the conference server. The first user sends a SIP INVITE message to the conference server to set up a conference call. The conference server sends a “SIP 302 temporarily moved” message with a conference ID that the user uses to send another INVITE message. Then the user sends a SIP REFER message for each user that needs to be placed in the conference call. The conference server sends a SIP INVITE message to all users separately and places them in the conference call as soon as it gets a SIP 200 OK message from them. If it does not receive such a message, it will skip that user (i.e., it will not put that user in a conference call). It notifies the first user (the one who initiated the conference call) whenever any user is added to the conference call.

### Buddy Lists

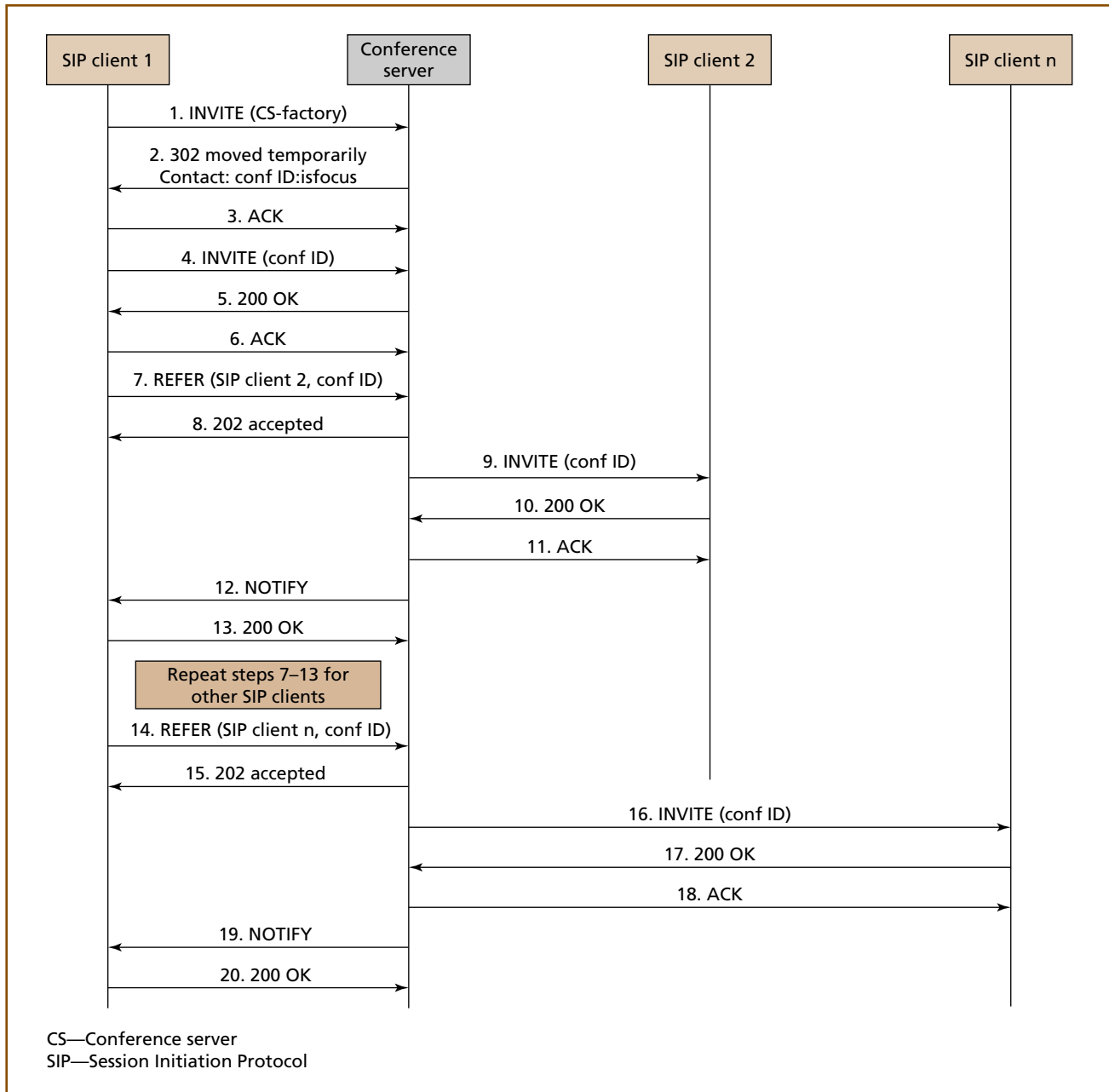
The Alcatel-Lucent IMS client supports buddy lists. Any user can add and remove other users from his buddy lists. Using SIP SUBSCRIBE and NOTIFY, the SIP proxy server notifies any user whenever the login status (e.g., logged in, busy, not available) of any of his buddies changes. That is, any user who

needs to be notified of the status of his buddies subscribes with a presence server for the service. During registration, any user subscribed to this service informs a presence server of his presence. The presence server relays the user’s presence status to his buddies and notifies the user of any buddies that are available online. Similarly, the status of any user is refreshed on his buddy’s list whenever the status of the user changes.

### SMS and MMS

The IMS client implements SMS and MMS using “Colibria\* gateway,” a gateway by a third party vendor that is capable of delivering SMS and MMS messages to both cell phones using Signaling System 7 (SS7) networks, as well as other SIP clients using IP networks. **Figure 7** shows the call flow to deliver short messages. The client sends a short message to another device using a “send SMS” message. This message first goes to the Colibria gateway. The gateway sends the message to the short message service center (SMSc) using Universal Computer Protocol (UCP). If the end user device for which the SMS is intended is a cell phone that is not IP enabled, SMSc delivers the message to it using the “deliver SMS” message. Otherwise, it forwards the message back to the Colibria gateway, since SMSc itself cannot handle SIP messages. The gateway, in turn, delivers the message to the target device using the “deliver SMS” message.

MMS delivery is similar to the SMS delivery. The call flow for MMS is given in **Figure 8**. The client sends a multimedia message to the Colibria gateway using a “send MMS” message. That, in turn, sends the message to the multimedia message service center (MMSc). If the end device to which the message is to be sent is a cell phone that is not IP enabled, MMSc sends the message to the SMSc using “MMS notification via SMS.” SMSc notifies the end device of the pending multimedia message using “MMS notification via SMS.” The device can download the message from SMSc using MMS retrieval. On the other hand, if the message is to be delivered to an IP device that supports SIP signaling, MMS forwards the message to the Colibria gateway directly since MMS cannot handle SIP messages. The gateway, in turn, delivers the message to the target client.



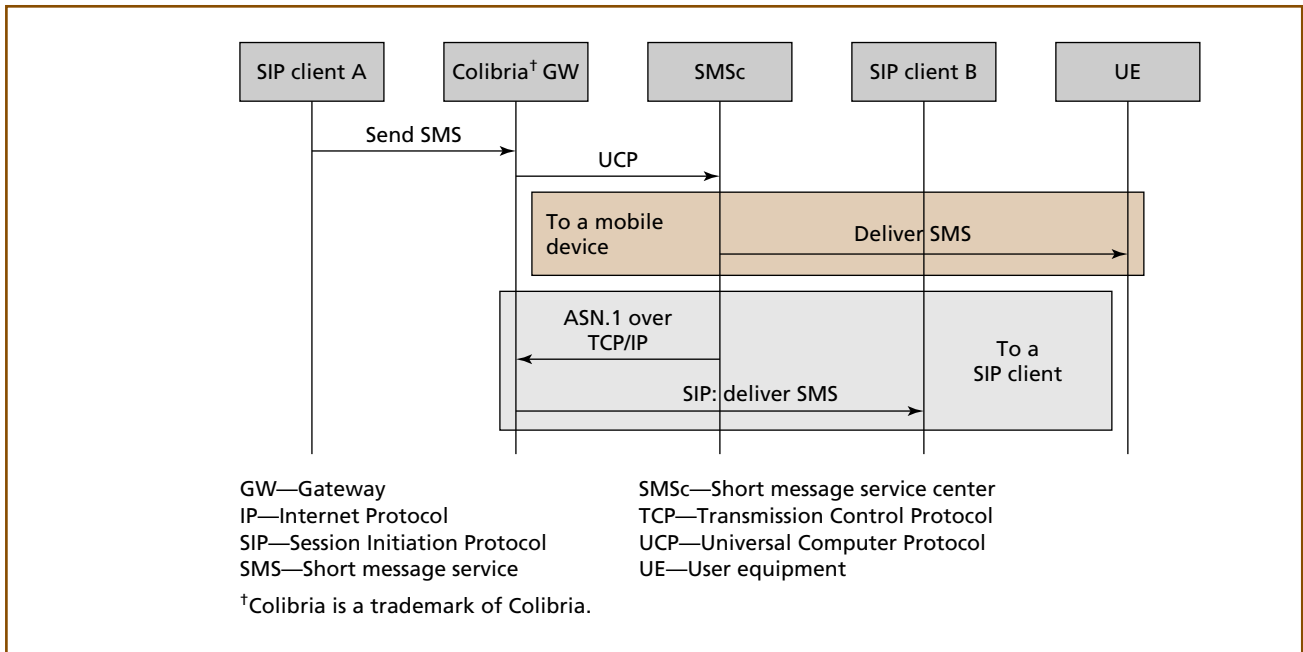
**Figure 6.**  
*Conference calling using a conference server.*

### Multimedia Ringing

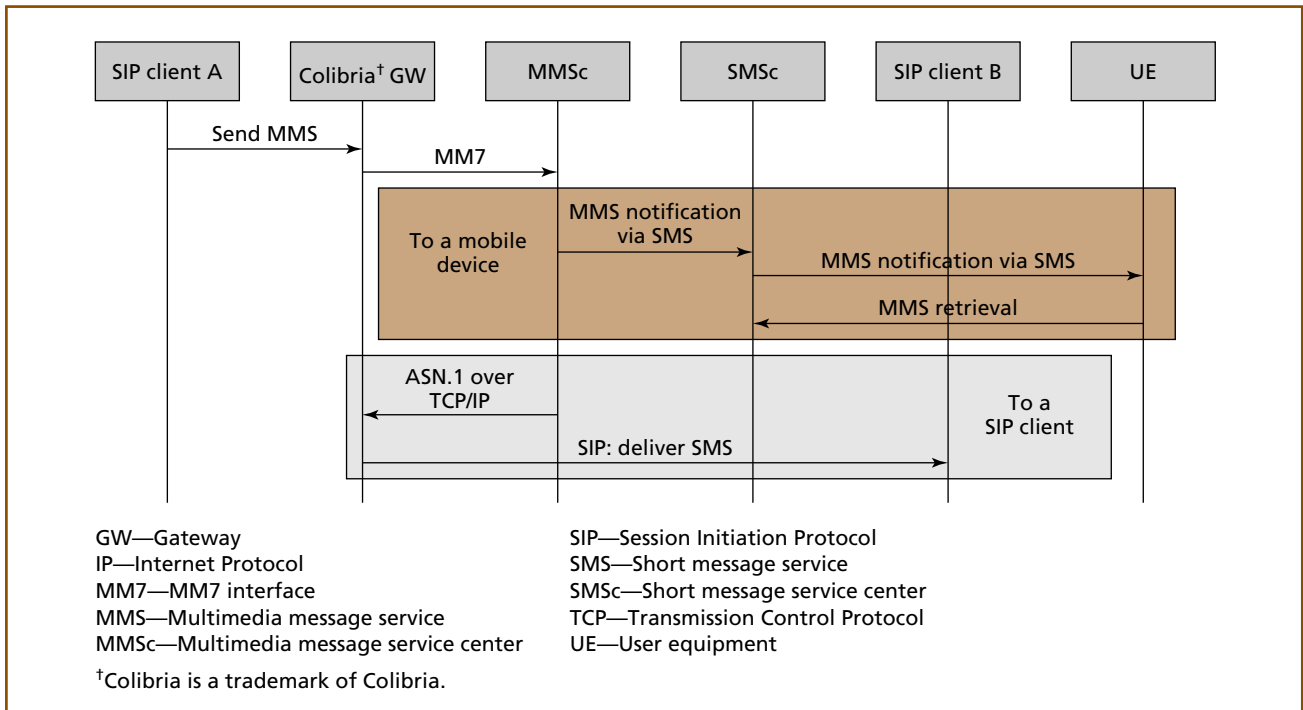
Multimedia ringing is a feature where for any outgoing call to a client that has enabled this feature, the client (originating the call) plays a multimedia file (audio or video) before the call is set up. If the terminating client does not support multimedia ringback, then the client plays the normal ringback (i.e., regular ringing tone) while the call is being set up. The

IMS client supports multimedia ringing using an enhanced media resource server (eMRS).

The call flow to implement this feature is shown in **Figure 9**. As shown in the figure, the client sends a SIP INVITE message to the SIP proxy server to initiate a call to another client. The message is sent to the eMRS, which responds with a SIP 183 ringing message that has an SDP message in it with a URL link



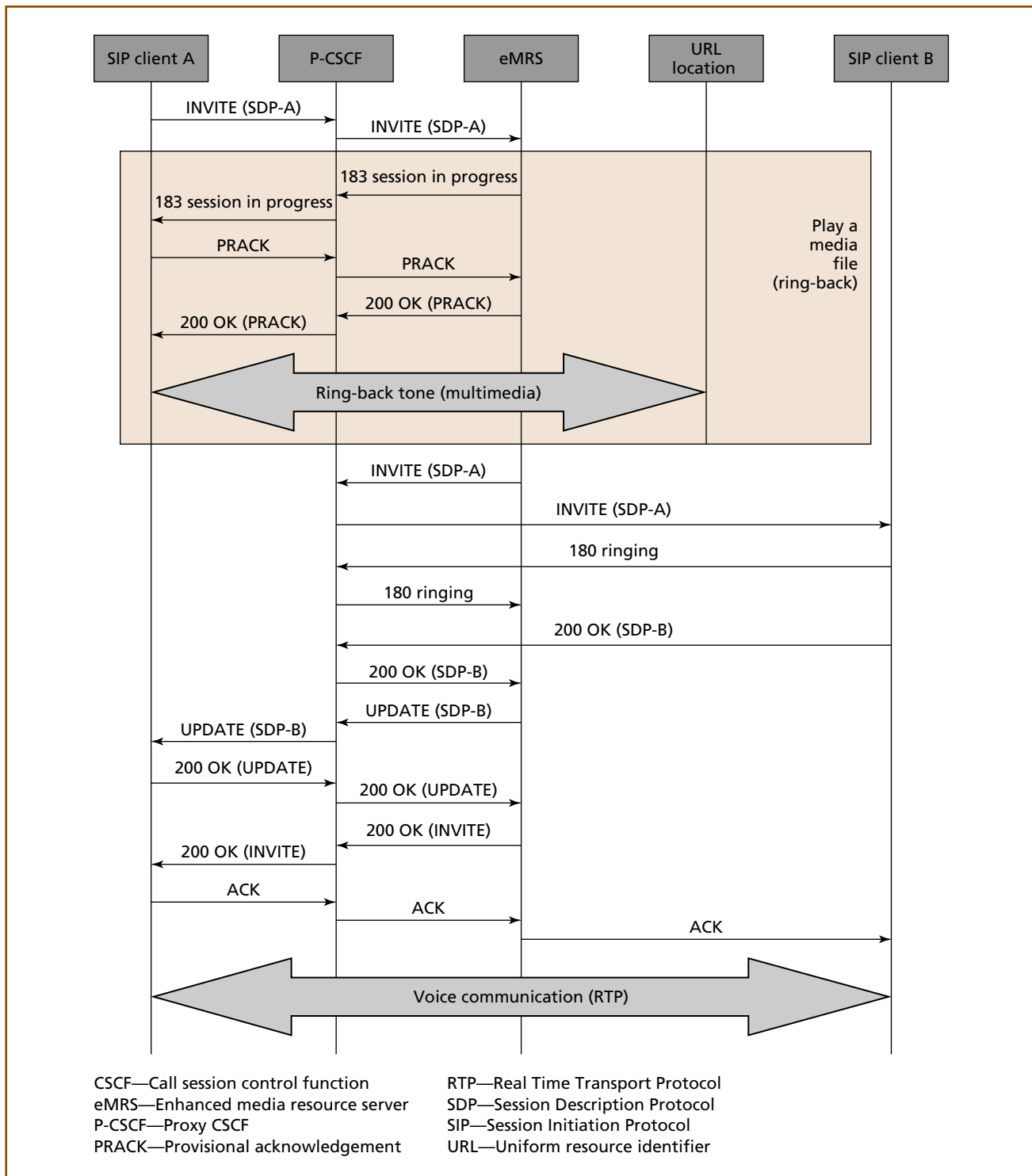
**Figure 7.**  
**SMS delivery to other SIP clients or to a mobile device.**



**Figure 8.**  
**MMS delivery to other SIP clients or to a mobile device.**

to the location of the media file to play. That is, the “m” field in the SDP indicates the type of media file (audio or video) along with the RTP port, and the “u” field of SDP gives the location of the media file. The client sends

a provisional acknowledgement (PRACK) message to eMRS through P-CSCF, to which eMRS responds with a 200 OK. The PRACK request message plays the same role of the ACK message—to initiate bearer traffic—but



**Figure 9.**  
**Call flow for multimedia ringing.**

it is a provisional message. In this case, it starts the multimedia ringback. The client launches the required (audio or video) media player, downloads the content of the URL, and plays it using the media player.

The rest of the messages in the figure show how a voice call is set up. They are similar to the messages in the call flow shown in Figure 3. The SIP INVITE message is sent to the terminating SIP client, which

responds with SIP 180 ringing, followed by SIP 200 OK messages. The SIP 200 OK message has the SDP of the terminating side. eMRS sends an UPDATE message, along with the SDP parameters of the terminating side, to the SIP client on the originating side to inform it of the SDP parameters of the terminating side. The call originating SIP client replies with a SIP 200 OK to the UPDATE message, and then with a SIP ACK message to the 200 OK message it receives from the eMRS. The client stops playing the multimedia file, then opens its RTP port for the VoIP call, and a voice session is established.

### Peer-to-Peer Video

The call flow for peer-to-peer video is the same as the VoIP call shown in Figure 2 (using IETF SIP) and Figure 3 (using IMS SIP). That is, the SIP messages are the same for peer-to-peer video and a VoIP call. However, the SDP parameters in the SIP messages—INVITE and 200 OK—will be different. The IMS client implements peer-to-peer video by setting the appropriate SDP parameters in the SIP message. That is, we set “m=video” along with the RTP port and the supported video codecs. In peer-to-peer video, live video data captured continuously by a camera attached to an IMS client is sent to the other client. Either client can send video data to the other; i.e., only one client or both clients can send video to its peer.

### Security—Authentication and IPsec

Security is a big issue for any device that supports such a wide variety of features. The Alcatel-Lucent IMS clients implemented on Windows 2000, Windows XP, and Windows CE support HTTP digest authentication using MD5 [11]. Additionally, the clients on Windows 2000 and Windows XP also support AKA [10]. They can use the two keys, integrity key (IK), and cipher key (CK) generated by AKA to create security associations with P-CSCF (IMS core network) as specified in [6].

#### HTTP digest authentication using MD5.

Authentication occurs during registration, as shown in Figure 1. The client sends an initial SIP REGISTER message to which the SIP proxy server replies with a 401 unauthorized message. This message includes a random nonce (i.e., a random string of characters generated by the SIP authenticator). The client

computes the response using the random nonce along with a password that is known only to the client and the authentication, authorization and accounting (AAA) server in the core network. It sends the response in an authorization header in a second REGISTER message to the proxy server [11]. The proxy server responds with a SIP 200 OK message indicating that the authentication is successful, or a SIP 403 forbidden message, if the authentication fails.

**Authentication and key agreement.** The message flow for AKA is similar to that of HTTP digest authentication using MD5, shown in Figure 1. However, the parameters that are used in the SIP 401 unauthorized message and the authentication algorithm are different. AKA is a two way algorithm in which the proxy server authenticates the client and the client also authenticates the proxy server (P-CSCF and the core network in the case of IMS). The basis for authentication is through the association of an “IMS private identity” (IMPI) with a long-term secret (K), both of which are provisioned in the client and the proxy server. The SIP 401 unauthorized message from the proxy server contains a random challenge (RAND) and an authentication token (AUTN) field, among others, that are sent by the proxy server. The client uses AUTN to authenticate the server and creates a response using RAND and K. It then uses the response to create an authorization header that is sent as part of the second REGISTER message to the proxy server. The proxy server uses this response to authenticate the client [10]. If the authentication fails either at the client or at the server, the process terminates and the registration is unsuccessful.

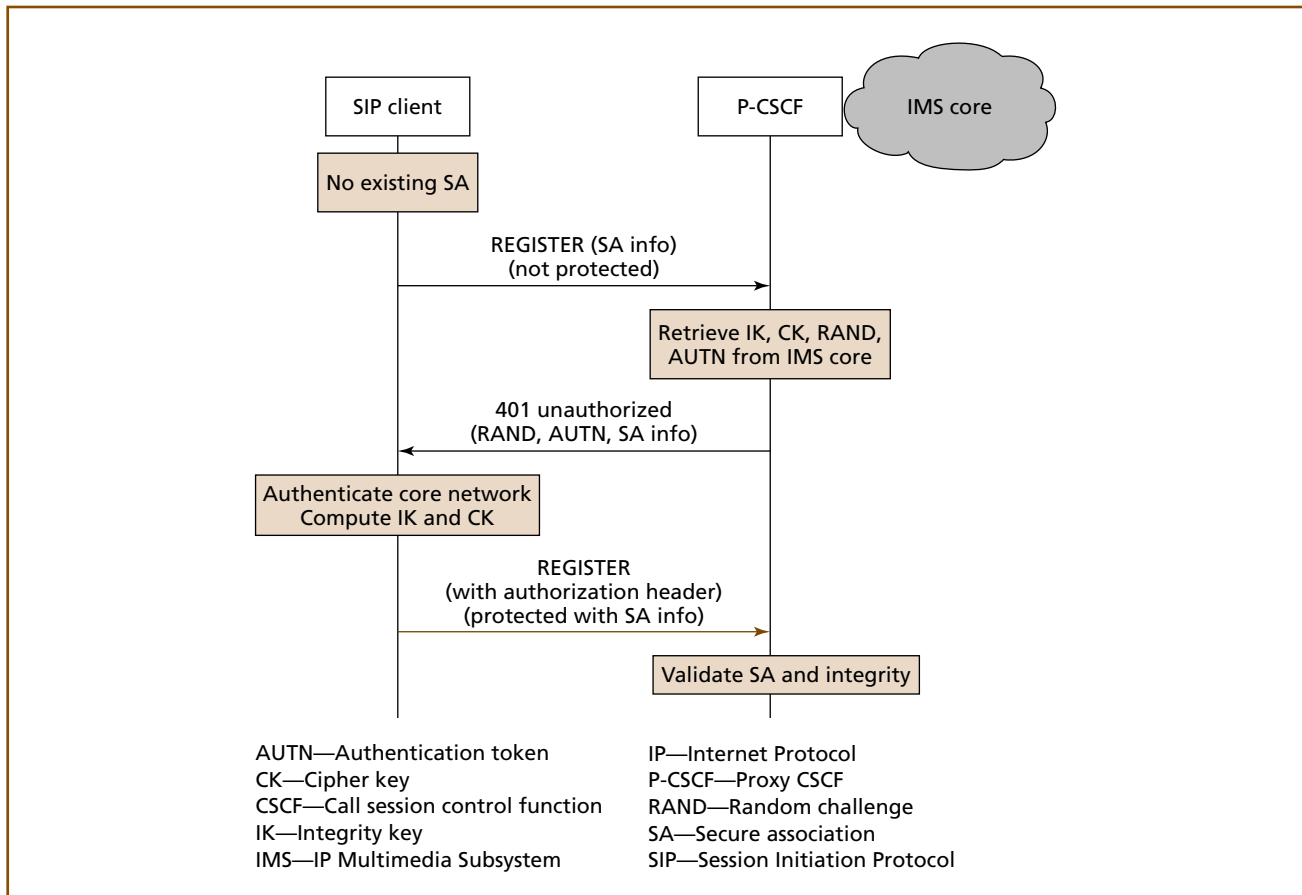
**IPsec.** IPsec in the IMS client is based on the keys generated by AKA; i.e., the client performs an AKA authentication as explained earlier. As a result of successful authentication, the client uses the RAND (it receives from the server) and K fields to generate two keys, IK and CK. The two keys are used to create IPsec tunnels; IK is used to create the “authorization header” (AH) and CK is used to create the “encapsulating security payload” (ESP) header.

Creating secure IPsec tunnels basically involves negotiating a “security association” (SA) between the client and server (P-CSCF). The client and server negotiate the integrity algorithm, encryption algorithm,

“security parameter index” (SPI), two ports on the client—one port is used as a “server port” to accept incoming SIP messages from the server, and the other port is used as a “client port” to send SIP messages to the server—and two ports on the server. Negotiating these parameters to create security associations happens during registration [6]. The call flow for this is shown in **Figure 10**. It is similar to the generic registration call flow shown in Figure 1. The first REGISTER message includes the authentication and encryption algorithms that the client supports, and the client’s server port and client port, in addition to the other parameters required for registration. The server sends a 401 unauthorized message to the client that includes the authentication and encryption algorithms that the server supports, and the server’s client and server ports, apart from the other parameters required in the message. The client performs AKA as

explained earlier. As part of the second REGISTER message that carries the authorization header, the client indicates the authentication and encryption algorithms that are supported by both the client and the server. Additionally, the client indicates its server and client ports in the REGISTER message. The parameters required to set up a security association are thus negotiated.

Once the parameters are negotiated, any subsequent SIP messages from the client to the server, or vice versa, going on the ports of the security association are sent over the secure IPsec tunnels. That is, the initial SIP REGISTER and 401 unauthorized messages are not protected by the IPsec tunnels; they are used to create tunnels. The subsequent REGISTER and other messages from the client to the server (or vice versa) go through secure IPsec tunnels. The security associations are renewed periodically, before they expire.



**Figure 10.**  
*Secure associations negotiation during registration.*

## Related Work

There are a number of VoIP clients available today, many of which have been developed in the last two or three years. Some of these are commercial products, while some others are based on open source software. In spite of the availability of many VoIP clients that support a variety of features, there is no single client that supports as many features as the Alcatel-Lucent IMS client, which also addresses security issues in terms of HTTP digest authentication using MD5 and AKA. Currently, none of the other clients support IPsec for SIP signaling.

## Conclusions

This paper presents the details of the Alcatel-Lucent IMS client, as a topic of research, implemented on Windows 2000, Windows XP, and Windows CE. It describes the various features supported by the Alcatel-Lucent IMS client with call flows described wherever relevant.

## Contributions

The main contribution of this paper is an integrated IMS client that can be used for a variety of purposes. When used as a VoIP phone, it offers most of the traditional PSTN phone features like supplementary services and conference calling. As an IMS client, it provides novel features of IMS like peer-to-peer video, multimedia ringback, SMS, and MMS, as well as enhanced security features that include HTTP digest authentication using MD5, AKA, and IPsec. It secures all the signaling messages to the IMS core network and can be used to establish media, i.e., audio or video sessions, among clients. It can deliver audio calls using a variety of codecs like G723, enhanced variable rate codec (EVRC), G711 ( $\mu$ -law, A-law), and selectable mode vocoder (SMV). The sessions can be peer-to-peer or can be set up by a SIP proxy server. Calls can also be made into the PSTN phones using the proper gateways.

## Future Work

There are several interesting and useful extensions to this work. One useful feature to support on an IMS client is real time file sharing. This can be a useful tool for shared white board diagrams, in that text can be drawn on the board by either party and seen by the other in real time. Another interesting

extension to IMS clients is the creation of multiple media sessions between two or more clients. For example, users can share a white board while engaging in voice conversations. In "shared applications" several remote users can control an application, thus enabling true real time interaction among many remote users. Interfacing with online gaming applications is yet another useful application. Implementation of network address books and video conferencing are other useful extensions to this work.

Handheld devices such as cell phones and personal digital assistants (PDAs) have become quite sophisticated, providing numerous programming tools. These tools are being used in a variety of applications, not just for voice calls. It will be useful to have a VoIP client with as many features as possible implemented on these devices. A prototype VoIP client exists on binary runtime environment for Windows (BREW) and Java\* 2 Micro Edition (J2ME\*) devices and efforts are underway to implement the features listed in this paper on these devices.

## \*Trademarks

Colibria is a trademark of Colibria.

Java and J2ME are trademarks of Sun Microsystems Inc. Windows is a registered trademark of Microsoft Corporation.

## References

- [1] 3rd Generation Partnership Project, "Network Architecture," 3GPP TS 23.002, <<http://www.3gpp.org/ftp/Specs/html-info/23002.htm>>.
- [2] 3rd Generation Partnership Project, "IP Multimedia Subsystem (IMS), Stage 2," 3GPP TS 23.228, <<http://www.3gpp.org/ftp/Specs/html-info/23228.htm>>.
- [3] 3rd Generation Partnership Project, "Signaling Flows for the IP Multimedia Call Control Based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP), Stage 3," 3GPP TS 24.228, <<http://www.3gpp.org/ftp/Specs/html-info/24228.htm>>.
- [4] 3rd Generation Partnership Project, "Internet Protocol (IP) Multimedia Call Control Protocol Based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP), Stage 3," 3GPP TS 24.229, <<http://www.3gpp.org/ftp/Specs/html-info/24229.htm>>.
- [5] 3rd Generation Partnership Project, "3G Security, Security Architecture," 3GPP TS

- 33.102, <<http://www.3gpp.org/ftp/Specs/html-info/33102.htm>>.
- [6] 3rd Generation Partnership Project, "3G Security, Access Security for IP-Based Services," 3GPP TS 33.203, <<http://www.3gpp.org/ftp/Specs/html-info/33203.htm>>.
  - [7] J. Arkko, V. Torvinen, G. Camarillo, A. Niemi, and T. Haukka, "Security Mechanism Agreement for the Session Initiation Protocol (SIP)," IETF RFC 3329, Jan. 2003, <<http://www.apps.ietf.org/rfc/rfc3329.txt>>.
  - [8] M. Handley, V. Jacobson, and C. Perkins, "SDP: Session Description Protocol," IETF RFC 4566, July 2006, <<http://www.apps.ietf.org/rfc/rfc4566.txt>>.
  - [9] Institute of Electrical and Electronics Engineers, "IEEE Standard for Local and Metropolitan Area Networks, Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems," IEEE 802.16e-2005, Feb. 28, 2006, <<http://www.ieee.org>>.
  - [10] A. Niemi, J. Arkko, and V. Torvinen, "Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA)," IETF RFC 3310, Sept. 2002, <<http://www.apps.ietf.org/rfc/rfc3310.txt>>.
  - [11] R. Rivest, "The MD5 Message-Digest Algorithm," IETF RFC 1321, Apr. 1992, <<http://www.apps.ietf.org/rfc/rfc1321.txt>>.
  - [12] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "SIP: Session Initiation Protocol," IETF RFC 3261, June 2002, <<http://www.apps.ietf.org/rfc/rfc3261.txt>>.
  - [13] Telecommunications Industry Association, "cdma2000<sup>®</sup> High Rate Packet Data Air Interface Specification," TIA-856-A, Apr. 2004, <<http://www.tiaonline.org>>.

*(Manuscript approved August 2007)*

*RAMANA ISUKAPALLI is a member of technical staff in the Wireless IP Access Technology Group, Wireless Systems Core Technology Department, at Alcatel-Lucent in Whippany, New Jersey. He holds a Ph.D. in computer science from Rutgers University, New Brunswick, New Jersey; M.S. in computer science from Oregon State University, Corvallis; and a B.Tech. in metallurgy from the Indian Institute of Technology (IIT), Madras, India. He worked on several aspects of*



*the Alcatel-Lucent IMS client, including the registration module, MD5 authentication, and IPsec. In the past, he worked on the design and development of the registration module of SuperDHLR for UMTS and CDMA. His current interests include efficient content distribution to mobile devices, mobile IP and interworking issues between different wireless technologies, such as WiMAX and EV-DO on mobile devices. He also worked in machine learning and computer vision in the past and published several research papers in international conferences in these areas. Dr. Isukapalli is a senior member of the Institute of Electrical and Electronics Engineers (IEEE).*



*STEVEN BENNO is a member of technical staff in Bell Labs' Multimedia Processing Technology unit in Murray Hill, New Jersey. He is currently developing a version of the IMS personal computer (PC) client for various deployments. His interest is to create a client that provides a rich user experience by blending multiple services into a seamless and intuitive user experience, and to create a client platform for rapid development of new services and features. He received his B.S.E.E. from Rutgers University, New Brunswick, New Jersey; M.S.E.E. from Columbia University, New York, New York; and Ph.D. from Carnegie Mellon University, Pittsburgh, Pennsylvania, all in electrical engineering. In AT&T Bell Labs Federal Systems, he developed sonar signal processing algorithms and systems. After earning his Ph.D., he developed speech compression algorithms for wireless networks. In 2000, he was awarded the Excellence in Access Award from the Association of Access Engineering Specialists (AAES) for his work to make wireless systems accessible to people with hearing and speech impairments.*



*CANDACE PARK is a member of technical staff in the Wireless IP Access Technology Group, Wireless Systems Core Technology Department, at Alcatel-Lucent in Whippany, New Jersey. She received a B.S.C.S. from Fairleigh Dickinson University in Madison, New Jersey, and a M.S.C.S. from the New Jersey Institute of Technology in Newark. She has worked in the areas of digital transmission, pair gain testing, and video delivery. She is currently working on authentication and authorization issues in WiMAX.*

*PERETZ M. FEDER is a technical manager in the*



*Wireless Open Innovation Labs at Alcatel-Lucent in Whippany, New Jersey. He leads a group of network engineers who define, develop, characterize and verify wireless mobile IP networks including cellular and Institute of Electrical and Electronics Engineers (IEEE) systems. He holds numerous patents in the areas of wireless data networking and radio technology. He has worked with first, second, and third generation wireless systems, developing radio channel cards and protocols for the Alcatel-Lucent flagship AMPS and Flexent® networks. Recently, he has been working with high-speed fixed and fourth generation wireless systems and currently leads the Alcatel-Lucent Network Working Group delegation at the WiMAX Forum. A member of IEEE, Mr. Feder holds B.S.E.E and M.S.E.E degrees from the Columbia University School of Engineering in New York City. ♦*