

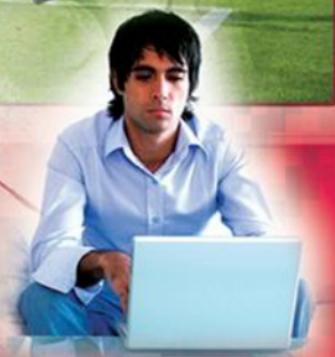
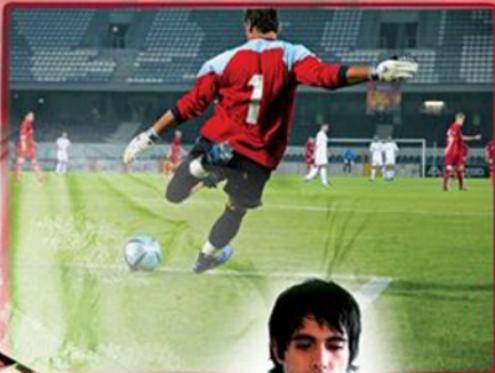
 WILEY

The  
**IMS**  
IP Multimedia  
Concepts and Services

Miikka Poikselkä

Georg Mayer

Third Edition



# **THE IMS**

## **IP MULTIMEDIA CONCEPTS AND SERVICES, THIRD EDITION**

**Miikka Poikselkä**

*Nokia Siemens Networks, Finland*

**Georg Mayer**

*Nokia, Finland*



A John Wiley and Sons, Ltd., Publication



# THE IMS



# **THE IMS**

## **IP MULTIMEDIA CONCEPTS AND SERVICES, THIRD EDITION**

**Miikka Poikselkä**

*Nokia Siemens Networks, Finland*

**Georg Mayer**

*Nokia, Finland*



A John Wiley and Sons, Ltd., Publication

This edition first published 2009

© 2009 John Wiley & Sons Ltd

*Registered office*

John Wiley & Sons Ltd, The Atrium, Southern Gate, Chichester, West Sussex, PO19 8SQ, United Kingdom

For details of our global editorial offices, for customer services and for information about how to apply for permission to reuse the copyright material in this book please see our website at [www.wiley.com](http://www.wiley.com).

The right of the author to be identified as the author of this work has been asserted in accordance with the Copyright, Designs and Patents Act 1988.

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, except as permitted by the UK Copyright, Designs and Patents Act 1988, without the prior permission of the publisher.

Wiley also publishes its books in a variety of electronic formats. Some content that appears in print may not be available in electronic books.

Designations used by companies to distinguish their products are often claimed as trademarks. All brand names and product names used in this book are trade names, service marks, trademarks or registered trademarks of their respective owners. The publisher is not associated with any product or vendor mentioned in this book. This publication is designed to provide accurate and authoritative information in regard to the subject matter covered. It is sold on the understanding that the publisher is not engaged in rendering professional services. If professional advice or other expert assistance is required, the services of a competent professional should be sought.

***Library of Congress Cataloging-in-Publication Data***

Poikselka, Miikka.

The IMS : IP multimedia concepts and services / Miikka Poikselka, Georg Mayer. – 3rd ed.

p. cm.

Rev. ed. of: IMS / Miikka Poikselka ... [et al.]. 2006

Includes bibliographical references and index.

ISBN 978-0-470-72196-4 (cloth)

1. Multimedia communications. 2. Wireless communication systems. 3. Mobile communication systems. I. Mayer, Georg, 1970-. II. IMS. III. Title.

TK5105.15.P65 2008

621.382'12 – dc22

2008032207

***British Library Cataloguing in Publication Data***

A catalogue record for this book is available from the British Library

ISBN 978-0-470-72196-4

Typeset in 10/12 Times by Laserwords Private Limited, Chennai, India

Printed and bound in Great Britain by Antony Rowe Ltd, Chippenham, Wiltshire

# Contents

<b>Foreword</b>	<b>xv</b>
<b>Preface</b>	<b>xvii</b>
<b>Acknowledgements</b>	<b>xix</b>
<b>List of Figures</b>	<b>xxi</b>
<b>List of Tables</b>	<b>xxvii</b>

## PART I IMS ARCHITECTURE AND CONCEPTS

<b>1 Introduction</b>	<b>3</b>
1.1 What is the Internet Protocol Multimedia Subsystem (IMS)?	3
1.2 Fixed and Mobile Convergence	5
1.3 Example of IMS Services	7
1.4 Where did it come from?	9
1.4.1 3GPP Release 99 (3GPP R99)	9
1.4.2 3GPP Release 4	10
1.4.3 3GPP Releases 5 and 6	10
1.4.4 IMS Development in other Standardization Development Organizations	11
1.4.5 3GPP Release 7 and common IMS	12
1.4.6 Insight to 3GPP Release 8	13
1.5 Why a SIP Solution Based on 3GPP Standards?	13
<b>2 IP Multimedia Subsystem Architecture</b>	<b>15</b>
2.1 Architectural Requirements	15
2.1.1 IP Multimedia Sessions	15
2.1.2 IP Connectivity	16
2.1.3 Ensuring Quality of Service for IP Multimedia Services	17
2.1.4 IP Policy Control for Ensuring Correct Usage of Media Resources	17
2.1.5 Secure Communication	18
2.1.6 Charging Arrangements	18
2.1.7 Support of Roaming	19

2.1.8	<i>Interworking with Other Networks</i>	20
2.1.9	<i>Service Control Model</i>	20
2.1.10	<i>Layered Design and Access Independence</i>	21
2.2	Description of IMS-related Entities and Functionalities	22
2.2.1	<i>Call Session Control Functions (CSCF)</i>	23
2.2.2	<i>Emergency Call Session Control Function (E-CSCF)</i>	25
2.2.3	<i>Databases</i>	26
2.2.4	<i>Service Functions</i>	27
2.2.5	<i>IMS-CS Interworking Functions</i>	29
2.2.6	<i>Support Functions</i>	30
2.2.7	<i>Charging Entities</i>	32
2.2.8	<i>GPRS Entities</i>	32
2.3	IMS Reference Points	33
2.3.1	<i>Gm Reference Point</i>	33
2.3.2	<i>Mw Reference Point</i>	34
2.3.3	<i>IMS Service Control (ISC) Reference Point</i>	35
2.3.4	<i>Ma Reference Point</i>	35
2.3.5	<i>Cx Reference Point</i>	35
2.3.6	<i>Dx Reference Point</i>	38
2.3.7	<i>Sh Reference Point</i>	39
2.3.8	<i>Dh Reference Point</i>	40
2.3.9	<i>Si Reference Point</i>	40
2.3.10	<i>Mi Reference Point</i>	42
2.3.11	<i>Mj Reference Point</i>	43
2.3.12	<i>Mk Reference Point</i>	43
2.3.13	<i>Mg Reference Point</i>	43
2.3.14	<i>Mm Reference Point</i>	43
2.3.15	<i>Mr Reference Point</i>	43
2.3.16	<i>Mp Reference Point</i>	43
2.3.17	<i>Mn Reference Point</i>	44
2.3.18	<i>Gx Reference Point</i>	44
2.3.19	<i>Rx Reference Point</i>	45
2.3.20	<i>Charging Reference Points</i>	45
2.3.21	<i>Mx, Ix and Iq Reference Point</i>	45
2.3.22	<i>Ml Reference Point</i>	45
2.3.23	<i>Ut Reference Point</i>	46
3	<b>IMS Concepts</b>	<b>47</b>
3.1	Overview	47
3.2	Registration	48
3.3	Mechanism to Register Multiple User Identities at a Go	49
3.4	Session Initiation	50
3.5	Identification	51
3.5.1	<i>Public User Identity</i>	51
3.5.2	<i>Private User Identity</i>	52
3.5.3	<i>Relationship between Private and Public User Identities</i>	52
3.5.4	<i>Identity Generation Without ISIM</i>	53
3.5.5	<i>Identification of Services (Public Service Identities)</i>	55

3.5.6	<i>Identification of User's Device</i>	55
3.5.7	<i>Identification of Network Entities</i>	56
3.6	IP Multimedia Services Identity Module (ISIM)	57
3.7	Sharing a Single User Identity between Multiple Devices	57
3.8	Discovering the IMS Entry Point	58
3.9	S-CSCF Assignment	59
3.9.1	<i>S-CSCF Assignment during Registration</i>	60
3.9.2	<i>S-CSCF Assignment to Execute Services for an Unregistered User</i>	60
3.9.3	<i>S-CSCF Assignment in Error Cases</i>	61
3.9.4	<i>S-CSCF De-Assignment</i>	61
3.9.5	<i>Maintaining S-CSCF Assignment</i>	61
3.10	Mechanism for Controlling Bearer Traffic	61
3.10.1	<i>Introduction</i>	61
3.10.2	<i>Gating and QoS Control</i>	63
3.10.3	<i>Traffic Plane Event Reporting</i>	71
3.10.4	<i>Network Initiated Bearer Activation</i>	71
3.10.5	<i>Usage of Rx Reference Point</i>	73
3.11	Charging	75
3.11.1	<i>Introduction</i>	75
3.11.2	<i>Charging Architecture</i>	76
3.11.3	<i>Offline Charging</i>	77
3.11.4	<i>Online Charging</i>	79
3.11.5	<i>Flow-Based Charging</i>	80
3.11.6	<i>Charging Reference Points</i>	80
3.11.7	<i>Charging Information Correlation</i>	85
3.11.8	<i>Charging Information Distribution</i>	85
3.12	User Profile	86
3.12.1	<i>Introduction</i>	86
3.12.2	<i>Public Identification</i>	87
3.12.3	<i>Core Network Service Authorization</i>	88
3.12.4	<i>Service-Triggering Information</i>	89
3.13	Service Provision	90
3.13.1	<i>Introduction</i>	90
3.13.2	<i>Creation of Filter Criteria</i>	91
3.13.3	<i>Selection of AS</i>	93
3.13.4	<i>AS Behaviour</i>	93
3.14	Connectivity between Traditional CS Users and IMS Users	94
3.14.1	<i>Introduction</i>	94
3.14.2	<i>IMS-Originated Session Toward a User in the CS Core Network</i>	94
3.14.3	<i>CS-Originated Session Toward a User in IMS</i>	95
3.15	IMS Transit	96
3.16	Support for Local Dialling Plans	98
3.17	IMS Emergency Sessions	100
3.17.1	<i>Introduction and Architecture</i>	100
3.17.2	<i>Emergency Registration</i>	101
3.17.3	<i>Emergency Session Setup</i>	101
3.18	SIP Compression	102
3.18.1	<i>Introduction</i>	102
3.18.2	<i>SigComp Architecture</i>	103
3.18.3	<i>Compressing a SIP Message in IMS</i>	104

3.19	Combination of CS and IMS Services – Combinational Services	105
3.19.1	<i>Introduction</i>	105
3.19.2	<i>Capability Exchange</i>	105
3.19.3	<i>Parallel CS and IMS Services</i>	107
3.20	Voice Call Continuity	107
3.20.1	<i>Introduction</i>	107
3.20.2	<i>Voice Call Continuity Functionality</i>	108
3.20.3	<i>Voice Call Continuity Session Initiation and Termination</i>	108
3.20.4	<i>Voice Call Continuity Domain Transfer Procedure</i>	111
3.20.5	<i>Supplementary Services</i>	113
3.21	Security Services in the IMS	113
3.21.1	<i>IMS Security Model</i>	114
3.21.2	<i>Authentication</i>	114
3.21.3	<i>Network Domain Security (NDS)</i>	118
3.21.4	<i>IMS Access Security for SIP-Based Services</i>	121
3.21.5	<i>IMS Access Security for HTTP-Based Services</i>	125
3.22	Interworking between IPv4 and IPv6 in the IMS	126
3.22.1	<i>Introduction</i>	126
3.22.2	<i>Network Address Translation</i>	127
3.22.3	<i>IPv6-Only Versus Dual Stack</i>	132
3.22.4	<i>Interworking Scenarios</i>	133
3.22.5	<i>Intra-Domain Scenarios</i>	133
3.22.6	<i>Inter-Domain Scenarios</i>	133
3.22.7	<i>Configuration and Bootstrapping</i>	133
3.22.8	<i>IPv4-Only Access Networks</i>	134

**PART II IMS SERVICES**

<b>4</b>	<b>Presence</b>	<b>139</b>
4.1	Who will use the Presence Service?	139
4.2	Presence-Enhanced Services	140
4.3	Presence Contributing to Business	140
4.4	What is Presence?	141
4.5	Presence Service in IMS	142
4.6	Publishing Presence	144
4.7	Subscribing Presence	145
4.8	Watcher Information	147
4.9	Setting Presence Authorization	149
<b>5</b>	<b>Group Management</b>	<b>151</b>
5.1	Group Management's Contribution to Business	152
5.2	What is Group Management?	152
5.3	What is XML Configuration Access Protocol?	153
5.4	What is Common Policy?	153
5.4.1	<i>Model and Rule Structure</i>	154

5.4.2	<i>Data Types and Permission Processing</i>	155
5.5	Resource List	156
5.6	XCAP Usage for Resource Lists	156
5.7	Open Mobile Alliance Solution for Group Management	159
5.7.1	<i>Service Specific XML Document Management Servers</i>	159
5.7.2	<i>Shared XML Document Management Servers</i>	168
5.8	Multimedia Telephony and Service Management	171
5.8.1	<i>Communication Barring</i>	172
5.8.2	<i>Communication Diversion</i>	172
5.8.3	<i>Originating Identification Services</i>	172
5.8.4	<i>Terminating Identification Services</i>	173
5.8.5	<i>Multimedia Telephony Service Management Example</i>	173
<b>6</b>	<b>Push to Talk Over Cellular</b>	<b>175</b>
6.1	PoC Architecture	176
6.1.1	<i>PoC Server</i>	177
6.1.2	<i>PoC Client</i>	178
6.2	PoC Features	178
6.2.1	<i>PoC Communication</i>	178
6.2.2	<i>Simultaneous PoC Sessions</i>	180
6.2.3	<i>PoC Session Establishment Models</i>	181
6.2.4	<i>Incoming PoC Session Treatment</i>	183
6.2.5	<i>Instant Personal Alerts</i>	186
6.2.6	<i>Group Advertisement</i>	187
6.2.7	<i>Barring Features</i>	188
6.2.8	<i>Participant Information</i>	188
6.3	User Plane	189
6.3.1	<i>Talk Bursts</i>	189
6.3.2	<i>Talk Burst Control</i>	190
6.3.3	<i>Quality Feedback</i>	191
6.4	PoC Service Settings	192
<b>7</b>	<b>Messaging</b>	<b>195</b>
7.1	Overview of IMS Messaging	195
7.2	Immediate Messaging	195
7.3	Session-Based Messaging	197
7.4	Messaging Interworking	198
7.5	Instant Messaging by Open Mobile Alliance	201
7.5.1	<i>OMA IM Architecture</i>	202
7.5.2	<i>IM Communication</i>	203
7.5.3	<i>Conversation History</i>	211
7.5.4	<i>Deferred Messaging</i>	213
7.5.5	<i>IM Service Settings</i>	215
7.5.6	<i>IM User-Plane</i>	217
7.5.7	<i>Delivery Reports</i>	218

---

<b>8</b>	<b>Conferencing</b>	<b>221</b>
8.1	IMS Conferencing Architecture and Principles	221
8.1.1	<i>SIP Focus/Conferencing AS/MRFC</i>	221
8.1.2	<i>Conference Mixer–MRFP</i>	221
8.1.3	<i>Conference Participant</i>	222
8.1.4	<i>Conference Moderator, Floor Control and Conference Policy Control</i>	222
8.1.5	<i>Sidebars</i>	223
8.2	IMS Conferencing Procedures	223
8.2.1	<i>Conference Creation</i>	223
8.2.2	<i>Joining a Conference</i>	226
8.2.3	<i>Conference State Event Package</i>	228
8.2.4	<i>Floor Control</i>	230
<b>9</b>	<b>Multimedia Telephony</b>	<b>233</b>
9.1	Introduction	233
9.2	Multimedia Telephony Communication	234
9.2.1	<i>SIP and IMS Multimedia Telephony</i>	234
9.2.2	<i>IMS Communication Service Identification (ICSI) and Telephony Application Server (TAS)</i>	234
9.3	Supplementary Services	235
9.3.1	<i>Communication Barring</i>	235
9.3.2	<i>Communication Diversion</i>	236
9.3.3	<i>Communication Hold</i>	238
9.3.4	<i>Conference</i>	239
9.3.5	<i>Message Waiting</i>	240
9.3.6	<i>Originating Identification Presentation</i>	242
9.3.7	<i>Originating Identification Restriction</i>	243
9.3.8	<i>Terminating Identification Presentation (TIP)</i>	244
9.3.9	<i>Terminating Identification Restriction (TIR)</i>	244
9.3.10	<i>Explicit Communication Transfer</i>	245

## PART III DETAILED PROCEDURES

<b>10</b>	<b>Introduction to Detailed Procedures</b>	<b>249</b>
10.1	The Example Scenario	249
10.2	Base Standards	251
<b>11</b>	<b>An Example of IMS Registration</b>	<b>253</b>
11.1	Overview	253
11.2	Initial Parameters and IMS Management Object	255
11.3	Signalling PDP Context Establishment	256
11.4	P-CSCF Discovery	257
11.4.1	<i>Overview</i>	257
11.4.2	<i>SIP and DNS Server Configuration via DCHPv6</i>	258
11.4.3	<i>DNS Naming Authority Pointer (NAPTR) Resolving</i>	259
11.4.4	<i>Transport Protocol Selection and DNS Service (SRV) Resolving</i>	259
11.4.5	<i>DNS IPv6 Address Resolving</i>	260

11.4.6	<i>Related Standards</i>	260
11.5	SIP Registration and Registration Routing Aspects	260
11.5.1	<i>Overview</i>	260
11.5.2	<i>Constructing the REGISTER Request</i>	262
11.5.3	<i>From the UE to the P-CSCF</i>	263
11.5.4	<i>From the P-CSCF to the I-CSCF</i>	264
11.5.5	<i>From the I-CSCF to the S-CSCF</i>	264
11.5.6	<i>Registration at the S-CSCF</i>	266
11.5.7	<i>The 200 (OK) Response</i>	268
11.5.8	<i>The Service-Route Header</i>	269
11.5.9	<i>The Path Header</i>	270
11.5.10	<i>Third-Party Registration to Application Servers</i>	271
11.5.11	<i>Updating the User Profile</i>	272
11.5.12	<i>Related Standards</i>	273
11.6	Authentication	273
11.6.1	<i>Overview</i>	273
11.6.2	<i>HTTP Digest and 3GPP AKA</i>	275
11.6.3	<i>Authentication Information in the Initial REGISTER Request</i>	275
11.6.4	<i>S-CSCF Downloads the Authentication Vector (AV) from the HSS</i>	276
11.6.5	<i>S-CSCF Challenges the UE</i>	277
11.6.6	<i>UE's Response to the Challenge</i>	278
11.6.7	<i>Integrity Protection and Successful Authentication</i>	279
11.6.8	<i>Related Standards</i>	279
11.7	Access Security – IPsec SAs	279
11.7.1	<i>Overview</i>	279
11.7.2	<i>Establishing an SA During Initial Registration</i>	280
11.7.3	<i>Handling of Multiple Sets of SAs in the Case of Re-authentication</i>	282
11.7.4	<i>SA Lifetime</i>	284
11.7.5	<i>Port Setting and Routing</i>	285
11.7.6	<i>Related Standards</i>	288
11.8	SIP Security Mechanism Agreement	289
11.8.1	<i>Why the SIP Security Mechanism Agreement is Needed</i>	289
11.8.2	<i>Overview</i>	289
11.8.3	<i>Sip-Sec-Agree-Related Headers in the Initial REGISTER Request</i>	290
11.8.4	<i>The Security-Server Header in the 401 (Unauthorized) Response</i>	291
11.8.5	<i>Sip-Sec-Agree Headers in the Second REGISTER</i>	292
11.8.6	<i>Sip-Sec-Agree and Re-Registration</i>	292
11.8.7	<i>Related Standards</i>	294
11.9	IMS Communication Service Identification and other Callee Capabilities	294
11.9.1	<i>Overview</i>	294
11.9.2	<i>Feature Tags: Callee Capabilities</i>	295
11.9.3	<i>IMS Communication Service Identification (ICSI) and IMS Application Reference Identification (IARI)</i>	295
11.9.4	<i>Related Standards and Links</i>	297
11.10	Compression Negotiation	297
11.10.1	<i>Overview</i>	297
11.10.2	<i>Indicating willingness to use SigComp</i>	298
11.10.3	<i>comp=SigComp Parameter During Registration</i>	298
11.10.4	<i>comp=SigComp Parameter in Other Requests</i>	299
11.10.5	<i>Related Standards</i>	300

11.11	Access and Location Information	300
11.11.1	<i>P-Access-Network-Info</i>	300
11.11.2	<i>P-Visited-Network-ID</i>	300
11.11.3	<i>Related Standards</i>	301
11.12	Charging-Related Information During Registration	301
11.13	User Identities	301
11.13.1	<i>Overview</i>	301
11.13.2	<i>Public and Private User Identities for Registration</i>	302
11.13.3	<i>Identity Derivation from USIM</i>	303
11.13.4	<i>Default Public User Identity/P-Associated-URI Header</i>	303
11.13.5	<i>Assignment of a Globally Routable User Agent URI</i>	304
11.13.6	<i>UE's Subscription to Registration-State Information</i>	305
11.13.7	<i>P-CSCF's Subscription to Registration-State Information</i>	308
11.13.8	<i>Elements of Registration-State Information</i>	309
11.13.9	<i>Registration-State Information in the Body of the NOTIFY Request</i>	309
11.13.10	<i>Example Registration-State Information</i>	311
11.13.11	<i>Multiple Terminals and Registration-State Information</i>	315
11.13.12	<i>Related Standards</i>	316
11.14	Re-Registration and Re-Authentication	317
11.14.1	<i>User-initiated Re-registration</i>	317
11.14.2	<i>Network-Initiated Re-Authentication</i>	317
11.14.3	<i>Network-Initiated Re-Authentication Notification</i>	318
11.14.4	<i>Related Standards</i>	319
11.15	De-Registration	319
11.15.1	<i>Overview</i>	319
11.15.2	<i>User-Initiated De-Registration</i>	321
11.15.3	<i>Network-Initiated De-Registration</i>	324
11.15.4	<i>Related Standards</i>	326
11.16	GPRS-IMS-Bundled Authentication (GIBA)	326
11.16.1	<i>Example IMS Registration with Fallback to GIBA</i>	326
11.16.2	<i>GIBA Scenarios</i>	329
<b>12</b>	<b>An Example IMS Multimedia Telephony Session</b>	<b>331</b>
12.1	Overview	331
12.2	Caller and Callee Identities	333
12.2.1	<i>Overview</i>	333
12.2.2	<i>From and To Headers</i>	333
12.2.3	<i>Identification of the Calling User: P-Preferred-Identity and P-Asserted-Identity</i>	334
12.2.4	<i>Identification of the Called User</i>	335
12.2.5	<i>Related Standards</i>	337
12.3	Routing	337
12.3.1	<i>Overview</i>	337
12.3.2	<i>Session, Dialog, Transactions and Branch</i>	338
12.3.3	<i>Routing of the INVITE Request</i>	340
12.3.4	<i>Routing of the First Response</i>	345
12.3.5	<i>Re-transmission of the INVITE Request and the 100 (Trying) Response</i>	347
12.3.6	<i>Routing of Subsequent Requests in a Dialog</i>	347
12.3.7	<i>Standalone Transactions from One UE to Another</i>	349
12.3.8	<i>Routing to and from ASs</i>	349

12.3.9	<i>IMS Communication Service Identification</i>	352
12.3.10	<i>Related Standards</i>	360
12.4	Compression Negotiation	360
12.4.1	<i>Overview</i>	360
12.4.2	<i>Compression of the Initial Request</i>	360
12.4.3	<i>Compression of Responses</i>	361
12.4.4	<i>Compression of Subsequent Requests</i>	362
12.4.5	<i>Related Standards</i>	362
12.5	Media Negotiation	362
12.5.1	<i>Overview</i>	362
12.5.2	<i>Reliability of Provisional Responses</i>	364
12.5.3	<i>SDP Offer/Answer in IMS</i>	365
12.5.4	<i>Related Standards</i>	373
12.6	Resource Reservation	373
12.6.1	<i>Overview</i>	373
12.6.2	<i>The 183 (Session in Progress) Response</i>	374
12.6.3	<i>Are Preconditions Mandatorily Supported?</i>	374
12.6.4	<i>Preconditions</i>	376
12.6.5	<i>Establishing the Media Resources and PCC Related Actions</i>	381
12.6.6	<i>Media Policing</i>	382
12.6.7	<i>Related Standards</i>	383
12.7	Charging-Related Procedures During Session Establishment for Sessions	383
12.7.1	<i>Overview</i>	383
12.7.2	<i>Inter-Operator Identifier Exchange of ICID for a Media Session</i>	384
12.7.3	<i>Correlation of GCID and ICID</i>	385
12.7.4	<i>Distribution of Charging Function Addresses</i>	386
12.7.5	<i>Related Standards</i>	387
12.8	Release of a Session	387
12.8.1	<i>User-Initiated Session Release</i>	387
12.8.2	<i>P-CSCF Performing Network-Initiated Session Release</i>	388
12.8.3	<i>S-CSCF Performing Network-Initiated Session Release</i>	389
12.9	Alternative IMS Session Establishment Procedures	389
12.9.1	<i>Overview</i>	389
12.9.2	<i>Session with a Uni-Directional Media Stream and Available Resources on A Side</i>	390
12.9.3	<i>Session with a Uni-Directional Media Stream and Resources Need to be Reserved on A and B Side</i>	395
12.9.4	<i>Resources Available on B Side Only</i>	398
12.9.5	<i>Network Initiated Resource Reservation, Resources Available Only at A-Side</i>	400
12.9.6	<i>Network Initiated Resource Reservation at A Side</i>	404
12.9.7	<i>Resources Available on A Side and B Side</i>	406
12.9.8	<i>Early Media and Reliable Ring-Back Tone</i>	408
12.9.9	<i>Session Towards a Non-IMS SIP Terminal</i>	410
12.9.10	<i>Session From Non-IMS SIP Terminal</i>	414
12.9.11	<i>Related Standards</i>	415
12.10	Routing of GRUUs	415
12.10.1	<i>Theresa Registers her Laptop</i>	415
12.10.2	<i>REFER Request in Order to Transfer the Ongoing Call to Theresa's Laptop</i>	416
12.10.3	<i>Setting up the New Call to Theresa's Laptop</i>	417

12.11	Routing of PSIs	418
12.11.1	<i>Scenario 1: Routing From a User to a PSI</i>	418
12.11.2	<i>Scenario 2: Routing From a PSI to a User</i>	419
12.11.3	<i>Scenario 3: Routing From a PSI to Another PSI</i>	420
12.12	A Short Introduction to GPRS	420
12.12.1	<i>Overview</i>	420
12.12.2	<i>Packet Data Protocol (PDP)</i>	421
12.12.3	<i>PDP Context Types</i>	422
<b>13</b>	<b>An example IMS Voice Call Continuity Procedures</b>	<b>425</b>
13.1	Overview	425
13.2	Configuring the Clients with Communication Continuity Configuration Parameters	427
13.3	Setting up the Initial Call and Call Anchoring	429
13.3.1	<i>Tobias Sets up a CS Call Towards Theresa</i>	429
13.3.2	<i>Anchoring Decision and Routing the CS Call to the MGCF</i>	429
13.3.3	<i>Interworking the CS Call to IMS at the MGCF</i>	431
13.3.4	<i>Forwarding the IMS call to the VCC Application Server (resolving and direct routing of PSI)</i>	436
13.3.5	<i>Anchoring the Call in Tobias's Domain</i>	437
13.3.6	<i>Forwarding the Call to Theresa's Domain</i>	441
13.3.7	<i>Anchoring the call in Theresa's Domain</i>	447
13.3.8	<i>Delivering the Call to Theresa</i>	452
13.3.9	<i>Establishing the End-to-End Call</i>	453
13.3.10	<i>Scenario After Anchoring</i>	456
13.4	Domain Transfer: CS to IMS	457
13.4.1	<i>Tobias's Phone Invokes VCC Procedures</i>	457
13.4.2	<i>Routing to Tobias's VCC AS</i>	459
13.4.3	<i>Tobias's VCC AS Performs the CS to IMS Domain Transfer</i>	460
13.4.4	<i>Scenario after CS to IMS Domain Transfer</i>	462
13.5	Theresa adds Video to the Call	463
13.6	Domain Transfer: IMS to CS	465
13.6.1	<i>Theresa's Phone Starts VCC Procedures</i>	465
13.6.2	<i>From Theresa's Phone to Theresa's VCC AS</i>	466
13.6.3	<i>Performing the IMS to CS Domain Transfer</i>	466
13.6.4	<i>Scenario after IMS to CS Domain Transfer</i>	467
13.7	Related Standards	468
<b>References</b>	<b>471</b>	
<b>List of Abbreviations</b>	<b>477</b>	
<b>Index</b>	<b>487</b>	

# Foreword

The telecommunications industry is undergoing a fundamental change and the catalyst for this change is the business models and technologies of the Internet. The ubiquitous use of the Internet Protocol suite (IP) for voice, data, media and entertainment purposes, is driving the convergence of industries, services, networks and business models.

Network convergence is the route through which operators facilitate better access to end-user services and applications. IP provides a common foundation offering end-users seamless access to any service, any time, anywhere, and with any device. Full convergence is driven by enabling technologies such as HTTP/SIP, IPv6, VoIP, and the deployment of wireless broadband technologies such as WLAN, CDMA2000, and UMTS/HSPA.

The 3<sup>rd</sup> Generation Partnership Projects (3GPP and 3GPP2) have taken these developments into account whilst designing the IP-based Multimedia System (IMS). IMS is an overlay service provisioning platform through which telecommunications operators can utilise Internet technologies to their greatest advantage. It operates across fixed and mobile access technologies including WLAN, UMTS/HSPA, and DSL, along with many others.

The telecommunications industry has high expectations for IMS. This technology offers the prospect of new value chains and business models for operators on the one side, and the increase of the end-user experience through converged and blended services on the other.

This book provides a comprehensive overview of the IMS architecture, its concepts and interfaces, and is an excellent quick reference for IMS practitioners. It tackles questions such as: How can services be implemented with IMS? What are the procedures involved? What do typical call-flows look like?

The authors are recognized contributors to the development and standardization of IMS and, with the first commercial deployments of IMS occurring in various countries, their effort and commitment is starting to pay off.

*Mika Vehviläinen  
Chief Operating Officer  
Nokia Siemens Networks*



# Preface

Internet Protocol (IP) Multimedia Subsystem, better known as “IMS”, is based on the specification of Session Initiation Protocol (SIP) as standardized by Internet Engineering Task Force (IETF). But SIP as a protocol is only one part of it. IMS is more than just a protocol; it is an architecture for the convergence of data, speech, fixed and mobile networks and is based on a wide range of protocols, most of which have been developed by IETF. IMS combines and enhances them to allow real-time services on top of various kind of packet-switched technologies (GPRS, ADSL, WLAN, Cable, WiMAX, EPS).

This book was written to provide a detailed insight into what IMS is – i.e., its concepts, architecture, service and protocols. Its intended audience ranges from marketing managers, research engineers, development and test engineers to university students. The book is written in a manner that allows readers to choose the level of knowledge they need and the depth of understanding of IMS they desire to achieve. The book is also very well suited as a reference.

The first few chapters in Part I provide a detailed overview of the system architecture and the entities that, when combined, are necessary to provide IMS. These chapters also present the reference points (interfaces) between these entities and introduces the protocols assigned to these interfaces. This part ends with extensive description of essential IMS concepts such as registration, session establishment, policy and charging control, service provisioning, security, IP version interworking.

In IMS, services are not limited to audio, but also include presence, group management, Push to talk over Cellular, messaging, conferencing and IMS Multimedia Telephony. In Part II of this book, we introduce these advanced services in IMS, including call flows. This part proves that the convergence of services and networks is not a myth, but will have real added value for the user.

SIP and SDP are two of the main building blocks within IMS and their usage gets complemented by a large number of important extensions. Part III goes step by step through an example IMS registration and IMS Multimedia Telephony and Voice Call Continuity at the protocol level, detailing the procedures taken at every entity.

Third Generation Partnership Project (3GPP) and IETF have worked together during recent years in an amazing way to bring about IMS and the protocols used by it. We, the authors, have had the chance to participate in many technical discussions regarding the architecture and protocols and are still very active in further discussions on the ever-improving protocols and communication systems. Some of these discussions, which often can be described as debates or negotiations, frequently take a long time to conclude

and even more frequently do not result in an agreement or consensus on the technical solutions. We want to thank all the people in these standardization bodies as well as those in our own companies who have come up with ideas, have shown great patience and have worked hard to standardize this communication system of the future called IMS.

# Acknowledgements

The authors of this book would like to extend their thanks to colleagues working in 3GPP and IETF for their great efforts in creating the IMS specifications and related protocols. The authors would also like to give special thanks to the following who helped in the writing of this book providing excellent review comments and suggestions:

Erkki Koivusalo, Hannu Hietalahti, Peter Leis, Tao Haukka, Markku Tuohino, Juha Räsänen, Peter Vestergaard, Tapio Paavonen, Kalle Luukkainen, Pavel Dostal, Jozsef Varga, Martin Öttl, Thomas Belling, Ulrich Wiehe, Krisztian Kiss, Hans Rohnert, Antti Laurila and Adamu Haruna.

The authors want to especially give thanks to Hisham Khatabil and Aki Niemi for the very good team work and their excellent and major contributions during the first two editions, without which this book would not have been possible.

The authors welcome any comments and suggestions for improvements or changes that could be used to improve future editions of this book. Our e-mail addresses are:

[miikka.poikselka@nsn.com](mailto:miikka.poikselka@nsn.com)  
[georg.mayer@nokia.com](mailto:georg.mayer@nokia.com)



# List of Figures

<b>Figure 1.1</b>	IMS in converged networks	4
<b>Figure 1.2</b>	Convergence of networks	6
<b>Figure 1.3</b>	Multimedia messaging	8
<b>Figure 1.4</b>	The role of the IMS in the packet switched networks	9
<b>Figure 1.5</b>	Road to standardized common IMS standards	12
<b>Figure 2.1</b>	IMS connectivity options when a user is roaming	16
<b>Figure 2.2</b>	Overview of IMS security	18
<b>Figure 2.3</b>	IMS charging overview	19
<b>Figure 2.4</b>	IMS/CS roaming alternatives	20
<b>Figure 2.5</b>	IMS and layered architecture	21
<b>Figure 2.6</b>	Access independence	22
<b>Figure 2.7</b>	S-CSCF routing and basic IMS session setup	25
<b>Figure 2.8</b>	Structure of HSS	26
<b>Figure 2.9</b>	Relationship between different application server types	28
<b>Figure 2.10</b>	Signalling conversion in the SGW	29
<b>Figure 2.11</b>	Possible deployments for Interconnection Border Control Function	31
<b>Figure 2.12</b>	IMS architecture	33
<b>Figure 2.13</b>	HSS resolution using the SLF	39
<b>Figure 3.1</b>	High-level IMS registration flow	48
<b>Figure 3.2</b>	Example of implicit registration sets	49
<b>Figure 3.3</b>	High-level IMS session establishment flow	50
<b>Figure 3.4</b>	Relationship of user identities	53
<b>Figure 3.5</b>	Relationship between user identities including shared identity	54
<b>Figure 3.6</b>	Relationship between UE, GRUU and Public User Identities	56
<b>Figure 3.7</b>	Sharing a single user identity between multiple devices	58
<b>Figure 3.8</b>	A GPRS specific mechanism for discovering P-CSCF	59
<b>Figure 3.9</b>	A generic mechanism for discovering P-CSCF	59

<b>Figure 3.10</b>	Example of S-CSCF assignment	61
<b>Figure 3.11</b>	Policy control entities	62
<b>Figure 3.12</b>	Bearer authorization in UE initiated model	64
<b>Figure 3.13</b>	Example of IMS based gating in the Access Gateway	70
<b>Figure 3.14</b>	Subscription to IMS signaling bearer status	72
<b>Figure 3.15</b>	Bearer authorization in network initiated model	73
<b>Figure 3.16</b>	IMS charging architecture	77
<b>Figure 3.17</b>	Example of offline charging	79
<b>Figure 3.18</b>	Session- and event-based offline charging example	82
<b>Figure 3.19</b>	Session- and event-based online charging example	83
<b>Figure 3.20</b>	IMS charging correlation	86
<b>Figure 3.21</b>	Distribution of charging information	87
<b>Figure 3.22</b>	Structure of IMS user profile	88
<b>Figure 3.23</b>	Media authorization in S-CSCF	89
<b>Figure 3.24</b>	Shared initial filter criteria	89
<b>Figure 3.25</b>	Structure of initial filter criteria	90
<b>Figure 3.26</b>	Structure of service point trigger	91
<b>Figure 3.27</b>	IMS-CS interworking configuration when an IMS user calls a CS user	95
<b>Figure 3.28</b>	IMS-CS interworking configuration when a CS user calls an IMS user	96
<b>Figure 3.29</b>	IMS transit solution for PSTN/ISDN	96
<b>Figure 3.30</b>	IMS as a general transit network	97
<b>Figure 3.31</b>	Derivation rules for local dialing plans	98
<b>Figure 3.32</b>	IMS emergency session setup	101
<b>Figure 3.33</b>	Signalling compression architecture	104
<b>Figure 3.34</b>	Capability exchange during an ongoing CS call	106
<b>Figure 3.35</b>	Example for parallel connections when combining IMS and CS services	107
<b>Figure 3.36</b>	Voice call continuity and IMS originated call	109
<b>Figure 3.37</b>	Voice call continuity and CS originated call	109
<b>Figure 3.38</b>	Voice call continuity and terminated call	110
<b>Figure 3.39</b>	Domain transfer from CS to IMS	112
<b>Figure 3.40</b>	Domain transfer from IMS to CS	113
<b>Figure 3.41</b>	Security architecture of the IMS	115
<b>Figure 3.42</b>	NASS bundled authentication	117
<b>Figure 3.43</b>	Security domains in the IMS	119

<b>Figure 3.44</b>	NDS/IP and SEGs	121
<b>Figure 3.45</b>	Generic bootstrapping architecture	125
<b>Figure 3.46</b>	Application layer gateway in IMS	128
<b>Figure 3.47</b>	Routing based on SIP Outbound flows	130
<b>Figure 3.48</b>	UE discovers reflexive and relayed addresses via STUN/TURN	131
<b>Figure 3.49</b>	Simplified STUN/TURN/ICE flow	132
<b>Figure 3.50</b>	End-to-end and interconnection scenarios	134
<b>Figure 3.51</b>	IPv6 to IPv4 tunnelling mechanism	135
<b>Figure 4.1</b>	Dynamic presence	140
<b>Figure 4.2</b>	Examples of enhanced presence service	141
<b>Figure 4.3</b>	Overview of presence	142
<b>Figure 4.4</b>	Presence architecture	143
<b>Figure 4.5</b>	Presence publication	145
<b>Figure 4.6</b>	Subscription to presence information	146
<b>Figure 4.7</b>	Subscription to watcher information	148
<b>Figure 5.1</b>	XCAP operations	154
<b>Figure 5.2</b>	Common policy data model	155
<b>Figure 5.3</b>	Presence subscription example flow, no RLS	157
<b>Figure 5.4</b>	Presence subscription example flow, with RLS	158
<b>Figure 5.5</b>	Example resource list flow	158
<b>Figure 5.6</b>	OMA XDM architecture	160
<b>Figure 5.7</b>	Storing conversation history metadata and retrieving it	166
<b>Figure 6.1</b>	Push to talk over cellular	176
<b>Figure 6.2</b>	Voice call versus push to talk over cellular	176
<b>Figure 6.3</b>	Push to talk over cellular architecture	177
<b>Figure 6.4</b>	PoC server architecture	178
<b>Figure 6.5</b>	Different PoC communication models	180
<b>Figure 6.6</b>	Pre-established PoC session setup	182
<b>Figure 6.7</b>	On-demand PoC session setup using an unconfirmed mode in the terminating network	183
<b>Figure 6.8</b>	Incoming session treatment decision tree showing impact of access control list and user's answer mode	186
<b>Figure 6.9</b>	User plane Protocol entities	189
<b>Figure 6.10</b>	RTP control Protocol APP packet format	191
<b>Figure 7.1</b>	Instant messaging types	196
<b>Figure 7.2</b>	Immediate messaging flow	196
<b>Figure 7.3</b>	Session-based messaging flow	197

<b>Figure 7.4</b>	Example of terminating SMS over IP	199
<b>Figure 7.5</b>	Example of originating SMS over IP	200
<b>Figure 7.6</b>	OMA IM architecture	202
<b>Figure 7.7</b>	OMA IM server architecture	204
<b>Figure 7.8</b>	Originating immediate message in OMA IM	205
<b>Figure 7.9</b>	Terminating immediate message in OMA IM	205
<b>Figure 7.10</b>	Large message mode in OMA IM	206
<b>Figure 7.11</b>	Different IM session types	207
<b>Figure 7.12</b>	OMA IM session initiation	208
<b>Figure 7.13</b>	OMA IM session termination	209
<b>Figure 7.14</b>	Conversation history function	212
<b>Figure 7.15</b>	Store and forward functionality for IM users	214
<b>Figure 7.16</b>	OMA IM user plane	217
<b>Figure 7.17</b>	OMA IM user plane for deferred messaging and conversation history	218
<b>Figure 8.1</b>	IMS conferencing architecture	222
<b>Figure 8.2</b>	Ad-hoc conference creation	224
<b>Figure 8.3</b>	User calling into a conference	226
<b>Figure 8.4</b>	Referring users into a conference via conference AS/MRFC	228
<b>Figure 8.5</b>	Floor control with BFCP	231
<b>Figure 9.1</b>	Example of incoming communication barring supplementary service	236
<b>Figure 9.2</b>	Example of outgoing communication barring supplementary service	237
<b>Figure 9.3</b>	Example of communication diversion supplementary service	238
<b>Figure 9.4</b>	Example of communication hold supplementary service	239
<b>Figure 9.5</b>	Example of conference supplementary service	240
<b>Figure 9.6</b>	Example of explicit call transfer	245
<b>Figure 10.1</b>	The example scenario	250
<b>Figure 11.1</b>	Initial registration flow	254
<b>Figure 11.2</b>	Discovering the P-CSCF via DHCP/DNS	257
<b>Figure 11.3</b>	Routing during registration	270
<b>Figure 11.4</b>	Third party register by S-CSCF	271
<b>Figure 11.5</b>	Authentication information flows during IMS registration	274
<b>Figure 11.6</b>	SA establishment during initial registration	281
<b>Figure 11.7</b>	Two sets of SAs during re-authentication	283
<b>Figure 11.8</b>	Taking a new set of SAs into use and dropping an old set of SAs	284
<b>Figure 11.9</b>	Request and response routing between UE and P-CSCF over UDP	288
<b>Figure 11.10</b>	Request and response routing between UE and P-CSCF over TCP	288

<b>Figure 11.11</b> Sip-Sec-Agree during initial registration	294
<b>Figure 11.12</b> Tobias's subscription to his registration-state information	307
<b>Figure 11.13</b> P-CSCF subscription to Tobias's registration-state information	308
<b>Figure 11.14</b> User-initiated re-registration (without re-authentication)	314
<b>Figure 11.15</b> Network-initiated re-authentication	315
<b>Figure 11.16</b> User-initiated de-registration	320
<b>Figure 11.17</b> Network-initiated de-registration	320
<b>Figure 11.18</b> Example early IMS security flow	329
<b>Figure 12.1</b> IMS session establishment call flow	332
<b>Figure 12.2</b> Routing an initial INVITE request and its responses	339
<b>Figure 12.3</b> Routing of subsequent requests and their responses	348
<b>Figure 12.4</b> Routing to an application server	351
<b>Figure 12.5</b> Registration of feature tags	354
<b>Figure 12.6</b> Routing based on caller preferences	356
<b>Figure 12.7</b> Routing based on caller preferences: require	358
<b>Figure 12.8</b> Routing based on caller preferences: explicit	359
<b>Figure 12.9</b> Routing based on caller preferences: require; explicit	359
<b>Figure 12.10</b> SDP offer/answer in IMS	363
<b>Figure 12.11</b> SIP, SDP offer/answer and preconditions during session establishment	375
<b>Figure 12.12</b> SIP session establishment without preconditions	376
<b>Figure 12.13</b> Media streams and transport in the example scenario	380
<b>Figure 12.14</b> Worst case scenario for media policing	382
<b>Figure 12.15</b> Theresa releases the session	388
<b>Figure 12.16</b> P-CSCF terminates a session	388
<b>Figure 12.17</b> S-CSCF terminates a session	389
<b>Figure 12.18</b> Session establishment – resources available at A side	391
<b>Figure 12.19</b> Session establishment – uni-directional stream with resource reservation on both sides	397
<b>Figure 12.20</b> Session establishment – resources available at B side	399
<b>Figure 12.21</b> Session establishment – network initiated resources at B side	401
<b>Figure 12.22</b> Session establishment – network initiated resources at A side	404
<b>Figure 12.23</b> Session establishment – resources available on both sides	407
<b>Figure 12.24</b> Session establishment – early media and ringback tones	409
<b>Figure 12.25</b> Session establishment towards a non-IMS terminal	411
<b>Figure 12.26</b> Session establishment from a non-IMS terminal	414
<b>Figure 12.27</b> Routing of GRUU	416

<b>Figure 12.28</b> Routing from a user to a PSI	419
<b>Figure 12.29</b> Routing from a PSI to a user	419
<b>Figure 12.30</b> Routing from an AS to a PSI	420
<b>Figure 12.31</b> PDP context types	422
<b>Figure 13.1</b> Basic interworking of CS and IMS calls at MGCF	430
<b>Figure 13.2</b> Basic dialog mapping at Tobias's VCC AS (DTF), acting as a SIP B2BUA	431
<b>Figure 13.3</b> VCC Anchoring – simplified call flow	440
<b>Figure 13.4</b> VCC – connections after anchoring	457
<b>Figure 13.5</b> VCC – connections After CS to PS Domain Transfer (A-Side)	463
<b>Figure 13.6</b> VCC – connections After PS to CS Domain Transfer (B-Side)	468

# List of Tables

<b>Table 2.1</b>	Cx commands	36
<b>Table 2.2</b>	Sh commands	39
<b>Table 2.3</b>	Summary of reference points	41
<b>Table 3.1</b>	Information in the PCRF#1	66
<b>Table 3.2</b>	IP QoS class mapping to UMTS QoS	67
<b>Table 3.3</b>	The maximum data rates and QoS class in the PCRF#1	67
<b>Table 3.4</b>	Requested QoS parameters	68
<b>Table 3.5</b>	The maximum authorized traffic class per media type in the UE	69
<b>Table 3.6</b>	The values of the maximum authorized UMTS QoS parameters as calculated by UE #1 (Tobias) from the example	69
<b>Table 3.7</b>	The values of the maximum authorized UMTS QoS parameters as calculated by UE #1 from the example	69
<b>Table 3.8</b>	Rx commands	74
<b>Table 3.9</b>	Summary of offline charging functions	78
<b>Table 3.10</b>	Examples of local dialling strings	100
<b>Table 3.11</b>	Authentication and key agreement parameters	116
<b>Table 6.1</b>	PoC server functional distribution	179
<b>Table 6.2</b>	Summary of different PoC session setup combinations	184
<b>Table 6.3</b>	Mapping of subtype bit patterns to TBCP Protocol messages	192
<b>Table 7.1</b>	OMA IM service settings and possible values	216
<b>Table 10.1</b>	Location of CSCFs and GPRS access for the example scenario	250
<b>Table 11.1</b>	Routing-related headers	261
<b>Table 11.2</b>	Filter criteria in Tobias's S-CSCF	271
<b>Table 11.3</b>	Tobias's public user identities	301
<b>Table 11.4</b>	GIBA registration scenarios	328

<b>Table 12.1</b>	Filter criteria in Tobias's S-CSCF	349
<b>Table 13.1</b>	VCC Related Telephone Numbers and Addresses	427
<b>Table 13.2</b>	VCC Related Routing Numbers and SIP Addresses	427
<b>Table 13.3</b>	SIP dialogs at Tobias's VCC AS (B2BUA)	442
<b>Table 13.4</b>	SIP dialogs at Theresa's VCC AS (B2BUA)	450

# Part I

## IMS Architecture and Concepts



# 1

## Introduction

### 1.1 What is the Internet Protocol Multimedia Subsystem (IMS)?

Fixed and mobile networks have gone through a major transition in the past 20 years. In the mobile world, first-generation (1G) systems were introduced in the mid-1980s. These networks offered basic services for users. The main emphasis was on speech and speech-related services. Second-generation (2G) systems in the 1990s brought some data services and more sophisticated supplementary services to the users. The third generation (3G and 3.5G) and its evolution (LTE) is now enabling faster data rates and various multi-media services. In the fixed side, traditional Public Switched Telephone Network (PSTN) and Integrated Services Digital Network (ISDN) networks have dominated traditional voice and video communication. In recent years the usage of the Internet has exploded and more and more users are taking advantage of faster and cheaper Internet connection such as Asymmetric Digital Subscriber Line (ADSL). These types of Internet connections enable always-on connectivity, which is a necessity for people to start using real-time communication means – e.g., chatting applications, online gaming, Voice over IP (VoIP).

At the moment we are experiencing the fast convergence of fixed and mobile worlds as the penetration of mobile devices is increasing on a yearly basis. These mobile devices have large, high-precision displays, they have built-in cameras and a lot of resources for applications. They are always-on always-connected application devices. This redefines applications. Applications are no longer isolated entities exchanging information only with the user interface. The next generation of more exciting applications are peer-to-peer entities, which facilitate sharing: shared browsing, shared whiteboard, shared game experience, shared two-way radio session (i.e., Push to Talk Over Cellular). The concept of being connected will be redefined. Dialling a number and talking will soon be seen as a narrow subset of networking. The ability to establish a peer-to-peer connection between the new Internet Protocol (IP) enabled devices is the key required ingredient. This new paradigm of communications reaches far beyond the capabilities of the Plain Old Telephone Service (POTS).

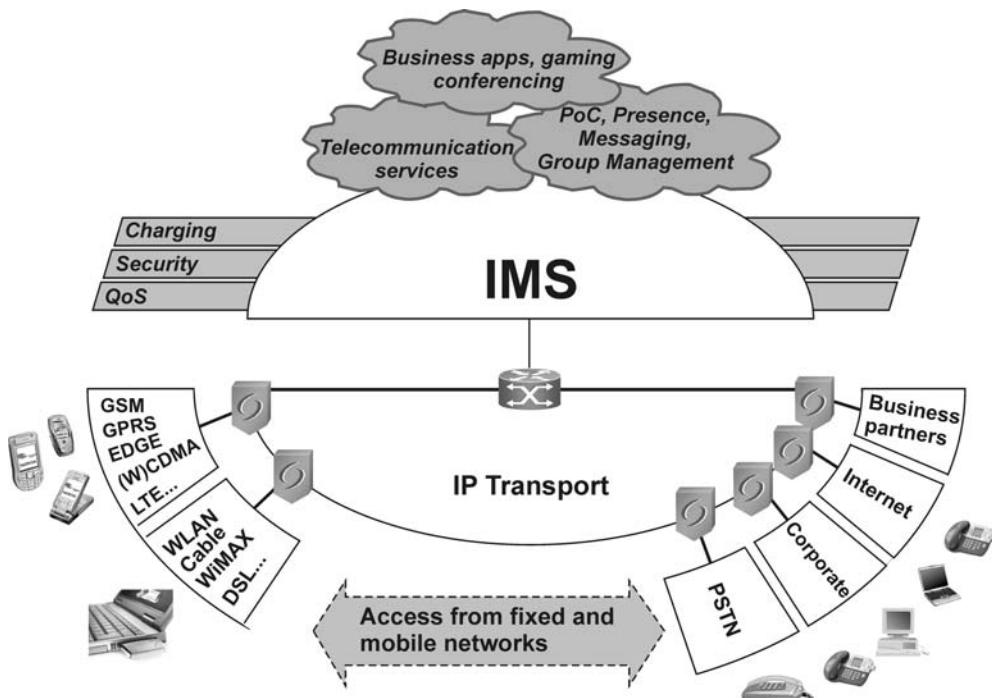
In order to communicate, IP-based applications must have a mechanism to reach the correspondent. The telephone network currently provides this critical task of establishing a connection. By dialling the peer, the network can establish an ad hoc connection

between any two terminals over the IP network. This critical IP connectivity capability is offered only in isolated and single-service provider environments in the Internet; closed systems compete on user base, where user lock-in is key and interworking between service providers is an unwelcome feature. Therefore, we need a global system – the IP Multimedia Subsystem (IMS). It allows applications in IP-enabled devices to establish peer-to-peer and peer-to-content connections easily and securely. Our definition for the IMS is:

**IMS is a global, access-independent and standard-based IP connectivity and service control architecture that enables various types of multimedia services to end-users using common Internet-based protocols.**

True integration of voice and data services increases productivity and overall effectiveness, while the development of innovative applications integrating voice, data and multimedia will create demands for new services, such as presence, multimedia chat, push to talk and conferencing. The skill to combine mobility and the IP network will be crucial to service success in the future.

Figure 1.1 shows a converged communication network for the fixed mobile environment. It is the IMS which introduces multimedia session control in the packet-switched domain and at the same time brings circuit-switched functionality in the packet-switched domain. The IMS is a key technology for such network consolidation.



**Figure 1.1** IMS in converged networks

## 1.2 Fixed and Mobile Convergence

Since the IMS architecture integrates both wireless and wireline networks, the IMS becomes an inexpensive medium for Fixed to Mobile Convergence (FMC). It is currently one of the crucial strategic issues in the telecommunications industry. Trends in different regions and countries are different, but on a global level operators are facing increasing competition and declining prices for voice traffic, fixed lines and fixed minutes. At the same time, mobile voice traffic is growing rapidly and substituting that of voice traffic over fixed lines. End users now expect high quality with reliable mobility and are using the Internet more as the penetration of broadband grows rapidly. Now, Voice over IP (VoIP) is starting to substitute PSTN. Meanwhile, key enabling technologies, such as smart phones, wireline and wireless broadband and IMS for seamless service over different access types are readily available. Combined, this means that operators are looking for long-term evolutionary strategies towards converged, access-agnostic networks, with service integration and interoperability across domains and devices. From the end user's perspective this delivers seamless end user experience across multiple locations, devices and services. Convergence can be viewed from three separate angles:

- convergence of networks
- convergence of services
- convergence of devices

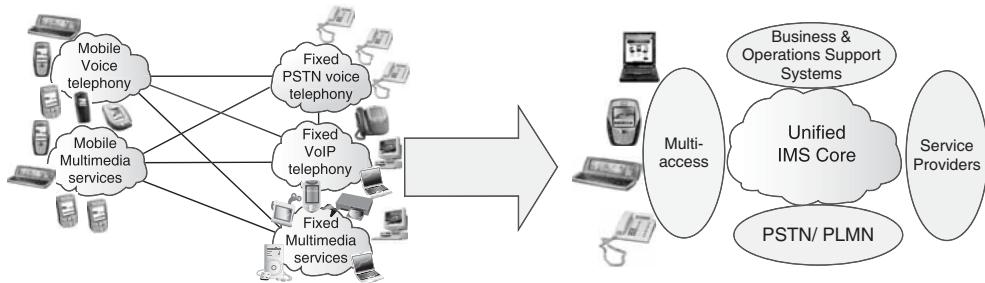
### *Convergence of Networks*

Network convergence simplifies the end user experience and dissolves the barriers and complexities that separate today's network islands. The same services are available across all networks and, in an ideal world, appear and perform in exactly the same way, making usage easy, transparent and intuitive.

From an operator's perspective, the goal of network convergence is to migrate today's separate PSTN, PLMN, backbone and IP networks to a fully converged network that supports any access technology. The full evolution includes a cost effective migration to an All-IP network using IMS as the unifying platform, allowing all new services to be accessed in a standard and consistent manner as shown in Figure 1.2 manner. Advancing in this evolution will be the key to an operator's ability to reduce OPEX and CAPEX, and increasing competitiveness and profitability.

Many locations, such as homes, enterprises and public places already have access networks available (xDSL, WLAN, cable etc.). When operators launch new services such as video streaming or hosted email they can take advantage of these existing networks, extending service access to more potential subscribers. In turn this will mean launching services to new market segments for new revenue opportunities. With multiple access networks operators can attract existing and new customers with an enhanced convergence service portfolio using unified billing.

A converged core network is the key enabler for converged networks. Multi-access to a common, converged core network enables cost optimization for both mobile and hybrid operators. Re-use of existing access network infrastructure and integration with the service infrastructure results in both OPEX and CAPEX savings. And multi-access enables operators to introduce end-to-end quadruple-play services (voice, data, video/TV and mobility), to new customers.



**Figure 1.2** Convergence of networks

IP-based access connection using the SIP protocol between the device and the converged core network – so called ‘Native IP access’ – allows voice, video and other multimedia applications over any access network. Native IP access supports a wide variety of applications in different devices, including mobile handsets, PC clients and SIP desktop phones. POTS phones too, can also be supported, via a connection to an SIP-capable DSLAM or analog terminal adapter (ATA). Native IP access architecture allows the introduction of new rich IP multimedia services through IMS functionality, such as presence, media push, multimedia telephony, games and various other SIP enabled applications, furthering revenue streams for operators.

### *Device Convergence*

Typically, a device is only used – in the main – for a single purpose and the support for its other functions is limited. PSTN phones, low end mobile phones and set-top boxes are good examples. Consumers use these devices for a single purpose. When they change tasks they change device and access network. This means service islands, which lead to mis-matched user experiences from different public and private networks. What’s needed are unifying devices that can access services in a similar and easy way.

Smart phones are serious contenders for voice-plus multimedia services in a truly mobile environment. Multiple radio interfaces provide access over circuit and packet-switched networks (cellular, WLAN etc) and IMS allows services and applications to traverse different IP networks. Mobile phone development has been rapid in the last decade and new models take increasing advantage of new technologies. They incorporate the enhanced colour displays and high quality imaging features needed to support service consumption and the creation of own content. Plus the exponential growth of memory capacity and processing power means that smart phones can now replicate the applications currently employed in notebook PCs and PDAs.

Consumers want the quality of fixed services with the flexibility of mobile and convergence lets this happen, by allowing service access through the most suitable access network, and by letting consumers choose the best device for the service. In many cases that device will be a smart phone, but it could just as easily be a PC or laptop with VoIP software or converged fixed clients who can share IM, presence etc with mobile devices, a fixed VoIP phone or even a TV with a set-top box.

### *Service Convergence*

The mobility model has become ‘me-centric’, with my phone book, my contact information, my agenda, my messages, my availability and preferred communication method, my Internet, my pictures and video clips (received and shared), my personal and business email, my wall-paper, my music and so on. Multimedia services, such as Presence, Push-to-talk, messaging, interactive applications, data or video sharing plus streaming, browsing and downloading, are being delivered over fixed and mobile packet networks. To launch new services and applications quickly, operators can use IMS to eliminate the complexity of different service platforms in the network. Standards based Service Delivery Framework (SDF) provides comprehensive lifecycle management, making the launch of new services and applications quicker and easier to integrate and operate; delivering solutions more speedily to market and reducing the total cost of ownership. In effect the operator can provision – and the end-user quickly and conveniently self-provision – the new services.

VoIP and Instant Messaging are two developments that helped kick-start service convergence. VoIP has had a seismic impact on telephony within enterprises and, as the penetration of broadband access increases, so does the availability of this transport mechanism within the home. Users also benefit from personalized VoIP, including same number, same contacts and the same supplementary services like call barring, call waiting, ring back tones, one voice mail, option for one postpaid bill or prepaid account, etc through any access network. IP DSLAMs are letting operators offer both DSL access and traditional two-wire POTS connections using a SIP client in the DSLAM. This development and others like fixed VoIP phones, Analog Telephony Adapters (ATA) and fixed soft switches place fixed line operators in an excellent position. They can offer multimedia services via DSL and attractive tariffs for analog POTS connected to an IP network, thereby maintaining existing services where required and evolving the core network to an IP-based solution. Smart phones, on the other hand, have WLAN interfaces so they can access fixed broadband networks. This allows the mobile phone to be used as an IP phone and users to continue employing their personalized services at home, or via WLANs, connected to DSL, in hot spots or offices. Convergence in this case enables a practical combination of cellular and fixed broadband access. The user experience doesn’t change: the same voice and multimedia services are used in the same way. Fixed to Mobile Substitution and fixed VoIP are gradually replacing PSTN voice telephony. Multimedia services are being delivered over fixed and mobile packet networks. Operators must now decide on the kinds of services they wish to provide by themselves or by partners, to whom and in which regions. And what they might offer is no longer limited to traditional telecom services only, but perhaps entry into new businesses.

### **1.3 Example of IMS Services**

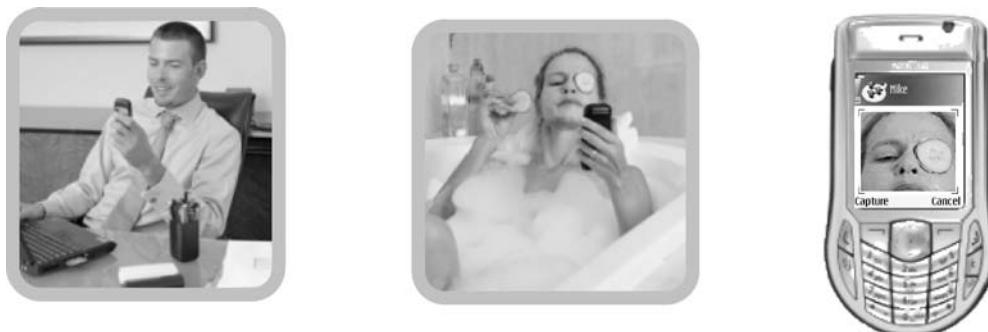
Switching on my Internet Protocol Multimedia Subsystem (IMS) enabled device, it will automatically register to the IMS network using information in the identity module (such as USIM). During registration both device and network are authenticated and my device will get my user identities from the network. After this single registration, all my services will be available, including push to talk, presence, voice and video sessions, messaging

and multiplayer games. Moreover, my availability information is updated at the presence server as being “online” and listing my current applications.

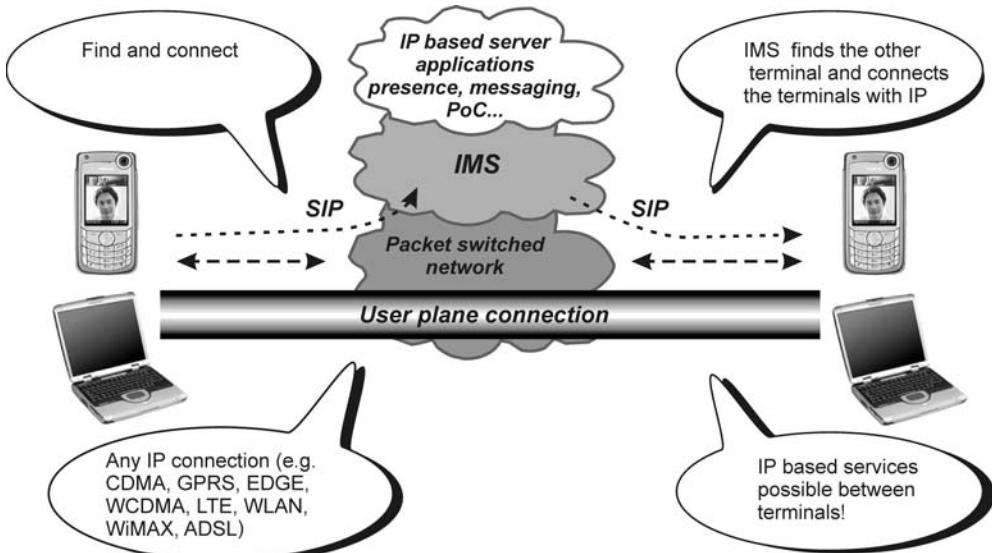
When I need to contact my friend Bob, I select Bob from my device’s phone book and, based on his presence information, I see immediately that he is available. After pressing the ‘green button’ on my device it will place an ‘ordinary’ call to him. The IMS network will take care of finding and setting up a Session Initiation Protocol (SIP) session between our devices, even though Bob is currently abroad. When my call reaches Bob’s terminal he will see that the call is coming from me and, additionally, he sees a text string inserted by me (‘Free tickets to movie next Wednesday’). Bob answers, but tells me that he’s not sure whether he is able to come. We decide to check the issue again on Sunday. Before hanging up, Bob says to me, ‘You won’t believe what I saw today but just wait a second, I’ll show you.’ Bob starts streaming a video clip to me, and while I’m watching the video, Bob keeps explaining what happened in the zoo earlier that day.

Mike realizes that today is the birthday of his good friend Jill. Although he’s travelling and can’t meet her today, he wants to send Jill a personal birthday message. While Mike is sitting in a local coffee shop enjoying coffee and reading the latest news from the Internet using his brand-new Wireless Local Area Network (WLAN) device, he decides to send her a video clip as a birthday greeting. Jill is having a bath when she hears her phone ringing. She sees that she has received a message and checks it. She saves the video clip and decides to send something in return. Knowing that Mike knows her weird sense of humour, she sends a picture of herself taking a bath (Figure 1.3).

Peter Simpson is a Londoner and a die-hard Arsenal fan. With sheer luck he has managed to get tickets to an Arsenal–Tottenham derby and sets off to see the game. There he is, sitting at the stadium during the match, when suddenly he gets an irresistible urge to make his friend envious. He gets his mobile phone and makes a call to his friend John Clark, a Tottenham supporter. John is sitting at his desk and receives an incoming call pop-up on his PC screen, informing him that Peter is calling. He answers and they start to talk. Peter can’t contain himself and starts the video-sharing application while zooming onto the field. John receives an incoming video request and accepts the stream. The PC client starts to show the game, and with a pang of jealousy and disappointment John watches Arsenal score. ‘Nice goal, huh?’ asks Peter. ‘It ain’t over yet,’ says John, gritting his teeth, and ends the video stream. They continue to argue good-naturedly about the game and their teams over the phone.



**Figure 1.3** Multimedia messaging



**Figure 1.4** The role of the IMS in the packet switched networks

All the required communication takes place using the IP connectivity provided by the IMS as shown in Figure 1.4. The IMS offers the capability to select the best and most suitable communication media, to change the media during the session spontaneously, and use the preferred (SIP-capable) communication device over any IP access.

## 1.4 Where did it come from?

The European Telecommunications Standards Institute (ETSI) was the standardization organization that defined the Global System for Mobile Communications (GSM) during the late 1980s and 1990s. ETSI also defined the General Packet Radio Service (GPRS) network architecture. The last GSM-only standard was produced in 1998, and in the same year the 3GPP was founded by standardization bodies from Europe, Japan, South Korea, the USA and China to specify a 3G mobile system comprising Wideband Code Division Multiple Access (WCDMA) and Time Division/Code Division Multiple Access (TD-CDMA) radio access and an evolved GSM core network ([www.3gpp.org/About/3gppagre.pdf](http://www.3gpp.org/About/3gppagre.pdf)). Most of the work and cornerstone specifications were inherited from the ETSI Special Mobile Group (SMG). The 3GPP originally decided to prepare specifications on a yearly basis, the first specification release being Release 99.

### 1.4.1 3GPP Release 99 (3GPP R99)

It took barely a year to produce the first release – Release 1999. The functionality of the release was frozen in December 1999 although some base specifications were frozen afterward – in March 2001. Fast completion was possible because the actual work was divided between two organizations: 3GPP and ETSI SMG. 3GPP developed the services, system architecture, WCDMA and TD-CDMA radio accesses, and the common core

network. ETSI SMG developed the GSM/Enhanced Data Rates for Global Evolution (EDGE) radio access.

WCDMA radio access was the most significant enhancement to the GSM-based 3G system in Release 1999. In addition to WCDMA, UMTS Terrestrial Radio Access Network (UTRAN) introduced the Iu interface as well. Compared with the A and Gb interfaces, there are two significant differences. First, speech transcoding for Iu is performed in the core network. In the GSM it was logically a Base Transceiver Station (BTS) functionality. Second, encryption and cell-level mobility management for Iu are done in the Radio Network Controller (RNC). In GSM they were done in the Serving GPRS Support Node (SGSN) for GPRS services.

The Open Service Architecture (OSA) was introduced for service creation. On the service side the target was to stop standardizing new services and to concentrate on service capabilities, such as toolkits (CAMEL, SIM Application Toolkit and OSA). This principle was followed quite well, even though the Virtual Home Environment (VHE), an umbrella concept that covers all service creation, still lacks a good definition.

#### 1.4.2 3GPP Release 4

After Release 1999, 3GPP started to specify Release 2000, including the so-called All-IP that was later renamed as the IMS. During 2000 it was realized that the development of IMS could not be completed during the year. Therefore, Release 2000 was split into Release 4 and Release 5.

It was decided that Release 4 would be completed without the IMS. The most significant new functionalities in 3GPP Release 4 were: the Mobile Switching Centre (MSC) Server–Media Gateway (MGW) concept, IP transport of core network protocols, Location Services (LCS) enhancements for UTRAN and multimedia messaging and IP transport for the Gb user plane.

3GPP Release 4 was functionally frozen and officially completed in March 2001. The backward compatibility requirement for changes, essential for the radio interface, was enforced as late as September 2002.

#### 1.4.3 3GPP Releases 5 and 6

Release 5 finally introduced the IMS as part of 3GPP specifications. The IMS is supposed to be a standardized access-independent IP-based architecture that interworks with existing voice and data networks for both fixed (e.g., PSTN, ISDN, Internet) and mobile users (e.g., GSM, CDMA). The IMS architecture makes it possible to establish peer-to-peer IP communications with all types of clients with the requisite quality of services. In addition to session management, the IMS architecture also addresses functionalities that are necessary for complete service delivery (e.g., registration, security, charging, bearer control, roaming). All in all, the IMS will form the heart of the IP core network.

The content of Release 5 was heavily discussed and, finally, the functional content of 3GPP Release 5 was frozen in March 2002. The consequence of this decision was that many features were postponed to the next release – Release 6. After freezing the content, the work continued and reached stability at the beginning of 2004. The Release 5 defines a finite architecture for SIP-based IP multimedia service machinery. It contains a functionality of logical elements, a description of how elements are connected, selected

protocols (see Chapter 2) and procedures (see Chapter 3). In addition, it is important to realize that optimization for the mobile communication environment has been also designed in the form of user authentication and authorization based on mobile identities (see Chapter 11), definite rules at the user network interface for compressing SIP messages (see Section 3.18) and security (see Section 3.21) and policy control mechanisms (see Section 3.10.3) that allow radio loss and recovery detection. Moreover, important aspects from the operator point of view are addressed while developing the architecture, such as the charging framework (see Section 3.11) and policy (see Section 3.10), and service control (see Section 3.13).

Release 6 IMS fixes the shortcomings in Release 5 IMS and also contains novel features. Release 6 was completed in September 2005. If Release 5 created the IMS machine we call Release 6 as the IMS application and interworking release. The Release 6 introduced standardized enhancements for services such as routing and signalling modifications e.g. Public Service Identity (see Section 3.5.5 and Section 12.11), sharing a single user identity between multiple devices (see Section 3.7). Improvements in routing capabilities smoothed the road to complete new standardized services such as presence (see Chapter 4), messaging (see Chapter 7), conferencing (see Chapter 8), PoC (see Chapter 6). In addition, IMS-CS voice interworking and WLAN access to IMS were completed. Moreover, improvements in security, policy and charging control and overall architecture were also completed.

#### *1.4.4 IMS Development in other Standardization Development Organizations*

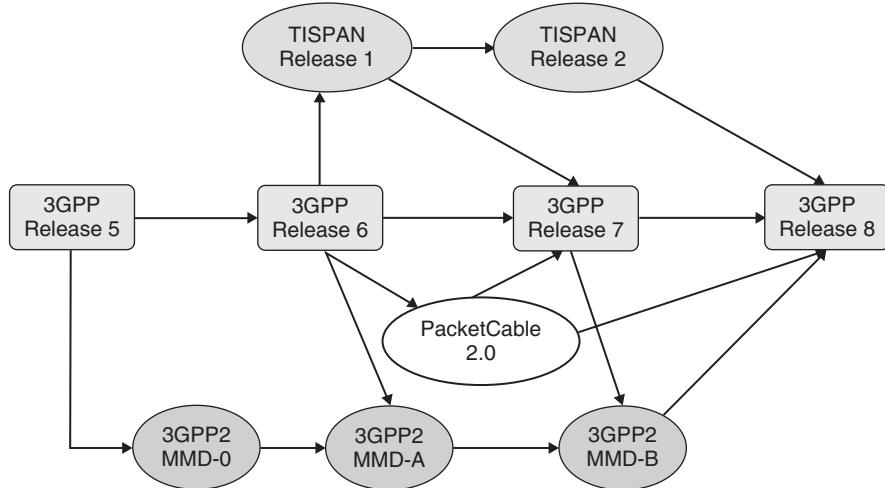
While 3GPP has finalized its Release 5 and Release 6 other standardization development organizations have done parallel developments to define their IMS variants. Most notable development organizations having own variants are ETSI Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN), Third Generation Partnership Project 2 (3GPP2) and Cablelabs.

TISPAN is the outcome of the merger of two ETSI bodies and it is building specifications to enable migration from fixed circuit switched networks to fixed packet-based networks with an architecture that can serve in both, also known as Next Generation Network (NGN). December 2005 TISPAN declared that they have completed the first NGN release (Release 1) that contained IP multimedia component. This IP multimedia component was based on 3GPP Release 6 and 7 IMS with TISPAN specific extensions and modifications. Since 2005 TISPAN has been working with its Release 2 which is expected to be completed during 2008.

3GPP2 created its IMS variant to support CDMA2000 access. This IMS variant is also known as Multimedia Domain (MMD). 3GPP2 has used 3GPP IMS specification as a starting point and at the time of writing 3GPP2 has three MMD releases: MMD-0, MMD-A, MMD-B.

Cablelabs is a development consortium that is defining specification for cable operators. An IMS like building block is present in their PacketCable 2.0 release. Again Cablelabs is using 3GPP IMS as a baseline.

Figure 1.5 depicts major development paths of IMS standards. This figure clearly reveals that there exists fragmentation in the IMS standardization arena. Luckily the industry has taken decisive steps towards harmonized IMS, the common IMS. Common IMS technology in all mobile and fixed ecosystems provides economies of scale to both the operators



**Figure 1.5** Road to standardized common IMS standards

and vendors in different ways. IMS vendors will be able to create the functionality once, and reuse it later. This means faster time to market, lower research and development cost due to eliminated replication effort. From an operator and service provider point of view it means that they will have a larger choice of vendors to select from and procurement cost of IMS products is lower. During 2007 3GPP and TISPAN made an agreement to stop IMS related development in TISPAN and focus all IMS development to 3GPP. Based on this agreement lot of functions and procedures developed in TISPAN Release 1 were included in 3GPP Release 7. TISPAN Release 2 is expected to be the last TISPAN release on IMS matters and functions and procedures are expected to be harmonized in 3GPP Release 8. Some Packetcable 'IMS' features were already included in 3GPP Release 7 and additional features and procedures originating from cable operators will be addressed in 3GPP Release 8. Harmonization of 3GPP2 MMD and 3GPP IMS is starting in Release 8. Due to late start full harmonization will probably happen in future 3GPP IMS releases.

#### 1.4.5 3GPP Release 7 and common IMS

3GPP Release 7 functional content was frozen in March 2007. It introduces two more access technologies (Data Over Cable Service Interface Specification (DOCSIS)<sup>1</sup> and xDSL<sup>2</sup>) and features and procedures originating from those and other general improvements. This can be considered as a step towards the ultimate goal of single common IMS. Major new features in Release 7 are: IMS multimedia telephony including supplementary services (see Chapter 9 and Chapter 12), SMS over any IP access (see Chapter 7, Section 7.4), Voice Call Continuity (see Section 3.20 and Chapter 13), local numbering (see Section 3.16), Combining CS calls and IMS sessions (see Section 3.19), Transit IMS (see Section 3.15), Interconnection Border Control Function (IBCF) (see Section 2.2.6.2),

<sup>1</sup> Access technology of Cablelabs.

<sup>2</sup> Access technology of TISPAN.

Globally Routable User Agent's URI (see Section 3.5.6 and Section 12.10), IMS emergency sessions (see Section 3.17), Identification of Communication Services in IMS (see Section 12.3.9) and new authentication model for fixed access (see Section 3.21.2.3).

#### 1.4.6 *Insight to 3GPP Release 8*

Standardization work on Release 8 is ongoing at the time of writing and work is expected to be completed by the end of 2008. This release will introduce a number of novel IMS features such as IMS centralized services which enables the use of IMS service machinery even though devices are using CS connection (GSM/3G CS radio) towards the network; multimedia session continuity which would improve the voice call continuity feature to enable continuity of multimedia media streams when IP access is changed; corporate access to IMS, a feature that enables integration of IP-PBX to the IMS network; service level interworking for messaging and number portability.

### 1.5 Why a SIP Solution Based on 3GPP Standards?

IETF is the protocol factory for Internet world and it is doing great work in this space but it does not define the ways that they are used, especially in the mobile domain. 3GPP is the body that took Session Initiation Protocol (SIP) as the control protocol for multimedia communication and 3GPP has built a finite architecture for SIP-based IP multimedia service machinery (the IMS). It contains a functionality of logical elements, a description of how elements are connected, selected protocols and procedures. 3GPP standardized solutions are needed to provide: interoperability between terminals from different vendors, interoperability between network elements from different vendors, interoperability across operator boundaries. The following advantages of 3GPP IMS against a pure IETF SIP service model can be listed:

- optimization for wireless usage:
  - SIP compression (see Section 3.18);
  - implicit registration (see Section 3.3);
  - network initiated re-authentication (see Section 11.14.2);
  - network initiated deregistration (see Section 11.15.3);
- authentication:
  - GPRS-IMS-Bundled Authentication (see Section 11.16);
  - NASS-IMS-Bundled Authentication (see Section 3.21.2.3);
  - ISIM/USIM authentication (see Section 11.6);
- policy control (see Section 3.10):
  - policy control and policy enforcement functions;
  - Rx and Gx reference points;
  - quality of Service (QoS);
- charging (see Section 3.11):
  - charging correlation (online and offline charging);
  - charging entity information;

- services and application server interfaces:
  - ISC interface (see Section 2.3.3);
  - Initial Filter Criteria (see Section 3.12.4);
- access network information available in IMS (see Section 11.11.1);
- mobility and roaming models defined (see Section 2.1.7);
- visited network identification (see Section 11.11.2);
- regulator requirements specified:
  - emergency call (incl. location information) (see Section 3.17);
  - legal interception;
  - number portability.

# 2

## IP Multimedia Subsystem Architecture

This chapter introduces the reader to the Internet Protocol (IP) Multimedia Subsystem (IMS). Section 2.1 explains basic architectural concepts: for instance, we explain why bearers are separated and why the home control model was selected. Section 2.2 gives a wide overview of IMS architecture, including an introduction to different network entities and main functionalities. Section 2.3 goes deeper and shows how the entities are connected and what protocols are used between them; it also describes their relationships to other domains: IP networks, Circuit Switched Core Network (CS CN) and IP Connectivity Access Networks (IP-CAN).

### 2.1 Architectural Requirements

There is a set of basic requirements which guides the way in which the IMS architecture has been created and how it should evolve in the future. This section covers the most significant requirements. Third Generation Partnership Project (3GPP) IMS requirements are documented in [3GPP TS 22.228].

#### 2.1.1 *IP Multimedia Sessions*

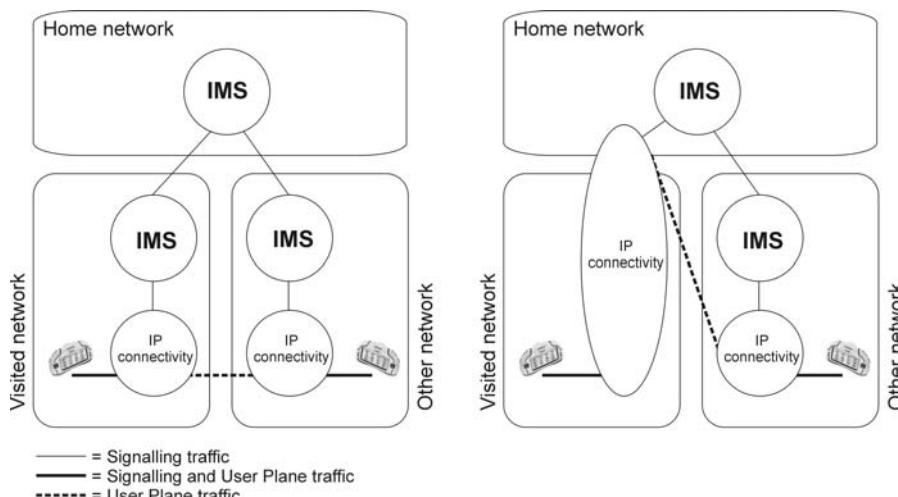
Existing communication networks are able to offer voice, video and messaging type of services using circuit-switched bearers. Naturally, end users' service offerings should not decline when users move to the packet-switched domain and start using the IMS. The IMS will take communication to the next level by offering enriched communication means see e.g. Chapter 9 and Chapter 12. IMS users are able to mix and match a variety of IP-based services in any way they choose during a single communication session. Users can integrate voice, video and text, content sharing and presence as part of their communication and can add or drop services as and when they choose. For example, two people can start a session as a voice session and later on add a game or video component to the same session.

### 2.1.2 IP Connectivity

As the name IP Multimedia Subsystem implies, a fundamental requirement is that a device has to have IP connectivity to access it. Peer-to-peer applications require end-to-end reachability and this connectivity is most easily attained with IP version 6 (IPv6) because IPv6 does not have address shortage. Therefore, 3GPP has arranged matters so that the IMS exclusively supports IPv6 [3GPP TS 23.221]. However, early IMS implementations and deployments may use IP version 4 (IPv4). 3GPP has created recommendations about how IP version interworking is handled in the IMS [3GPP TR 23.981]. This is further described in Section 3.17.

IP connectivity can be obtained either from the home network or the visited network. The furthest left part of Figure 2.1 presents an option in which User Equipment (UE) has obtained an IP address from a visited network. In the UMTS network, this means that the Radio Access Network (RAN), Serving GPRS Support Node (SGSN) and Gateway GPRS Support Node (GGSN) are located in the visited network when a user is roaming in the visited network. The furthest right part of Figure 2.1 presents an option in which a UE has obtained an IP address from the home network. In the UMTS network this means that the RAN and SGSN are located in the visited network when a user is roaming in the visited network. Obviously, when a user is located in the home network all necessary elements are in the home network and IP connectivity is obtained in that network.

It is important to note that a user can roam and obtain IP connectivity from the home network as shown in the figure. This would allow users to use new, fancy IMS services even when they are roaming in an area that does not have an IMS network but provides IP connectivity. In theory, it is possible to deploy an IMS network in a single area/country and use, say, General Packet Radio Service (GPRS) roaming to connect customers to the home network. In practice, this would not happen because routing efficiency would not be high enough. Consider routing Real-time Transport Protocol (RTP) voice packets from the USA to Europe and then back to the USA. However, this deployment model



**Figure 2.1** IMS connectivity options when a user is roaming

is important when operators are ramping up IMS networks or, in an initial phase, when they are offering non or near-real time multimedia services.

### 2.1.3 Ensuring Quality of Service for IP Multimedia Services

On the public Internet, delays tend to be high and variable, packets arrive out of order and some packets are lost or discarded. This will no longer be the case with the IMS. The underlying access and transport networks together with the IMS to provide end-to-end Quality of Service (QoS). Via the IMS, the UE negotiates its capabilities and expresses its QoS requirements during a Session Initiation Protocol (SIP) session setup or session modification procedure. The UE is able to negotiate such parameters as:

- media type, direction of traffic;
- media type bit rate, packet size, packet transport frequency;
- usage of RTP payload for media types;
- bandwidth adaptation.

After negotiating the parameters at the application level, UEs reserve suitable resources from the access network if not already available (typical case in mobile access). When end-to-end QoS is created, the UEs encode and packetize individual media types with an appropriate protocol (e.g., RTP) and send these media packets to the access and transport network by using a transport layer protocol (e.g., TCP or UDP) over IP. It is assumed that operators negotiate service-level agreements for guaranteeing the required QoS in the interconnection backbone. In the case of UMTS, operators could utilize the GPRS Roaming Exchange backbone.

### 2.1.4 IP Policy Control for Ensuring Correct Usage of Media Resources

IP policy control means the capability to authorize and control the usage of bearer traffic intended for IMS media, based on the signalling parameters at the IMS session. This requires interaction between the IP connectivity access network and the IMS. The means of setting up interaction can be divided into three different categories [3GPP TS 22.228, 23.207, 23.228]:

- The policy control element is able to verify that values negotiated in SIP signalling are used when activating bearers for media traffic. This allows an operator to verify that its bearer resources are not misused (e.g., the source and destination IP address and bandwidth in the bearer level are exactly the same as used in SIP session establishment).
- The policy control element is able to enforce when media traffic between the end points of a SIP session start or stop. This makes it possible to prevent the use of the bearer until session establishment is completed and allows traffic to start/stop in synchronization with the start/stop of charging for a session in IMS.
- The policy control element is able to receive notifications when the IP connectivity access network service has either modified, suspended or released the bearer(s) of a user associated with a session. This allows IMS to release an ongoing session because, for instance, the user is no longer in the coverage area.

This is described further in Section 3.10.

### 2.1.5 Secure Communication

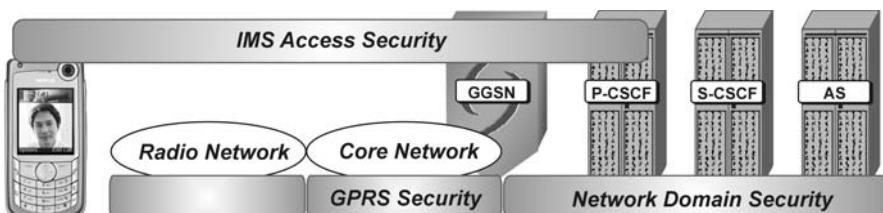
Security is a fundamental requirement in every telecommunication system and the IMS is not an exception. The IMS has its own authentication and authorization mechanisms between the UE and the IMS network in addition to access network procedures (e.g., GPRS network). Moreover, the integrity and optional confidentiality of the SIP messages is provided between the UE and the IMS network and between IMS network entities regardless of the underlaying core network (e.g., RAN and GPRS). Therefore, the IMS provides at least a similar level of security as the corresponding GPRS and circuit-switched networks: for example, the IMS ensures that users are authenticated before they can start using services, and users are able to request privacy when engaged in a session. Section 3.21 will discuss security features in more detail. An overview of applied security solutions is depicted in Figure 2.2.

### 2.1.6 Charging Arrangements

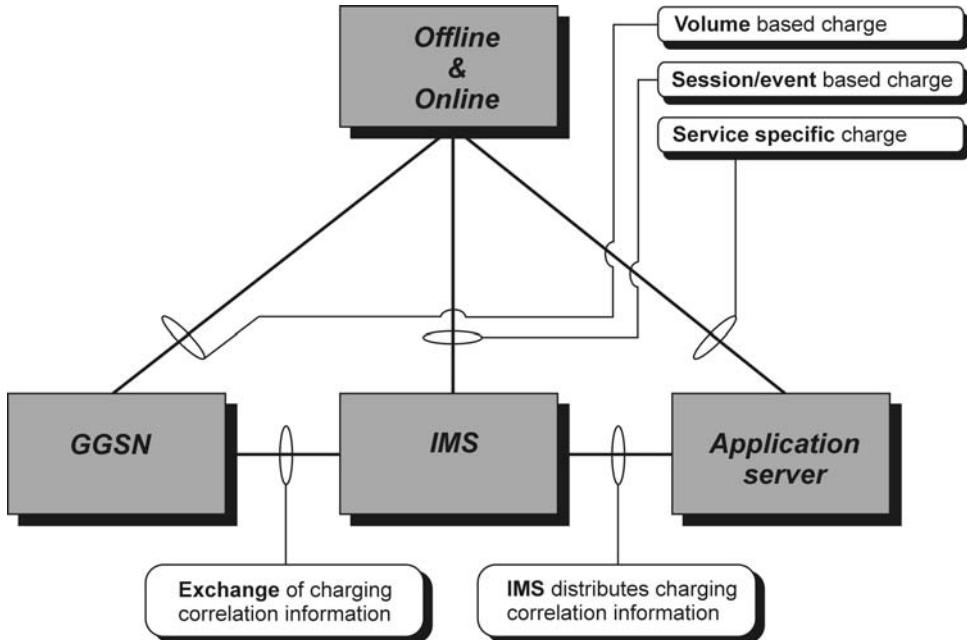
From an operator or service provider perspective the ability to charge users is a must in any network. The IMS architecture allows different charging models to be used. This includes, say, the capability to charge just the calling party or to charge both the calling party and the called party based on used resources in the transport level. In the latter case the calling party could be charged entirely on an IMS-level session: that is, it is possible to use different charging schemes at the transport and IMS level. However, an operator might be interested in correlating charging information generated at transport and IMS (service and content) charging levels. This capability is provided if an operator utilizes a policy control reference point. The charging correlation mechanism is further described in Section 3.11.7 and policy control is explained in Section 3.10.

As IMS sessions may include multiple media components (e.g., audio and video), it is required that the IMS provides a means for charging per media component. This would allow charging the called party if they add a new media component in a session. It is also required that different IMS networks are able to exchange information on the charging to be applied to a current session [3GPP TS 22.101, TR 23.815].

The IMS architecture supports both online and offline charging capabilities. Online charging is a charging process in which the charging information can affect in real time the service rendered and, therefore, directly interacts with session/service control. In practice, an operator could check the user's account before allowing the user to engage a session and to stop a session when all credits are consumed. Prepaid services are applications that need online charging capabilities. Offline charging is a charging process in which



**Figure 2.2** Overview of IMS security



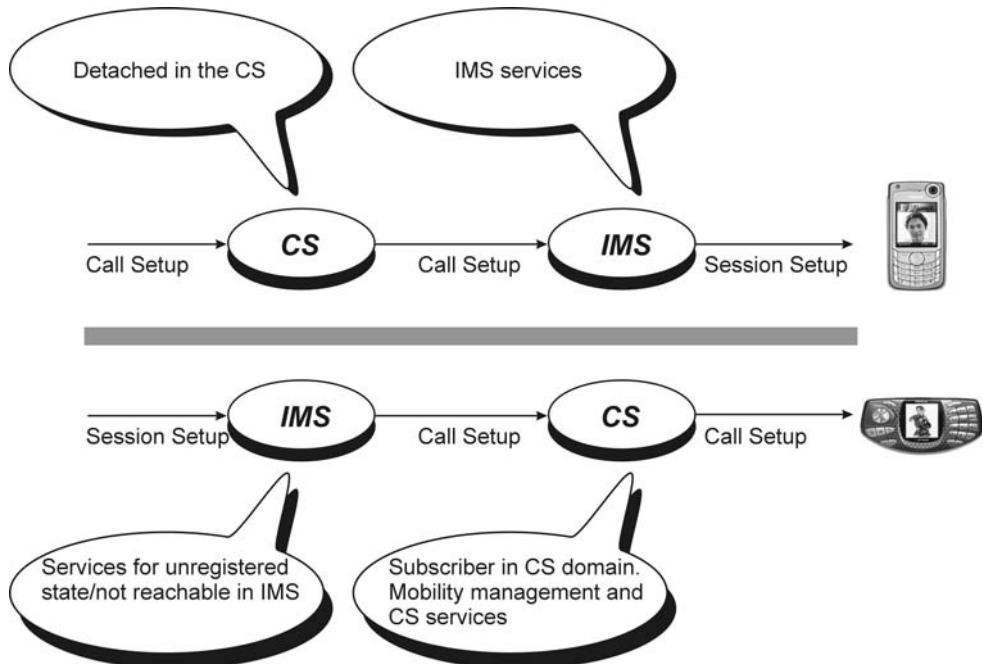
**Figure 2.3** IMS charging overview

the charging information does not affect in real time the service rendered. This is the traditional model in which the charging information is collected over a particular period and, at the end of the period, the operator posts a bill to the customer.

Figure 2.3 shows a simplified view of general charging arrangements in an IMS environment. The key observation is: the IMS adds the possibility to charge for user IP traffic in a more granular manner than before.

### 2.1.7 Support of Roaming

From a user point of view it is important to get access to their services regardless of their geographical location. The roaming feature makes it possible to use services even though the user is not geographically located in the service area of the home network. Section 2.1.2 has already described two instances of roaming: namely, GPRS roaming and IMS roaming. In addition to these two there exists an IMS circuit switched roaming case. GPRS roaming means the capability to access the IMS when the visited network provides the RAN and SGSN and the home network provides the GGSN and IMS. The IMS roaming model refers to a network configuration in which the visited network provides IP connectivity (e.g., RAN, SGSN, GGSN) and the IMS entry point (i.e., P-CSCF) and the home network provides the remaining IMS functionalities. The main benefit of this roaming model compared with the GPRS roaming model is optimum usage of user-plane resources. Roaming between the IMS and the CS CN domain refers to inter-domain roaming between IMS and CS. When a user is not registered or reachable in one domain a session can be routed to the other domain. It is important to note that both the CS CN



**Figure 2.4** IMS/CS roaming alternatives

domain and the IMS domain have their own services and cannot be used from another domain. Some services are similar and available in both domains (e.g., Voice over IP in IMS and speech telephony in CS CN). Figure 2.4 shows different IMS/CS roaming cases.

### 2.1.8 Interworking with Other Networks

It is evident that the IMS is not deployed over the world at the same time. Moreover, people may not be able to switch terminals or subscriptions very rapidly. This will raise the issue of being able to reach people regardless of what kind of terminals they have or where they live. To be a new, successful communication network technology and architecture the IMS has to be able to connect to as many users as possible. Therefore, the IMS supports communication with PSTN, ISDN, mobile and Internet users. Additionally, it will be possible to support sessions with Internet applications that have been developed outside the 3GPP community [3GPP TS 22.228]. Voice and video IMS-CS interworking is further described in Section 3.14. Messaging interworking is covered in Section 7.4.

### 2.1.9 Service Control Model

In 2G mobile networks the visited service control is in use. This means that, when a user is roaming, an entity in the visited network provides services and controls the traffic for the user. This entity in the second generation (2G) is called a visited mobile service switching centre. In the early days of Release 5 both visited and home service control models were supported. Supporting two models would have required that every problem

have more than one solution; moreover, it would reduce the number of optimal architecture solutions, as simple solutions may not fit both models. Supporting both models would have meant additional extensions for Internet Engineering Task Force (IETF) protocols and increased the work involved in registration and session flows. The visited service control was dropped because it was a complex solution and did not provide any noticeable added value compared with the home service control. On the contrary, the visited service control imposes some limitations. It requires a multiple relationship and roaming models between operators. Service development is slower as both the visited and home network would need to support similar services, otherwise roaming users would experience service degradations. In addition, the number of interoperator reference points increase, which requires complicated solutions (e.g., in terms of security and charging). Therefore, home service control was selected; this means that the entity that has access to the subscriber database and interacts directly with service platforms is always located at the user's home network.

#### 2.1.10 Layered Design and Access Independence

3GPP has decided to use a layered approach to architectural design. This means that transport and bearer services are separated from the IMS signalling network and session management services. Further services are run on top of the IMS signalling network. Figure 2.5 shows the design.

In some cases it may be impossible to distinguish between functionality at the upper and lower layers. The layered approach aims at a minimum dependence between layers. A benefit is that it facilitates the addition of new access networks to the system later on. The IMS was originally designed to be access-independent so that IMS services can be provided over any IP connectivity network. Unfortunately, Release 5 IMS specifications contain some GPRS-specific features. In Release 6 onwards access-specific issues are

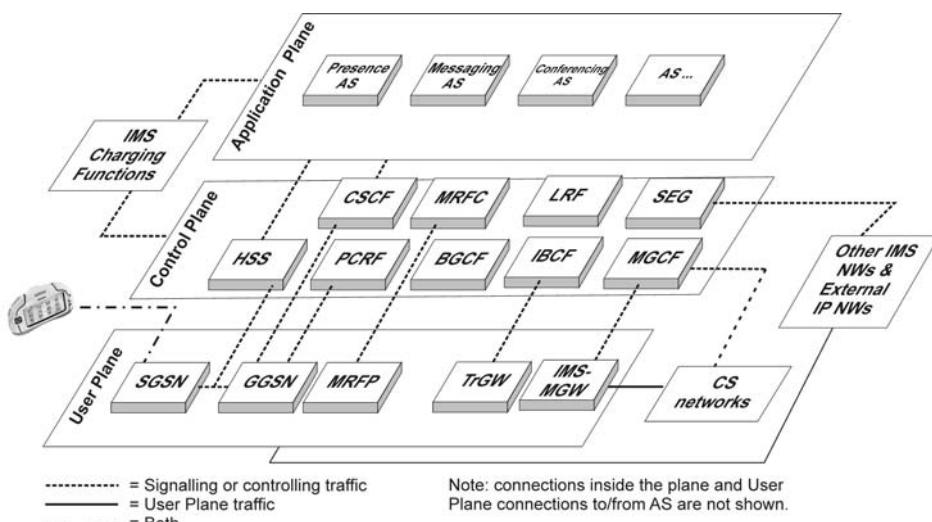
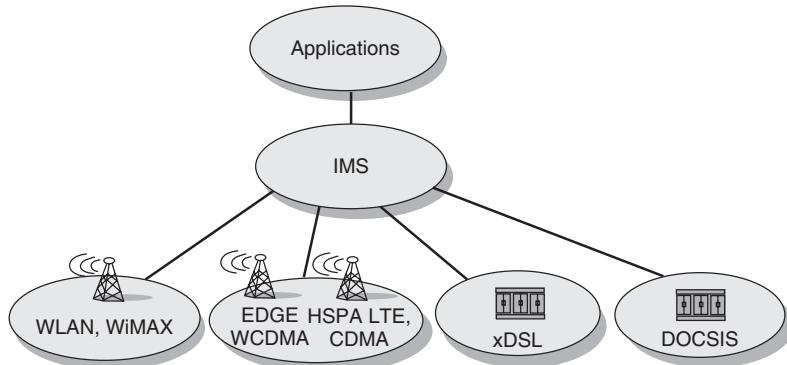


Figure 2.5 IMS and layered architecture



**Figure 2.6** Access independence

separated from the core IMS description and the IMS architecture returned to its born state (i.e., access independent). A number of different accesses have been added since then. Wireless Local Area Network (WLAN) access to the IMS was added in 3GPP Release 6, fixed broadband access was added in Release 7 and additional access like DOCSIS®, WiMAX™ and cdma2000® are being harmonized in Release 8 and post Release 8 as shown in Figure 2.6.

The layered approach increases the importance of the application layer as services are designed to work independently of the access network and the IMS is equipped to bridge the gap between them. Whether the subscriber is using a mobile phone or a PC client to communicate, the same presence and group list functions in IMS will be used. Different services have different requirements. These include:

- bandwidth;
- latency;
- processing power in the device.

This means that in order for different services to be executed properly, the network has to be equipped with access-aware control and service logic for multimedia services. Multi-access functionality is built into the IMS architecture, which offers a way for fixed and mobile operators to deliver a fixed to mobile convergence solution. This will enable service providers to use the characteristics and capabilities of the currently selected device and its network access method and adapt to it dynamically.

## 2.2 Description of IMS-related Entities and Functionalities

This section discusses IMS entities and key functionalities. These entities can be roughly classified into six main categories:

- session management and routing family (CSCFs);
- databases (HSS, SLF);
- services (application server, MRFC, MRFP);

- interworking functions (BGCF, MGCF, IMS-MGW, SGW);
- support functions (PCRF, SEG, IBCF, TrGW, LRF);
- charging.

It is important to understand that IMS specifications are set up so that the internal functionality of the network entities is not specified in detail. Instead, specifications describe reference points between entities and functionalities supported at the reference points. For instance, how does CSCF obtain user data from databases? Different reference points will be described in Section 2.3. Additionally, GPRS functions are described at the end of this section.

### 2.2.1 *Call Session Control Functions (CSCF)*

There are four different kinds of Call Session Control Functions (CSCF): Proxy-CSCF (P-CSCF), Serving-CSCF (S-CSCF), Interrogating-CSCF (I-CSCF) and Emergency-CSCF (E-CSCF). Each CSCF has its own special tasks and these tasks are described in the following subsections. Common to P-CSCF, S-CSCF and I-CSCF is that they all play a role during registration and session establishment and form the SIP routing machinery. Moreover, all functions are able to send charging data to an offline charging function. There are some common functions that P-CSCF and S-CSCF are able to perform. Both entities are able to release sessions on behalf of the user (e.g., when S-CSCF detects a hanging session or P-CSCF receives a notification that a media bearer is lost) and are able to check that the content of the SIP request or response conforms operator's policy and user's subscription (e.g. content of the Session Description Protocol (SDP) payload contains media types or codecs, which are allowed for a user).

#### 2.2.1.1 **Proxy Call Session Control Function (P-CSCF)**

Proxy Call Session Control Function (P-CSCF) is the first contact point for users within the IMS. It means that all SIP signalling traffic from the UE will be sent to the P-CSCF. Similarly, all terminating SIP signalling from the network is sent from the P-CSCF to the UE. There are four unique tasks assigned for the P-CSCF: SIP compression, IPSec security association, interaction with Policy and Charging Rules Function (PCRF) and emergency session detection.

As the SIP protocol is a text-based signalling protocol, it contains a large number of headers and header parameters, including extensions and security-related information which means that their message sizes are larger than with binary-encoded protocols. For speeding up the session establishment 3GPP has mandated the support of SIP compression between the UE and P-CSCF. The P-CSCF needs to compress messages if the UE has indicated that it wants to receive signalling messages compressed. SIP compression is described in Sections 3.18 and 11.10 and 12.4.

P-CSCF is responsible for maintaining Security Associations (SAs) and applying integrity and confidential protection for SIP signalling. This is achieved during SIP registration as the UE and P-CSCF negotiate IPSec SAs. After the initial registration the P-CSCF is able to apply integrity and confidential protection of SIP signalling. See Sections 11.6 and 11.7 and 11.8 for more detailed descriptions.

The P-CSCF is tasked to relay session and media-related information to the PCRF when an operator wants to apply policy and charging control. Based on the received information the PCRF is able to derive authorized IP QoS information and charging rules that will be passed to the access gateway (e.g. GGSN). This concept is covered in Section 3.10. Moreover, via the PCRF and P-CSCF the IMS is able to deliver IMS charging correlation information to the access network and, similarly, via the PCRF and P-CSCF the IMS is able to receive access charging correlation information from the access network. This makes it possible to merge charging data records coming from the IMS and access networks in the billing system. How this is done is shown in Section 3.11.7.

P-CSCF plays an important role in IMS emergency session handling as the P-CSCF is tasked to detect emergency requests in all possible cases. P-CSCF is expected to reject emergency attempts based on operator policy (e.g. user is attempting to make emergency call via home P-CSCF when roaming) or based on network capability (P-CSCF or the rest of the IMS core is pre-Release 7 which do not support IMS functionality).

### **2.2.1.2 Interrogating Call Session Control Function (I-CSCF)**

Interrogating Call Session Control Function (I-CSCF) is a contact point within an operator's network for all connections destined to a subscriber of that network operator. There are three unique tasks assigned for the I-CSCF:

- Obtaining the name of the next hop (either S-CSCF or application server) from the Home Subscriber Server (HSS).
- Assigning an S-CSCF based on received capabilities from the HSS. The assignment of the S-CSCF will take place when a user is registering with the network or a user receives a SIP request while they are unregistered from the network but has services related to an unregistered state (e.g., voice mail). This procedure is described in more detail in Section 3.9.
- Routing incoming requests further to an assigned S-CSCF or the application server (in the case of public service identity see Section 12.11).

### **2.2.1.3 Serving Call Session Control Function (S-CSCF)**

Serving Call Session Control Function (S-CSCF) is the focal point of the IMS as it is responsible for handling registration processes, making routing decisions and maintaining session states and storing the service profile(s). When a user sends a registration request it will be routed to the S-CSCF, which downloads authentication data from the HSS. Based on the authentication data it generates a challenge to the UE. After receiving the response and verifying it the S-CSCF accepts the registration and starts supervising the registration status. After this procedure the user is able to initiate and receive IMS services. Moreover, the S-CSCF downloads a service profile from the HSS as part of the registration process and delivers user (e.g. information about implicitly registered identities see Section 3.3) and device specific information to the registered UE see Section 3.5.6).

A service profile is a collection of user-specific information that is permanently stored in the HSS. The S-CSCF downloads the service profile associated with a particular public user identity (e.g., joe.doe@ims.example.com) when this particular public user identity

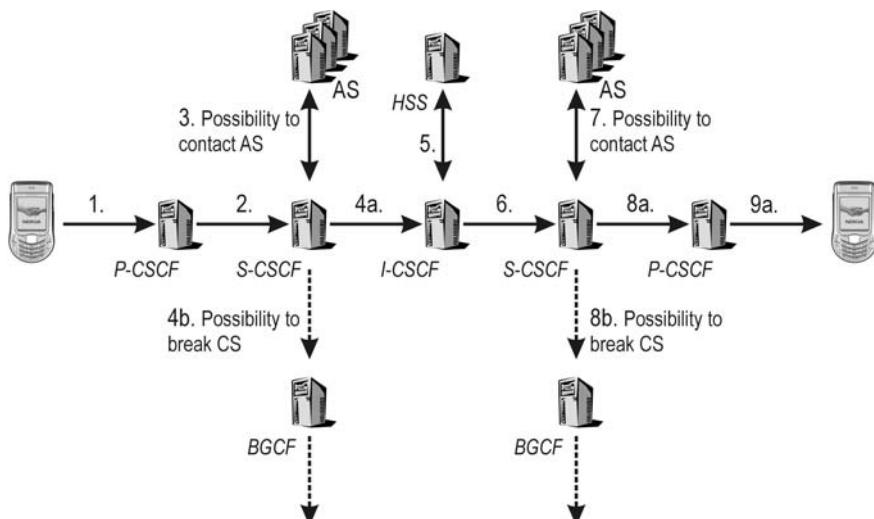
is registered in the IMS. The S-CSCF uses information included in the service profile to decide when and, in particular, which application server(s) is contacted when a user sends a SIP request or receives a request from somebody. Moreover, the service profile may contain further instructions about what kind of media policy the S-CSCF needs to apply – for example, it may indicate that a user is only allowed to use audio and application media components but not video media components.

The S-CSCF is responsible for key routing decisions as it receives all UE-originated and UE-terminated sessions and transactions. When the S-CSCF receives a UE-originating request via the P-CSCF it needs to decide if application servers are contacted prior to sending the request further on. After possible application server(s) interaction the S-CSCF either continues a session in IMS or breaks to other domains (CS or another IP network). When the UE uses a Mobile Station ISDN (MSISDN) number to address a called party then the S-CSCF converts the MSISDN number (i.e., a tel URL) to SIP Universal Resource Identifier (URI) format prior to sending the request further, as the IMS does not route requests based on MSISDN numbers. Similarly, the S-CSCF receives all requests which will be terminated at the UE. Although, the S-CSCF knows the IP address of the UE from the registration it routes all requests via the P-CSCF, as the P-CSCF takes care of SIP compression and security functions. Prior to sending a request to the P-CSCF, the S-CSCF may route the request to an application server(s), for instance, checking possible redirection instructions. Figure 2.7 illustrates the S-CSCF's role in routing decisions.

In addition, the S-CSCF is able to send accounting-related information to the Online Charging System for online charging purposes (i.e., supporting pre-paid subscribers).

### 2.2.2 Emergency Call Session Control Function (E-CSCF)

E-CSCF is a dedicated functionality to handle IMS emergency requests such as sessions towards police, fire brigade and ambulance. The main task of E-CSCF is to select an



**Figure 2.7** S-CSCF routing and basic IMS session setup

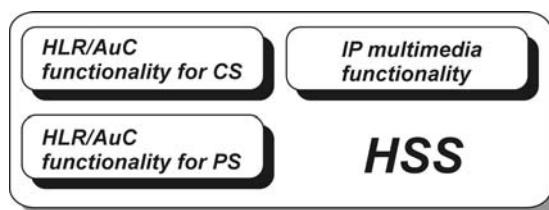
emergency centre also known as a Public Safety Answering Point where an emergency request should be delivered. Typically a selection criterion is a calling user's location and possible type of emergency (e.g. police, coast guard). Once the appropriate emergency centre is selected the E-CSCF routes the request to the emergency centre. IMS emergency sessions are described in more detail in Section 3.17.

### 2.2.3 Databases

There are two main databases in the IMS architecture: Home Subscriber Server (HSS) and Subscription Locator Function (SLF).

The HSS is the main data storage for all subscriber and service-related data of the IMS. The main data stored in the HSS include user identities, registration information, access parameters and service-triggering information [3GPP TS 23.002]. User identities consist of two types: private and public user identities (see Sections 3.5.2 and 3.5.1). The private user identity is a user identity that is assigned by the home network operator and is used for such purposes as registration and authorization, while the public user identity is the identity that other users can use for requesting communication with the end-user. IMS access parameters are used to set up sessions and include parameters like user authentication, roaming authorization and allocated S-CSCF names. Service-triggering information enables SIP service execution. The HSS also provides user-specific requirements for S-CSCF capabilities. This information is used by the I-CSCF to select the most suitable S-CSCF for a user (see Section 3.9). In addition to functions related to IMS functionality, the HSS contains the subset of Home Location Register and Authentication Center (HLR/AUC) functionality required by the Packet-Switched (PS) domain and the Circuit-Switched (CS) domain. The structure of the HSS is shown in Figure 2.8. Communication between different HSS functions is not standardized.

HLR functionality is required to provide support to PS domain entities, such as SGSN and GGSN. This enables subscriber access to PS domain services. In similar fashion the HLR provides support for CS domain entities, like MSC/MSC servers. This enables subscriber access to CS domain services and supports roaming to Global System for Mobile Communications (GSM)/UMTS CS domain networks. The AUC stores a secret key for each mobile subscriber, which is used to generate dynamic security data for each mobile subscriber. Data are used for mutual authentication of the International Mobile Subscriber Identity (IMSI) and the network. Security data are also used to provide integrity protection and ciphering of the communication over the radio path between the UE and the network. There may be more than one HSS in a home network, depending on the



**Figure 2.8** Structure of HSS

number of mobile subscribers, the capacity of the equipment and the organization of the network. There are multiple reference points between the HSS and other network entities.

The SLF is used as a resolution mechanism that enables the I-CSCF, the S-CSCF and the AS to find the address of the HSS that holds the subscriber data for a given user identity when multiple and separately addressable HSSs have been deployed by the network operator (see Figure 2.12).

#### 2.2.4 Service Functions

Three functions in this book are categorized as IMS service-related functions – namely, Multimedia Resource Function Controller (MRFC), Multimedia Resource Function Processor (MRFP) and Application Server (AS).

Keeping in mind the layered design, ASs are not pure IMS entities; rather, they are functions on top of IMS. However, ASs are described here as part of IMS functions because ASs are entities that provide value-added multimedia services in the IMS, such as presence and Push to Talk Over Cellular. An AS resides in the user's home network or in a third-party location. The third party here means a network or a standalone AS. The main functions of the AS are:

- The possibility to process and impact an incoming SIP session received from the IMS.
- The capability to originate SIP requests.
- The capability to send accounting information to the charging functions.

The services offered are not limited purely to SIP-based services since an operator is able to offer access to services based on the Customized Applications for Mobile Network Enhanced Logic (CAMEL) Service Environment (CSE) and the Open Service Architecture (OSA) for its IMS subscribers [3GPP TS 23.228]. Therefore, AS is the term used generically to capture the behaviour of the SIP AS, OSA Service Capability Server (SCS) and CAMEL IP Multimedia Service Switching Function (IM-SSF).

Using the OSA an operator may utilize such service capability features as call control, user interaction, user status, data session control, terminal capabilities, account management, charging and policy management for developing services [3GPP TS 29.198]. An additional benefit of the OSA framework is that it can be used as a standardized mechanism for providing third-party ASs in a secure manner to the IMS, as the OSA itself contains initial access, authentication, authorization, registration and discovery features (the S-CSCF does not provide authentication and security functionality for secure direct third-party access to the IMS). As the support of OSA services is down to operator choice, it is not architecturally sound to support OSA protocols and features in multiple entities. Therefore, OSA SCS is used to terminate SIP signalling from the S-CSCF. The OSA SCS uses an OSA Application Program Interface (API) to communicate with an actual OSA application server.

The IM-SSF was introduced in the IMS architecture to support legacy services that are developed in the CSE. It hosts CAMEL network features (trigger detection points, CAMEL Service Switching Finite State Machine, etc.) and interworks with the CAMEL Application Part (CAP) interface.

A SIP AS is a SIP-based server that hosts a wide range of value-added multimedia services. A SIP AS could be used to provide presence, messaging, Push to talk Over

Cellular and conferencing services. The different functions of SIP servers are described in more detail in Sections 3.13.4, as part of service provisioning.

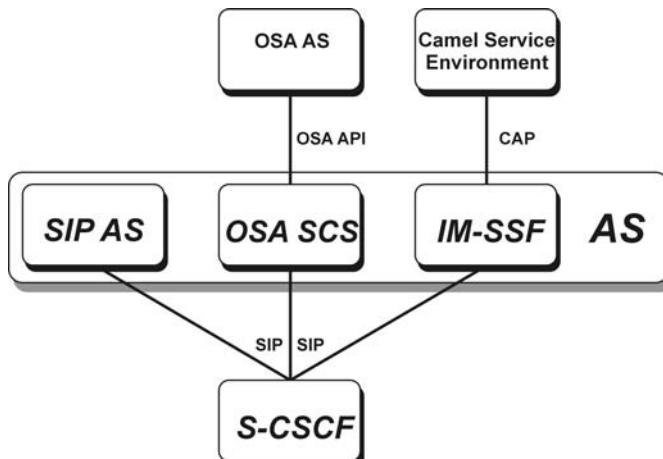
Figure 2.9 shows how different functions are connected. From the perspective of the S-CSCF SIP AS, the OSA service capability server and the IM-SSF exhibit the same reference point behaviour.

An AS may be dedicated to a single service and a user may have more than one service, therefore there may be one or more AS per subscriber. Additionally, there may be one or more AS involved in a single session. For example, an operator could have one AS to control terminating traffic to a user based on user preferences (e.g., redirecting all incoming multimedia sessions to an answer machine between 5 pm and 7 am) and another AS to adapt the content of instant messages according to the capabilities of the UE (screen size, number of colours, etc.).

MRFC and MRFP together provide mechanisms for bearer-related services such as conferencing, announcements to a user or bearer transcoding in the IMS architecture. The MRFC is tasked to handle SIP communication to and from the S-CSCF and to control the MRFP. The MRFP in turn provides user-plane resources that are requested and instructed by the MRFC. The MRFP performs the following functions:

- Mixing of incoming media streams (e.g., for multiple parties).
- Media stream source (for multimedia announcements).
- Media stream processing (e.g., audio transcoding, media analysis) [3GPP TS23.228, TS 23.002].

Currently, the role of MRFC in the IMS architecture is minor, as in IMS conferencing work [3GPP TS 24.147] and in IMS Multimedia Telephony service [3GPP TS 24.173] the MRFC is co-located with an AS.



**Figure 2.9** Relationship between different application server types

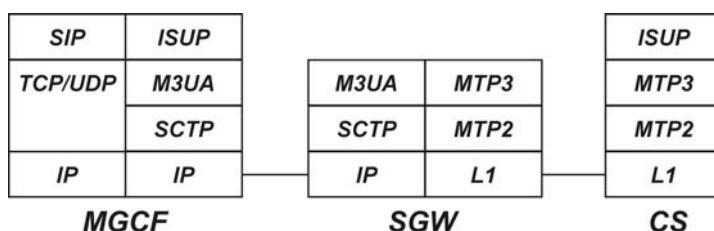
### 2.2.5 IMS-CS Interworking Functions

This section introduces four interworking functions, which are needed to enable voice and video interworking between IMS and the CS CN. Messaging interworking is covered in Section 7.4.

Section 2.2.1.3 explained that the S-CSCF decides when to break to the CS CN. For breaking out the S-CSCF sends a SIP session request to the Breakout Gateway Control Function (BGCF); it further chooses where a breakout to the CS domain occurs. The outcome of a selection process can be either a breakout in the same network in which the BGCF is located or another network. If the breakout happens in the same network, then the BGCF selects a Media Gateway Control Function (MGCF) to handle the session further. If the breakout takes place in another network, then the BGCF forwards the session to another BGCF in a selected network [3GPP TS 23.228]. The latter option allows routing of signalling and media over IP near to the called user.

When a SIP session request hits the MGCF it performs protocol conversion between SIP protocols and the ISDN User Part (ISUP), or the Bearer Independent Call Control (BICC) and sends a converted request via the Signalling Gateway (SGW) to the CS CN. The SGW performs signalling conversion (both ways) at the transport level between the IP-based transport of signalling (i.e., between Sigtran SCTP/IP and SS7 MTP) and the Signalling System No. 7 (SS7) based transport of signalling. The SGW does not interpret application layer (e.g., BICC, ISUP) messages, as is shown in Figure 2.10. The MGCF also controls the IMS Media Gateway (IMS-MGW). The IMS-MGW provides the user-plane link between CS CN networks and the IMS. It terminates the bearer channels from the CS network and media streams from the backbone network (e.g., RTP streams in an IP network or AAL2/ATM connections in an ATM backbone), executes the conversion between these terminations and performs transcoding and signal processing for the user plane when needed. In addition, the IMS-MGW is able to provide tones and announcements to CS users.

Similarly, all incoming call control signalling from a CS user to an IMS user is destined to the MGCF that performs the necessary protocol conversion and sends a SIP session request to the I-CSCF for a session termination. At the same time, the MGCF interacts with the IMS-MGW and reserves necessary IMS-MGW resources at the user plane. The overall IMS–CS interworking concept is covered in Section 3.14 which includes two example figures.



**Figure 2.10** Signalling conversion in the SGW

### 2.2.6 Support Functions

Several functions in this book are categorized as support functions – namely, Policy and Charging Rules Function (PCRF), Interconnection Border Control Function (IBCF), Transition Gateway (TrGW), Security Gateway (SEG) and Location Retrieval Function (LRF).

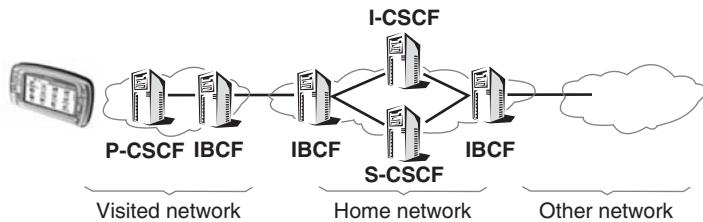
#### 2.2.6.1 Policy and Charging Rules Function

The PCRF is responsible for making policy and charging control decisions based on session and media-related information obtained from the P-CSCF. Session establishment in the IMS involves an end-to-end message exchange using SIP and SDP. During the message exchange UEs negotiate a set of media characteristics (e.g., common codec(s)). If an operator applies the policy and charging control, then the P-CSCF will forward the relevant SDP information to the PCRF together with an indication of the originator. The PCRF generates charging rules and authorizes the IP flows of the chosen media components by mapping from SDP parameters to authorized IP QoS parameters for transfer to the access network – e.g., GGSN in case of UMTS/GPRS access (GPRS access is assumed hereafter) – via the Gx reference point. On receiving the PDP context activation or modification, the GGSN asks for authorization information from the PCRF. Based on available information in the PCRF it makes an authorization decision which will be enforced in the GGSN. In addition to a bearer authorization decision the PCRF receives reports on transport plane events e.g. when the bearer is lost or when the bearer is released. Based on this information the PCRF is able to inform the P-CSCF about the occurred event. This allows the P-CSCF to effect charging, and it may even start releasing an IMS session on behalf of the user. Moreover, the PCRF can be used to exchange charging correlation identifiers which enables the operator to correlate charging detail records generated in the access network and IMS network. Policy control is described further in Section 3.10. Charging correlation and PCRF's role in charging is covered further in Section 3.11.

#### 2.2.6.2 Interconnection Border Control Function and Transition Gateway

An Interconnection Border Control Function (IBCF) provides specific functions in order to perform interconnection between two operator domains. It enables communication between IPv6 and IPv4 IMS applications, network topology hiding, controlling transport plane functions, screening of SIP signalling information, selecting the appropriate signalling interconnect and generation of charging data records. This entity was originally defined in ETSI TISPAN Release 1 architecture and it was included in 3GPP IMS in Release 7. Possible deployments for IBCF are shown in Figure 2.11.

Capability to translate between IPv4 and IPv6 addresses emerges when IMS communication takes place between operators that are supporting different IP address versions. IBCF is tasked to bridge these two domains by acting as an Application Level Gateway (ALG). ALG takes care of modifying SIP and SDP information in such a way that UEs using different (IPv6 and IPv4) IP version can communicate with each other. The ALG functionality inside the IBCF controls Transition Gateway (TrGW) which is responsible for providing IP version interworking in transport plane (i.e. modifying IP packets transporting actual IMS application media such as RTP). More information on IP version interworking can be found in Section 3.22.



**Figure 2.11** Possible deployments for Interconnection Border Control Function

Network topology hiding functionality could be used to hide the configuration, capacity and topology of the network from outside an operator's network.<sup>1</sup> If an operator wants to use hiding functionality then the operator must place an IBCF in the routing path when receiving requests or responses from other IMS networks. Similarly, the IBCF must be placed in the routing path when sending requests or responses to other IMS networks. The IBCF performs the encryption and decryption of all headers which reveal topology information about the operator's IMS network.

An operator may use IBCF to support its local policy. An operator may use IBCF as the entry/exit point for its network and IBCF can be used to screen signalling information (i.e. omit or modify some received SIP headers prior to forwarding SIP messages further to other networks).

Transition Gateway (TrGW) provides functions for network address/port translation and IPv4/IPv6 protocol translation. TrGW binds addresses in IPv6 network with addresses in IPv4 network and vice versa to provide transparent routing between the two IP domains without requiring any changes to UEs. This entity is located in transport plane and IP version interworking mechanisms are shown in Section 3.22.

### 2.2.6.3 Security Gateway

The Security Gateway (SEG) has the function of protecting control-plane traffic between security domains. The security domain refers to a network that is managed by a single administrative authority. Typically, this coincides with operator borders. The SEG is placed at the border of the security domain and it enforces the security policy of a security domain toward other SEGs in the destination security domain. In the IMS all traffic within the IMS is routed via SEGs, especially when the traffic is inter domain, meaning that it originates from a different security domain from the one where it is received. When protecting interdomain IMS traffic, both confidentiality as well as data integrity and authentication are mandated [3GPP TS 33.203]. The concept behind a security domain is described more thoroughly in Section 3.19.3.2.

### 2.2.6.4 Location Retrieval Function

The Location Retrieval Function (LRF) assists E-CSCF in handling IMS emergency sessions by delivering location information of the UE that has initiated an IMS emergency

<sup>1</sup> Prior 3GPP Release 7 function called Topology Hiding Inter-network Gateway (THIG) was responsible for this functionality. From Release 7 onwards IBCF supercedes THIG in IMS architecture.

session and/or address of Public Safety Answering Point (PSAP) where the session should be sent. To provide location information the LRF may contain location server or have interface towards external location server (e.g. gateway mobile location centre). To resolve appropriate PSAP it may contain Routing Determination Function (RDF) which is used to map the user's location to address of PSAP. The LRF may provide other emergency session parameters according to local regulations, for example, this information may include Emergency Service Query Key, Emergency Service Routing Number, Last Routing Option in North America, location number in EU, PSAP SIP URI or Tel URI. IMS emergency sessions are described in more detail in Section 3.17.

### 2.2.7 Charging Entities

Different charging entities and corresponding reference points will be described separately in Section 3.11.

### 2.2.8 GPRS Entities

#### 2.2.8.1 Serving GPRS Support Node (SGSN)

The Serving GPRS Support Node (SGSN) links the RAN to the packet core network. It is responsible for performing both control and traffic-handling functions for the PS domain. The control part contains two main functions: mobility management and session management. Mobility management deals with the location and state of the UE and authenticates both the subscriber and the UE. The control part of session management deals with connection admission control and any changes in the existing data connections. It also supervises 3G network services and resources. Traffic handling is the part of session management that is executed. The SGSN acts as a gateway for user data tunnelling: in other words, it relays user traffic between the UE and the GGSN. As a part of this function, the SGSN also ensures that connections receive the appropriate QoS. In addition, the SGSN generates charging information.

#### 2.2.8.2 Gateway GPRS Support Node (GGSN)

The Gateway GPRS Support Node (GGSN) provides interworking with external packet data networks. The prime function of the GGSN is to connect the UE to external data networks, where IP-based applications and services reside. The external data network could be the IMS or the Internet, for instance. In other words, the GGSN routes IP packets containing SIP signalling from the UE to the P-CSCF and vice versa. Additionally, the GGSN takes care of routing IMS media IP packets toward the destination network (e.g., to GGSN in the terminating network). The interworking service provided is realized as access points that relate to the different networks the subscriber wants to connect. In most cases the IMS has its own access point. When the UE activates a bearer (PDP context) toward an access point (IMS), the GGSN allocates a dynamic IP address to the UE. This allocated IP address is used in IMS registration and when the UE initiates a session as a contact address of the UE. Additionally, the GGSN polices and supervises the PDP context usage for IMS media traffic and generates charging information.

## 2.3 IMS Reference Points

This section explains how the previously described network entities are connected to each other and what protocol is used; moreover, the IMS architecture is depicted (Figure 2.12). You will also find an overview of SIP-based reference points (i.e., where SIP is used and the main procedures involved). However, you will realize that the level of description of SIP-based reference points is not so deep as with Diameter-based reference points. The reason for this division is that several chapters in this book are dedicated to SIP and SDP procedures where such descriptions are given in detail. Summary of reference points is given in Table 2.3.

For the sake of clarity, it is impossible to include everything in one figure; so, please note the following:

- Figure 2.12 does not show charging-related functions or reference points (see Section 3.11 for more details).
- The figure does not show different types of ASs (see Section 2.2.4 for more details).
- The figure does not show the user-plane connections between different IMS networks and the AS.
- The figure does not show the SEG at the Mm, Mk, Mw reference points.
- The figure does not show the Iq reference point.

### 2.3.1 Gm Reference Point

The Gm reference point connects the UE to the IMS. It is used to transport all SIP signalling messages between the UE and the IMS. The IMS counterpart is the P-CSCF.

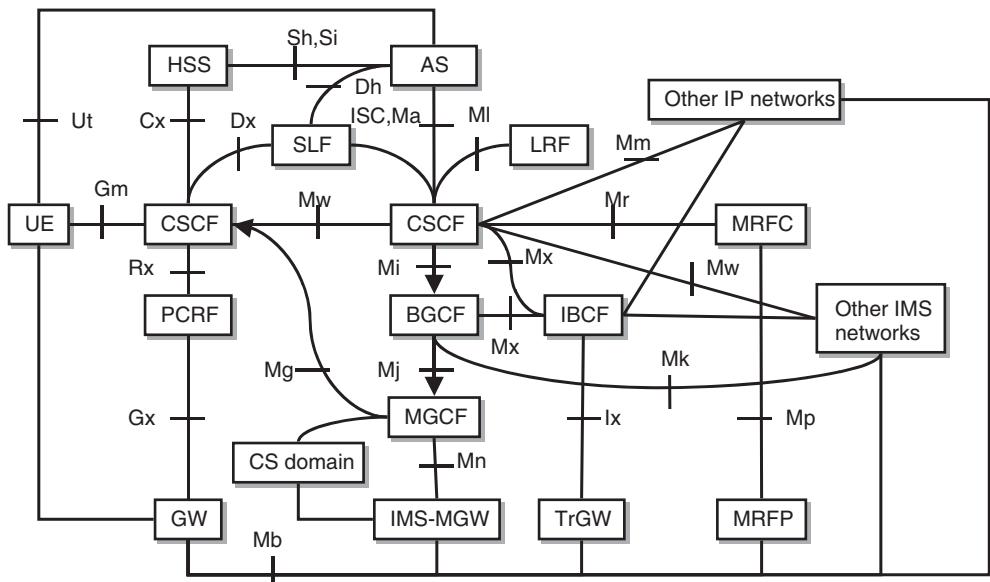


Figure 2.12 IMS architecture

Procedures in the Gm reference point can be divided into three main categories: registration, session control and transactions:

- In the registration procedure the UE uses the Gm reference point to send a registration request with an indication of supported security mechanisms to the P-CSCF. During the registration process the UE exchanges the necessary parameters for authenticating both itself and the network, gets implicit registered user identities, negotiates the necessary parameters for an SA with the P-CSCF and possibly starts SIP compression. In addition, the Gm reference point is used to inform the UE if network-initiated de-registration or network-initiated re-authentication occurs.
- Session control procedures contain mechanisms for both mobile-originated sessions and mobile-terminated sessions. In mobile-originated sessions the Gm reference point is used to forward requests from the UE to the P-CSCF. In mobile-terminated sessions the Gm reference point is used to forward requests from the P-CSCF to the UE.
- Transaction procedures are used to send standalone requests (e.g., MESSAGE) and to receive all responses (e.g., 200 OK) to that request via the Gm reference point. The difference between transaction procedures and session control procedures is that a dialog is not created.

### 2.3.2 Mw Reference Point

The Gm reference point links the UE to the IMS (namely, to the P-CSCF). Next, a SIP based reference point between different CSCFs is needed. This reference point is called Mw. The procedures in the Mw reference point can be divided into three main categories – registration, session control and transaction:

- In the registration procedure the P-CSCF uses the Mw reference point to forward a registration request from the UE to the I-CSCF. The I-CSCF then uses the Mw reference point to pass the request to the S-CSCF. Finally, the response from the S-CSCF traverses back via the Mw reference point. In addition, the S-CSCF uses the Mw reference point for network-initiated de-registration and network-initiated re-authentication procedures to inform the UE and P-CSCF about the event. Based on this information UE and P-CSCF can take necessary actions. For example, in case of network-initiated de-registration the P-CSCF can request PCRF to ensure that all transport plane resources are released accordingly.
- Session control procedures contain mechanisms for both mobile-originated sessions and mobile-terminated sessions. In mobile-originated sessions the Mw reference point is used to forward requests both from the P-CSCF to the S-CSCF and from the S-CSCF to the I-CSCF. In mobile-terminated sessions the Mw reference point is used to forward requests both from the I-CSCF to the S-CSCF and from the S-CSCF to the P-CSCF. This reference point is also used for network-initiated session releases: for example, the P-CSCF could initiate a session release toward the S-CSCF if it receives an indication from the PCRF that media bearer(s) are lost. In addition, charging-related information is conveyed via the Mw reference point.
- Transaction procedures are used to pass a standalone request (e.g., MESSAGE) and to receive all responses (e.g., 200 OK) to that request via the Mw reference point.

As already stated, the difference between transaction procedures and session control procedures is that a dialog is not created.

### 2.3.3 *IMS Service Control (ISC) Reference Point*

In the IMS architecture, Application Servers are entities that host and execute services, such as presence, messaging and multimedia telephony. Therefore, there has to be a reference point for sending and receiving SIP messages between S-CSCF and Application Server. This reference point is called the IMS Service Control (ISC) reference point and the selected protocol is SIP. ISC procedures can be divided into two main categories – routing the initial SIP request to an AS and AS-initiated SIP requests:

1. After successful IMS registration the S-CSCF downloads user profile (see Section 3.2) from HSS that contains among other things information when a request is to be sent to AS (see Section 3.12.4). Once the S-CSCF receives an initial SIP request it will analyze it. Based on the analysis the S-CSCF may decide to route the request to an AS for further processing. The AS may terminate, redirect or proxy the request from the S-CSCF.
2. Based on internal or external triggers an AS may initiate a request (e.g., on behalf of a user or on behalf of a service). For example, initiating timed instant messaging chat session or presence information publication from certain AS to Presence Server.

### 2.3.4 *Ma Reference Point*

The ISC reference allows communication towards Application Servers when there is need to execute service control in S-CSCF. Not all services require this therefore an optimized routing mechanism to bypass S-CSCF was introduced in Release 7. The Ma reference point is used to convey information directly between I-CSCF and AS. Typically this reference point is used when a target identity in the SIP request is IMS public service identity or AS initiates something on behalf of the service. See Section 12.11 for routing examples.

### 2.3.5 *Cx Reference Point*

Subscriber and service data are permanently stored in the HSS. These centralized data need to be utilized by the I-CSCF and the S-CSCF when the user registers or receives sessions. Therefore, there has to be a reference point between the HSS and the CSCF. This reference point is called the Cx reference point and the selected protocol is Diameter. The procedures can be divided into three main categories: location management, user data handling and user authentication. Table 2.1 summarizes the available Cx commands.

#### 2.3.5.1 **Location Management Procedures**

Location management procedures can be further divided in two groups: registration and de-registration and location retrieval.

When the I-CSCF receives a SIP REGISTER request from the P-CSCF via the Mw reference point it will invoke a user registration status query or, as it is known in the

**Table 2.1** Cx commands

Command-Name	Purpose	Abbreviation	Source	Destination
User-Authorization-Request/Answer	User-Authorization-Request/Answer (UAR/UAA) commands are used between the I-CSCF and the HSS during SIP registration for retrieving S-CSCF name or S-CSCF capabilities for S-CSCF selection and during SIP deregistration for retrieving S-CSCF name when the SIP method is REGISTER	UAR UAA	I-CSCF HSS	HSS I-CSCF
Server-Assignment-Request/Answer	Server-Assignment-Request/Answer (SAR/SAA) commands are used between the S-CSCF and the HSS to update the S-CSCF name to the HSS and to download the user profile data to the S-CSCF	SAR SAA	S-CSCF HSS	HSS S-CSCF
Location-Info-Request/Answer	Location-Info-Request/Answer (LIR/LIA) commands are used between the I-CSCF and the HSS during the SIP session set-up and transactions to obtain the name of the S-CSCF that is serving the user or S-CSCF capabilities for S-CSCF selection	LIR LIA	I-CSCF HSS	HSS I-CSCF
Multimedia-Auth-Request/Answer	Multimedia-Auth-Request/Answer (MAR/MAA) commands are used between the S-CSCF and the HSS to exchange information to support the authentication between the end user and the home IMS network	MAR MAA	S-CSCF HSS	HSS S-CSCF
Registration-Termination-Request/ Answer	Registration-Termination-Request/Answer (RTR/RTA) commands are used between the S-CSCF and the HSS when the HSS administratively de-registers one or more of the user's public identities	RTR RTA	HSS S-CSCF	S-CSCF HSS

**Table 2.1** (*continued*)

Command-Name	Purpose	Abbreviation	Source	Destination
Push-Profile-Request/Answer	Push-Profile-Request/Answer (PPR/PPA) commands are used between the HSS and the S-CSCF when user profile data are changed by a management operation in HSS and the data need to be updated to the S-CSCF	PPR PPA	HSS S-CSCF	S-CSCF HSS

standards, a User-Authorization-Request (UAR) command. After receiving the UAR command the HSS sends a User-Authorization-Answer (UAA) command. It contains the S-CSCF Name or S-CSCF Capabilities (if the UAR command does not fail due, say, to the private and public identities received in the request not belonging to the same user) depending on the user's current registration status. S-CSCF capabilities are returned if the user does not have an S-CSCF Name assigned yet in the HSS or if the I-CSCF explicitly requests S-CSCF capabilities. Otherwise, the S-CSCF name is returned. When capabilities are returned the I-CSCF needs to perform S-CSCF selection as described in Section 3.9.

We explained above how the I-CSCF finds an S-CSCF that will serve the user. Having done this, the I-CSCF forwards a SIP REGISTER request to the S-CSCF. When the S-CSCF receives the SIP REGISTER request from the I-CSCF it uses a Server-Assignment-Request (SAR) command to communicate with the HSS. The SAR command is used to inform the HSS about which S-CSCF will be serving the user when the expires value is not equal to zero. Similarly, if the expires value equals zero, then the SAR command is used to announce the fact that the S-CSCF is no longer serving a user. A precondition for sending the SAR command is that the user has been successfully authenticated by the S-CSCF. After receiving the SAR command the HSS will respond with a Server-Assignment-Answer (SAA) command. It contains among other things the User Profile and the addresses of the charging functions.

So far we have described how user-initiated registration procedures and user- or S-CSCF-initiated de-registration procedures are handled over the Cx reference point. There is still the need for additional operations to bring about network-initiated de-registration (e.g., due to a stolen UE or when a subscription is terminated). In this case it is the HSS that starts network-initiated de-registration by using a command called Registration-Termination-Request (RTR). The RTR command is acknowledged by a Registration-Termination-Answer (RTA) command, which simply indicates the result of the operation.

Previously, we have described how the I-CSCF uses a user registration status query (UAR command) to find the S-CSCF when it receives a SIP REGISTER request. Correspondingly, there has to be a procedure to find the S-CSCF when a SIP method is different from REGISTER. The required procedure is to make use of a Location-Info-Request (LIR) command. The HSS responds with a Location-Info-Answer (LIA) command. The response can contain the S-CSCF Name or S-CSCF Capabilities or an appropriate error response (e.g. user is not registered and no services available for un-registered state). The S-CSCF

name is returned when the S-CSCF has been previously assigned to serve the target user. The S-CSCF capabilities are returned if no S-CSCF is assigned but the user has services for the un-registered state.

### 2.3.5.2 User Data Handling Procedures

During the registration process, user and service-related data will be downloaded from the HSS to the S-CSCF via the Cx reference point using SAR and SAA commands as described earlier. However, it is possible for these data to be changed later when the S-CSCF is still serving a user. To update the data in the S-CSCF the HSS initiates a Push-Profile-Request (PPR) command. Update takes place immediately after the change with one exception: when the S-CSCF is serving an unregistered user or the S-CSCF is kept for an unregistered user as described in Section 3.9.5 and there is a change in the registered part of user profile, then the HSS will not send a PPR command. The PPR command is acknowledged by a Push-Profile-Answer (PPA) command, which simply indicates the result of the operation.

### 2.3.5.3 Authentication Procedures

IMS user authentication relies on a pre-configured shared secret. Shared secrets and sequence numbers are stored in the IP Multimedia Services Identity Module (ISIM) or UMTS subscriber identity module (USIM) in the UE and in the HSS in the network. Because S-CSCF takes care of user authorization, there exists the need to transfer security data over the Cx reference point. When the S-CSCF needs to authenticate a user it sends a Multimedia-Auth-Request (MAR) command to the HSS. The HSS responds with a Multimedia-Auth-Answer (MAA) command. The answer contains among other information Authentication Data. It includes one or more authentication vector, which is comprised of an Authentication Scheme (e.g., Digest-AKA<sup>v1</sup>-MD5), Authentication Information (authentication challenge RAND and the token AUTN), Authorization Information (expected response, or XRES), Integrity Key and a Confidentiality Key. Additionally, it contains an Item Number, which indicates the order in which the authentication vectors are to be consumed when multiple vectors are returned. The usage of these elements is further described in Section 11.6.

### 2.3.6 Dx Reference Point

When multiple and separately addressable HSSs have been deployed in a network, neither the I-CSCF nor the S-CSCF know which HSS they need to contact. However, they need to contact the SLF first. For this purpose the Dx reference point has been introduced. The Dx reference point is always used in conjunction with the Cx reference point. The protocol used in this reference point is based on Diameter. Its functionality is implemented by means of the routing mechanism provided by an enhanced Diameter redirect agent.

To get an HSS address the I-CSCF or the S-CSCF sends to the SLF the Cx requests aimed for the HSS. On receipt of the HSS address from the SLF, the I-CSCF or the S-CSCF will send the Cx requests to the HSS. Figure 2.13 shows how the SLF is used to find a correct HSS when the I-CSCF receives an INVITE request and three HSSs have been deployed.

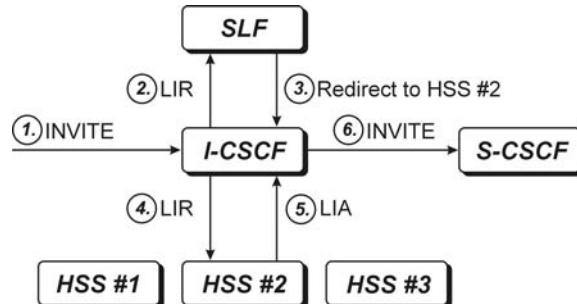


Figure 2.13 HSS resolution using the SLF

### 2.3.7 Sh Reference Point

An AS (SIP AS or OSA SCS) may need data (related to particular identity of user or related to public service identity) or need to know to which S-CSCF to send a SIP request. This type of information is stored in the HSS. Therefore, there has to be a reference point between the HSS and the AS. This reference point is called the Sh reference point and the protocol is Diameter. Procedures are divided into two main categories: data handling and subscription/notification. Table 2.2 summarizes the available Sh commands. The HSS maintains a list of ASs that are allowed to obtain or store data.

Table 2.2 Sh commands

Command-Name	Purpose	Abbreviation	Source	Destination
User-Data-Request/Answer	User-Data-Request/Answer (UDR/UDA) commands are used to deliver the user data of a particular user	UDR UDA	AS HSS	HSS AS
Profile-Update-Request/Answer	Profile-Update-Request/Answer (PUR/PUA) commands are used to update transparent data in the HSS	PUR PUA	AS HSS	HSS AS
Subscribe-Notifications-Request/Answer	Subscribe-Notifications-Request/Answer commands are used to make a subscription/ cancel a subscription to user's data on which notifications of change are required	SNR SNA	AS HSS	HSS AS
Push-Notification-Request/Answer	Push-Notification-Request/Answer commands are used to send the changed data to the AS	PNR PNA	HSS AS	AS HSS

### 2.3.7.1 Data Handling

Data-handling procedures contain the possibility of retrieving data from the HSS. Such data can contain service-related data (transparent or non-transparent), registration information, public user identities, initial filter criteria, the S-CSCF Name serving the user or public user identity, addresses of the charging functions and even location information from the CS and PS domains. Transparent data are understood syntactically but not semantically by the HSS. They are data that an AS may store in the HSS to support its service logic. On the contrary, nontransparent data are understood both syntactically and semantically by the HSS. The AS uses the User-Data-Request (UDR) command to request data. The request contains information about the request data. The HSS responds with the User-Data-Answer (UDA).

The AS can update transparent data in the HSS using the Profile-Update-Request (PUR) command, which contains data to be updated. The PUR command is acknowledged by a Profile-Update-Answer (PUA) command, which simply indicates the result of the operation.

### 2.3.7.2 Subscription/Notification

Subscription/Notification procedures allow the AS to get a notification when particular data for a specific user is updated in the HSS. The AS sends a Subscribe-Notifications-Request (SNR) command to receive a notification of when data indicated in the SNR command are changed in the HSS. The HSS acknowledges the subscription request by a Subscribe-Notifications-Answer (SNA) command, which simply indicates the result of the operation.

If the AS has sent the SNR command and requested a notification with subscription request type, then the HSS sends a Push-Notification-Request (PNR) command to the AS when particular data have changed giving details of the changed data. The PNR command is acknowledged by a Push-Notification-Answer (PNA) command, which simply indicates the result of the operation.

### 2.3.8 *Dh Reference Point*

When multiple and separately addressable HSSs have been deployed in the network, the AS cannot know which HSS it needs to contact. However, the AS needs to contact the SLF first. For this purpose the Dh reference point was introduced in Release 6. In Release 5 the correct HSS is discovered by using proprietary means. The Dh reference point is always used in conjunction with the Sh reference point. The protocol used in this reference point is based on Diameter. Its functionality is implemented by means of the routing mechanism provided by an enhanced Diameter re-direct agent.

To get an HSS address, the AS sends to the SLF the Sh request aimed for the HSS. On receipt of the HSS address from the SLF, the AS will send the Sh request to the HSS.

### 2.3.9 *Si Reference Point*

When the AS is a CAMEL AS (IM-SSF) it uses the Si reference point to communicate with the HSS. The Si reference point is used to transport CAMEL subscription information

**Table 2.3** Summary of reference points

Name of reference point	Involved entities	Purpose	Protocol
Gm	UE, P-CSCF	This reference point is used to exchange messages between UE and CSCFs	SIP
Mw	P-CSCF, I-CSCF, S-CSCF	This reference point is used to exchange messages between CSCFs	SIP
ISC	S-CSCF, AS	This reference point is used to exchange messages between S-CSCF and AS	SIP
Ma	I-CSCF, AS	This reference point is used to exchange messages between I-CSCF and AS	SIP
Cx	I-CSCF, S-CSCF, HSS	This reference point is used to communicate between I-CSCF/ S-CSCF and HSS	Diameter
Dx	I-CSCF, S-CSCF, SLF	This reference point is used by I-CSCF/S-CSCF to find a correct HSS in a multi-HSS environment	Diameter
Sh	SIP AS, OSA SCS, HSS	This reference point is used to exchange information between SIP AS/OSA SCS and HSS	Diameter
Si	IM-SSF, HSS	This reference point is used to exchange information between IM-SSF and HSS	MAP
Dh	SIP AS, OSA, SCF, IM-SSF, HSS	This reference point is used by AS to find a correct HSS in a multi-HSS environment	Diameter
Mm	I-CSCF, S-CSCF, IBCF, external IP network	This reference point will be used for exchanging messages between IMS and external IP networks	SIP
Mg	MGCF--> I-CSCF	This reference point is used to exchange messages between MGCF and I-CSCF	SIP
Mi	S-CSCF --> BGCF	This reference point is used to exchange messages between S-CSCF and BGCF	SIP
Mj	BGCF --> MGCF	This reference point is used to exchange messages between BGCF and MGCF in the same IMS network	SIP
Mk	BGCF --> BGCF	This reference point is used to exchange messages between BGCFs in different IMS networks	SIP
Mr	S-CSCF, MRFC	This reference point is used to exchange messages between S-CSCF and MRFC	SIP
Mp	MRFC, MRFP	This reference point allows control of user-plane resources of MRFP	H.248
Mn	MGCF, IMS-MGW	This reference point allows control of user-plane resources of IMS-MGW	H.248

(continued overleaf)

**Table 2.3** (*continued*)

Name of reference point	Involved entities	Purpose	Protocol
Ut	UE, AS (SIP AS, OSA SCS, IM-SSF)	This reference point enables UE to manage information related to his services	HTTP
Gx	PCRF, Access Gateway	This reference point is used to push policy and charging rules to access gateway, to get transport plane event notifications and to exchange charging identifiers	Diameter
Rx	P-CSCF, PCRF	This reference point is used to pass registration and session information to PCRF and to get transport plane event notifications and to exchange charging identifiers	Diameter
Ro	AS, MRCF, S-CSCF, OCS	This reference point is used by AS/MRFC/S-CSCF for online charging towards OCS. Note: there might exist an interworking function between the S-CSCF and OCS	Diameter
Rf	P-CSCF, S-CSCF, I-CSCF, BGCF, MGCF, AS, MRFC, IBCF, CDF	This reference point is used by IMS entities for offline charging towards CDF	Diameter
Ml	E-CSCF, LRF	This reference point is used to exchange necessary information for routing request to an emergency centre	Not specified
Mx	CSCF, BGCF, IBCF	This reference point is targeted to use capabilities of IBCF when communicating with different operator	Not specified
Ix	IBCF, TrGW	This reference point allows control of TrGW resources	Not specified
Iq	P-CSCF, IMS access gateway	This reference point allows control of IMS access gateway (NAT)	Not specified

including triggers from the HSS to the IM-SSF. The used protocol is Mobile Application Part (MAP).

### 2.3.10 Mi Reference Point

When the S-CSCF or E-CSCF discover that a session needs to be routed to the CS domain it uses the Mi reference point to forward the session to BGCF. The protocol used for the Mi reference point is SIP. Section 3.14 contains further details about IMS–CS interworking.

### 2.3.11 *Mj Reference Point*

When BGCF receives session signalling via the Mi reference point it selects the CS domain in which breakout is to occur. If breakout occurs in the same network, then it forwards the session to MGCF via the Mj reference point. The protocol used for the Mj reference point is SIP. Section 3.14 contains further details about IMS–CS interworking.

### 2.3.12 *Mk Reference Point*

When BGCF receives session signalling via the Mi reference point it selects the CS domain in which breakout is to occur. If the breakout occurs in another network, then it forwards the session to BGCF in the other network via the Mk reference point. The protocol used for the Mk reference point is SIP. Section 3.14 contains further details about IMS–CS interworking.

### 2.3.13 *Mg Reference Point*

The Mg reference point links the CS edge function, MGCF, to IMS (namely, to the I-CSCF). This reference point allows MGCF to forward incoming session signalling from the CS domain to the I-CSCF. The protocol used for the Mg reference point is SIP. MGCF is responsible for converting incoming ISUP signalling to SIP.

### 2.3.14 *Mm Reference Point*

For communicating with other multimedia IP networks, a reference point between the IMS and other multimedia IP networks is needed. The Mm reference point allows the I-CSCF to receive a session request from another SIP server or terminal. Similarly, the S-CSCF uses the Mm reference point to forward IMS UE-originated requests to other multimedia networks. At the time of writing, a detailed specification of the Mm reference point has not been provided. However, it is very likely that the protocol would be SIP.

### 2.3.15 *Mr Reference Point*

When the S-CSCF needs to activate bearer-related services it passes SIP signalling to the MRFC via the Mr reference point. The functionality of the Mr reference point is not fully standardized: for example, it is not specified how the S-CSCF informs the MRFC to play a specific announcement. The used protocol in the Mr reference point is SIP.

### 2.3.16 *Mp Reference Point*

Media server architecture in IMS consists of two entities MRFC and MRFP. These two entities are connected via the Mp reference point. Over this reference point the MRFC is able to ask MRFP to do the following things:

- play tone to user or number of users;
- play announcement to user or number of users e.g. ‘person you try to reach is currently out of coverage or not able to receive multimedia communication’;
- generate speech output from text or annotated text input;

- record audio or multimedia stream(s) and store it into a file. The function can be used in some services, such as the voice mail box service, conference service, etc;
- collect and report dialled DTMF digits e.g. to get PIN code for voice mail box;
- perform automatic speech recognition and report the results;
- play synchronized audio and video media streams to the user. The function can be used in the services, such as multimedia announcement, multimedia mail box service, etc;
- provide conferencing transport plane capabilities for audio and multimedia conferencing service;
- transcoding of audio and video streams.

The protocol on the Mp interface is defined to comply with ITU-T H.248.1 Gateway Control Protocol. 3GPP has defined a formal Profile within the H.248 protocol toolbox and it is documented in 3GPP TS 23.333 and 3GPP TS 29.333.

### *2.3.17 Mn Reference Point*

The Mn interface is the control reference point between the MGCF and IMS-MGW. The Mn interface controls the user plane between IP access and IMS-MGW (Mb reference point). Also, it controls the user plane between CS access (Nb and TDM interfaces) and IMS-MGW. The Mn interface is based on H.248 and is equivalent to the usage (encoding, decoding, etc.) of the Mc interface specified to control the CS-MGW. The difference between these two interfaces is that the Mn interface introduces new H.248 procedures for handling IP access end termination and also some additional procedures for CS end termination handling. The H.248 is primarily used to perform the following tasks:

- creating or releasing connection between IMS and CS user;
- connecting or releasing tones to end-point;
- connecting or releasing announcements to end-point e.g. ‘subscriber you try to reach is speaking on phone at the moment please hold’;
- sending or receiving DTMF tones e.g. PIN code to verify yourself to the banking service;
- transcoding of audio and video streams.

### *2.3.18 Gx Reference Point*

It is in operators’ interests to ensure that IMS session information is correctly taken in use in transport level and IMS entities gets information about main transport level events. For this purpose Diameter based Gx reference point between PCRF and access gateway (e.g. GGSN) was developed. The main procedures supported over this reference point are:

- delivering gating control ‘Firewall’ instructions i.e. how incoming and outgoing packets should be treated in access gateway;
- passing instructions what kind of QoS treatment should be applied for particular IP flows;
- reporting traffic plane event (e.g. bearer is released or lost) as well as the reporting of events related to the resources in the access gateway;

- mechanism to learn if UE and access network entities supports network initiated IP-CAN bearer setup and requesting the access gateway to initiate such bearer towards UE with desired gating and QoS control values;
- capability for exchange of IMS charging identifier and access charging identifier;
- distribution of primary and secondary addresses of offline and online charging entity addresses to the access gateway;
- activation of online and offline charging in access gateway (enabled/disabled);
- metering-method to be applied in access gateway (duration, volume or both);
- rating group information (e.g. 0.1€ per minute);
- desired reporting level in access gateway (based on given service or based on given service and rating-group).

### 2.3.19 Rx Reference Point

When policy and charging control is used in the network the P-CSCF sends information obtained from SIP/SDP session setup signalling to the PCRF via the Rx reference point. This information enables the PCRF to form authorized IP QoS data (e.g. maximum bandwidth and QoS class) and charging rules (e.g. 0.1€/minute) that will be delivered to the access gateway via the Gx reference point. The P-CSCF is tasked to send policy information to the PCRF about every SIP message that includes an SDP payload. This ensures that the PCRF passes the proper information to perform policy and charging control for all possible IMS session setup scenarios. The Rx usage for policy control is further described in Section 3.10.5 and for charging purposes in Section 3.11.6.3.

### 2.3.20 Charging Reference Points

Charging-related reference points Rf, Ro, Rx and Gx are described in Sections 3.11.6.1, 3.11.6.2, 3.11.5.3 and 3.11.5.4.

### 2.3.21 Mx, Ix and Iq Reference Point

These three reference points are defined in Release 7 architecture but protocol specifications do not exist.

The Mx reference point is targeted to enable communication between CSCFs/BGCF and Interconnection Border Control Function (IBCF) to use capabilities of IBCF such as IP version interworking and network topology hiding functionality. IBCF is further expected to control TrGW via the Ix reference point. The Iq reference point instead should enable P-CSCF to control IMS access gateway.

### 2.3.22 Ml Reference Point

This reference point is used for IMS emergency sessions. E-CSCF uses it when it needs to verify UE provided location information or to obtain location information or to obtain routing information to emergency centres from LRF. Protocol design for this reference point does not exist in 3GPP Release 7.

### 2.3.23 *Ut Reference Point*

The Ut reference point is the reference point between the UE and the AS. It enables users to securely manage and configure their network services related information hosted on an AS. Users can use the Ut reference point to create Public Service Identities (PSIs), such as a resource list, and manage the authorization policies that are used by the service. Examples of services that utilize the Ut reference point such as presence, Push to talk Over Cellular and Multimedia telephony. The XML Configuration Access Protocol (XCAP), as defined by Internet Engineering Task Force (IETF) is the chosen data protocol for the Ut reference point. Usage of the Ut reference point is described in more detail in Chapter 5.

# 3

# IMS Concepts

## 3.1 Overview

This chapter begins with a first-glance description of IP Multimedia Subsystem (IMS) registration and session establishment. It depicts the IMS entities that are involved. The intention is not to show a full-blown solution; rather, it is to give an overview and help the reader to understand the different IMS concepts explained in this chapter. Detailed registration and session establishment flows will be shown and explained later in the book.

Prior to IMS registration the User Equipment (UE) must discover the IMS entity to which it will send a REGISTER request. This concept is called a Proxy-Call Session Control Function (P-CSCF) discovery and is described in Section 3.8. In addition, before a registration process the UE needs to fetch user identities from identity module. Identity module is covered in Section 3.6 and identities are presented in Section 3.5. During the registration a Serving-CSCF (S-CSCF) will be assigned (Section 3.9), authentication will be performed and corresponding security associations will be established (Section 3.21), a user profile (Section 3.12) will be downloaded to the assigned S-CSCF, Session Initiation Protocol (SIP) compression will be initialized (Section 3.18) and implicitly registered public user identities will be delivered (Section 3.3). The concept of sharing a single user identity between multiple terminals is covered in Section 3.7.

Section 3.10 explains how Internet Protocol (IP) policy control is applied when a user is establishing a session, and Section 3.13 shows how services can be provisioned. Section 3.11 shows how an operator is able to charge a user. Section 3.17 describes IMS emergency session special procedures and usage of.

Interactions with Circuit Switched (CS) media component and networks are covered in several sections. Interworking with the Circuit Switched (CS) network is briefly described in Section 3.14. Simultaneous usage of CS and Packet Switched (PS) media components are covered in Section 3.19. Capability to offer mobile voice services for a multi-radio device user via both CS and PS/IMS access as preferred is covered in Section 3.20.

In addition, IP version interworking (Section 3.22), IMS local dialling 3.16 and IMS Transit (Section 3.15) are covered.

### 3.2 Registration

Prior to IMS registration, which allows the UE to use IMS services, the UE must obtain an IP connectivity bearer and discover an IMS entry point (i.e., the P-CSCF); for example, in the case of General Packet Radio Service (GPRS) access the UE performs the GPRS attach procedure and activates a Packet Data Protocol (PDP) context for SIP signalling. Section 12.12 gives a short overview of the PDP context, and P-CSCF discovery is explained in Section 3.8. This book does not describe the GPRS attach procedure (for further information see [3GPP TS 23.060]).

IMS registration contains two phases: the left hand part of Figure 3.1 shows the first phase – how the network challenges the UE. The right hand part of Figure 3.1 shows the second phase – how the UE responds to the challenge and completes the registration. First, the UE sends a SIP REGISTER request to the discovered P-CSCF. This request would contain, say, an identity to be registered and a home domain name (address of the Interrogating-CSCF, or I-CSCF). The P-CSCF processes the REGISTER request and uses the provided home domain name to resolve the IP address of the I-CSCF. The I-CSCF, in turn, will contact the Home Subscriber Server (HSS) to fetch the required capabilities for S-CSCF selection. After S-CSCF selection the I-CSCF forwards the REGISTER request to the S-CSCF. The S-CSCF realizes that the user is not authorized and, therefore, retrieves authentication data from the HSS and challenges the user with a 401 Unauthorized response. Second, the UE will calculate a response to the challenge and send another REGISTER request to the P-CSCF. Again the P-CSCF finds the I-CSCF and the I-CSCF, in turn, will find the S-CSCF. Finally, the S-CSCF checks the response and, if it is correct, downloads a user profile from the HSS and accepts the registration with a 200 OK response. Once the UE is successfully authorized, the UE is able to initiate and receive sessions. During the registration procedure both the UE and the P-CSCF learn which S-CSCF in the network will be serving the UE.

It is the UE's responsibility to keep its registration active by periodically refreshing its registration. If the UE does not refresh its registration, then the S-CSCF will silently remove the registration when the registration timer lapses. When the UE wants

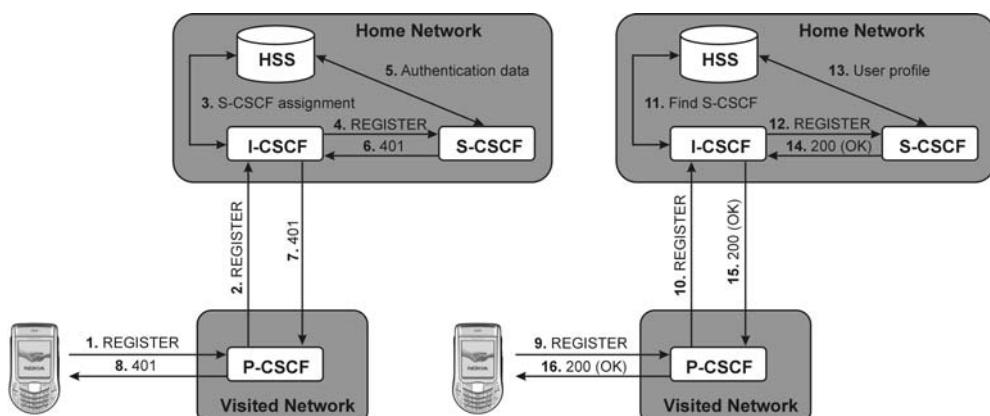


Figure 3.1 High-level IMS registration flow

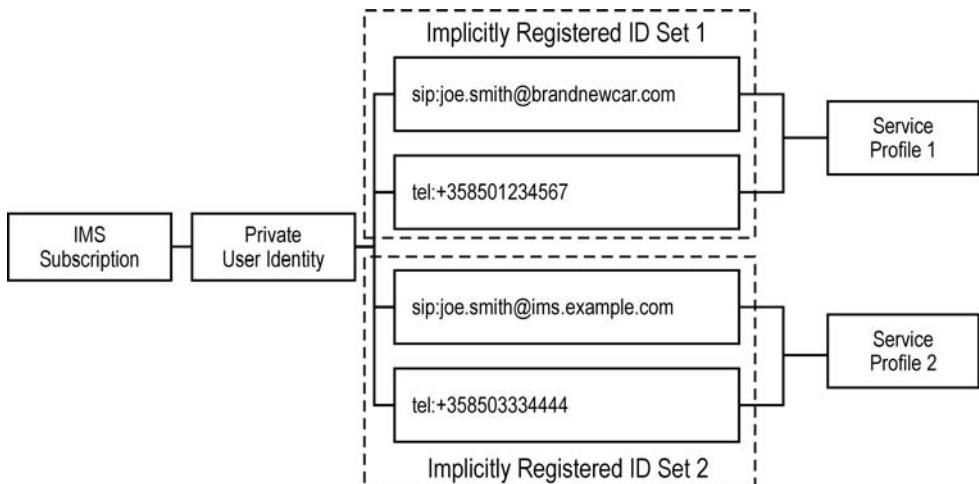
to de-register from the IMS it sets a registration timer to 0 and sends a REGISTER request. Sections 11.5 and 11.15 contain more detailed descriptions of IMS registration and de-registration.

### 3.3 Mechanism to Register Multiple User Identities at a Go

SIP allows one public user identity to be registered at a time; so, if a user has more than one public user identity, then she has to register every public user identity individually. This may be frustrating and time consuming from the end user perspective. Obviously, registering four public user identities would consume four times as much resource than registering one public user identity. It was for these reasons that Third Generation Partnership Project (3GPP) developed a mechanism to register more than one public user identity at a time. This concept is called “implicit registration”.

An implicit registration set is a group of public user identities that are registered via a single registration request. When one of the public user identities within the set is registered, all public user identities associated with the implicit registration set are registered at the same time. Similarly, when one of the public user identities within the set is de-registered, all public user identities that have been implicitly registered are de-registered at the same time. Public user identities belonging to an implicit registration set may point to different service profiles. Some of these public user identities may point to the same service profile [3GPP TS 23.228].

For example, a user has four public user identities that are grouped in two implicit registration sets (Figure 3.2). The first set contains `sip:joe.smith@brandnewcar.com` and `tel:+358501234567`. The second set contains `sip:joe.smith@ims.example.com` and `tel:+358503334444`. When Joe sends a REGISTER request containing `joe.smith@brandnewcar.com` as an identity to be registered, the allocated S-CSCF performs a normal registration procedure and, after successful authorization, the S-CSCF downloads the service profile



**Figure 3.2** Example of implicit registration sets

associated with this identity. The service profile includes information that the other identity, tel:+358501234567 belongs to the implicit registration set with sip:joe.smith@ims.example.com and therefore it gets activated as well. When the S-CSCF accepts the registration of sip:joe.smith@ims.example.com (SIP 200 OK in Figure 3.1) it will inform the UE that also tel:+358501234567 got registered. Similarly when public user identity sip:joe.smith@ims.example.com is registered network automatically registers tel:+358503334444 as well.

### 3.4 Session Initiation

When User A wants to have a session with User B, UE A generates a SIP INVITE request and sends it via the Gm reference point to the P-CSCF. The P-CSCF processes the request: for example, it decompresses the request and verifies the originating user's identity before forwarding the request via the Mw reference point to the S-CSCF. The S-CSCF processes the request, executes service control which may include interactions with Application Servers (ASs) and eventually determines the entry point of the home operator of User B based on User B's identity in the SIP INVITE request. The I-CSCF receives the request via the Mw reference point and contacts the HSS over the Cx reference point to find the S-CSCF that is serving User B. The request is passed to the S-CSCF via the Mw reference point. The S-CSCF takes charge of processing the terminating session, which may include interactions with ASs and eventually delivers the request to the P-CSCF over the Mw reference point. After further processing (e.g., compression and privacy checking), the P-CSCF uses the Gm reference point to deliver the SIP INVITE request to UE B. UE B generates a response – 183 Session Progress – which traverses back to UE A following the route that was created on the way from UE A (i.e., UE B –> P-CSCF –> S-CSCF –> I-CSCF –> S-CSCF –> P-CSCF –> UE A) (Figure 3.3). After a few more round trips, both sets of UE complete session establishment and are able to start the

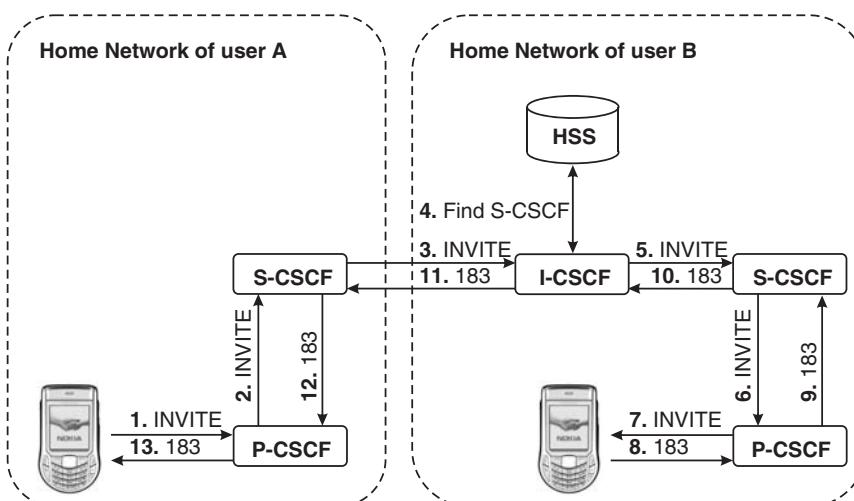


Figure 3.3 High-level IMS session establishment flow

actual application (e.g., a game of chess). During session establishment an operator may control the usage of bearers intended for media traffic. Section 3.10 explains how this can be done. Chapter 12 contains a more detailed description of IMS session initiation.

## 3.5 Identification

This section introduces different identifiers that are used to identify a user (public user identity), user's subscription (private user identity), user's device and public user identity combination (globally routable user agent URI), service (public service identity) and IMS network entities. In addition the relationship between different user identities is explained.

### 3.5.1 Public User Identity

User identities in IMS networks are called public user identities. They are the identities used for requesting communication with other users. Public identities can be published (e.g., in phone books, Web pages, business cards).

IMS users will be able to initiate sessions and receive sessions from many different networks, such as GSM networks and the Internet. To be reachable from the CS side, the public user identity must conform to telecom numbering (e.g., +358501234567). In similar manner, requesting communication with Internet clients, the public user identity must conform to Internet naming (e.g., joe.doe@example.com).

The IMS architecture imposes the following requirements for public user identity [3GPP TS 23.228, TS 23.003]:

- The public user identity/identities will take the form of either a SIP Uniform Resource Identifier (URI) or a telephone Uniform Resource Locator (tel URL) format.
- At least one public user identity will be securely stored in an ISIM application.
- It will not be possible for the UE to modify the public user identity stored in ISIM application.
- A public user identity will be registered before the identity can be used to originate IMS sessions and IMS session-unrelated procedures (e.g., MESSAGE, SUBSCRIBE, NOTIFY).
- A public user identity will be registered before terminating IMS sessions, and terminating IMS session-unrelated procedures will be delivered to the UE of the user that the public user identity belongs to. Subscriber-specific services for unregistered users may nevertheless be executed.
- It will be possible to register multiple public user identities through one single UE request. This is described further in Section 3.3.
- The network will not authenticate public user identities during registration.

The tel URL scheme is used to express traditional E.164 numbers in URL syntax. The tel URL is described in [RFC3966], and the SIP URI is described in [RFC3261] and [RFC2396]. Examples of public user identities are given below.

Example of SIP URI	sip:joe.doe@ims.example.com
--------------------	-----------------------------

<b>Example of tel URL</b>	<b>tel:+358 50 1234567</b>
---------------------------	----------------------------

### 3.5.2 Private User Identity

The private user identity is a unique global identity defined by the home network operator, which may be used within the home network to uniquely identify the user from a network perspective [3GPP TS 23.228]. It does not identify the user herself; on the contrary, it identifies the user's subscription. Therefore, it is mainly used for authentication purposes. It is possible to utilize private user identities for accounting and administration purposes as well. The IMS architecture imposes the following requirements for private user identity [3GPP TS 23.228, TS 23.003]:

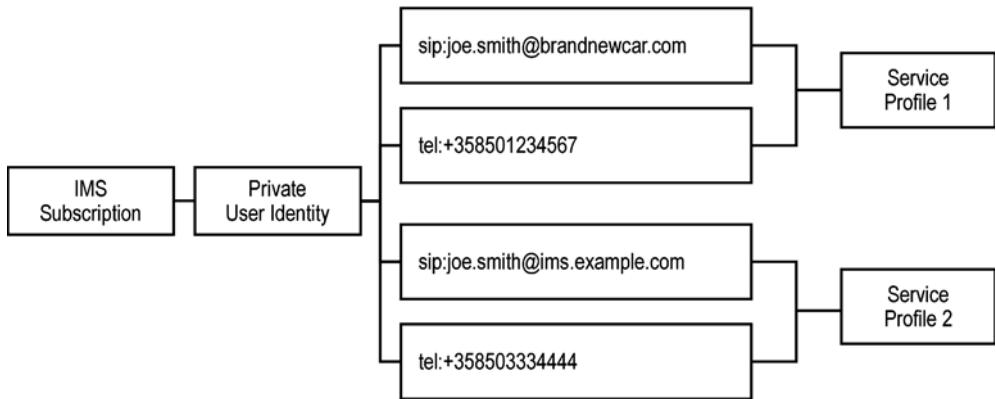
- The private user identity will take the form of a Network Access Identifier (NAI) defined in [RFC2486].
- The private user identity will be contained in all registration requests passed from the UE to the home network.
- The private user identity will be authenticated only during registration of the user (including re-registration and de-registration).
- The S-CSCF will need to obtain and store the private user identity on registration and on unregistered termination.
- The private user identity will not be used for routing of SIP messages.
- The private user identity will be permanently allocated to a user and securely stored in an IMS Identity Module (ISIM) application. The private user identity will be valid for the duration of the user's subscription within the home network.
- It will not be possible for the UE to modify the private user identity stored in an ISIM application.
- The HSS will need to store the private user identity.
- The private user identity will optionally be present in charging records based on operator policies.

<b>Example of NAI</b>	<b>private_user1@home1.operator.net</b>
-----------------------	---

### 3.5.3 Relationship between Private and Public User Identities

Here two basic examples show how different identities are linked to each other. In example one Joe is working for a car sales company and is using a single terminal for both his work life and his personal life. To handle work-related matters he has two public user identities: `sip:joe.smith@brandnewcar.com` and `tel:+358501234567`. When he is off-duty he uses two additional public user identities to manage his personal life: `sip:joe.smith@ims.example.com` and `tel:+358503334444`. By having two sets of public user identities he could have a totally different treatment for incoming sessions: for example, he is able to direct all incoming work-related sessions to a messaging system after 5 pm and during weekends and holidays.

Joe's user and service-related data are maintained in two different service profiles. One service profile contains information about his work life identities and is downloaded to



**Figure 3.4** Relationship of user identities

the S-CSCF from the HSS when needed: that is, when Joe registers a work life public user identity or when the S-CSCF needs to execute unregistered services for a work life public user identity. Similarly, another service profile contains information about his personal life identities and is downloaded to the S-CSCF from the HSS when needed. The concept of service profile is explained in Section 3.12.

Figure 3.4 shows how Joe's private user identity, public user identities and service profiles are linked together.

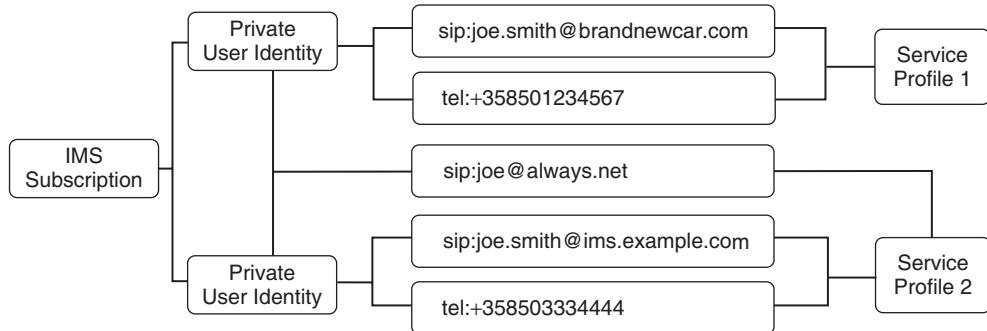
In example two Joe is using two devices<sup>1</sup> and in addition to identities listed in the previous example he is having one additional identity `sip:joe@always.net` that is shared to all his devices. It means that when someone uses this shared identity to initiate communication Joe can accept the incoming communication with any of his registered devices. According to IMS architecture shared public used identity must be shared with all private user identities within the IMS subscription [3GPP TS 23.228]. Figure 3.5 depicts this configuration (`sip:joe@always.net` could also point to own service profile if desired; here it is linked to service profile number 2).

### 3.5.4 Identity Generation Without ISIM

In Sections 3.5.1 and 3.5.2 the concepts of public user identity and private user identity have been explained. It was stated that these identities are stored in an ISIM application. When the IMS is deployed there will be a lot of UE in the market place that does not support the ISIM application; therefore, a mechanism to access the IMS without the ISIM was developed.

In this model, private user identity, public user identity and home domain name are derived from an International Mobile Subscriber Identifier (IMSI). This mechanism is suitable for UE that has a Universal Subscriber Identity Module (USIM) application.

<sup>1</sup> Each device must use different private user identity for registration and authentication. For example private user identity 1 is obtained from ISIM in device 1 and private user identity 2 is obtained from ISIM in device 2.



**Figure 3.5** Relationship between user identities including shared identity

### 3.5.4.1 Derived Private User Identity

The private user identity derived from the IMSI is built according to the following steps [3GPP TS 23.003]:

1. The user part of the private user identity is replaced with the whole string of digits from IMSI.
2. The domain part of the private user identity is composed of the MCC and MNC values of IMSI and has a pre-defined domain name, IMSI.3gppnetwork.org. These three parts are merged together and separated by dots in the following order: Mobile Network Code (MNC, a digit or a combination of digits uniquely identifying the public land mobile network), Mobile Country Code (MCC, code uniquely identifying the country of domicile of the mobile subscriber) and pre-defined domain name. For example:

```

IMSI in use: 23415099999999; where:
MCC: 234;
MNC: 15;
MSIN: 0999999999; and
Private user identity is:
23415099999999@234.15.IMSI.3gppnetwork.org
  
```

### 3.5.4.2 Temporary Public User Identity

If there is no ISIM application to host the public user identity, a temporary public user identity will be derived, based on the IMSI. The temporary public user identity will take the form of a SIP URI, ‘sip:user@domain’. The user and domain part are derived similarly from the method used for private user identity [3GPP TS 23.003]. Following our earlier example a corresponding temporary public user identity would be:

`sip:23415099999999@234.15.IMSI.3gppnetwork.org`

The IMS architecture imposes the following requirements for a temporary public user identity [3GPP TS 23.228]:

- It is strongly recommended that the temporary public user identity is set to ‘barred’ for IMS non-registration procedures so that it cannot be used for IMS communication. The following additional requirements apply if the temporary public user identity is ‘barred’:
  - the temporary public user identity will not be displayed to the user and will not be used for public usage (e.g., displayed on a business card);
  - the temporary public user identity will only be used during the registration to obtain implicitly registered public user identities (the concept of implicitly registered public user identities is explained in Section 3.3).
- Implicitly registered public user identities will be used for session handling, in other SIP messages and at subsequent registration processes.
- After the initial registration, only the UE will use the implicitly registered public user identity(s).
- The temporary public user identity will only be available to CSCF and HSS nodes.

### 3.5.5 Identification of Services (Public Service Identities)

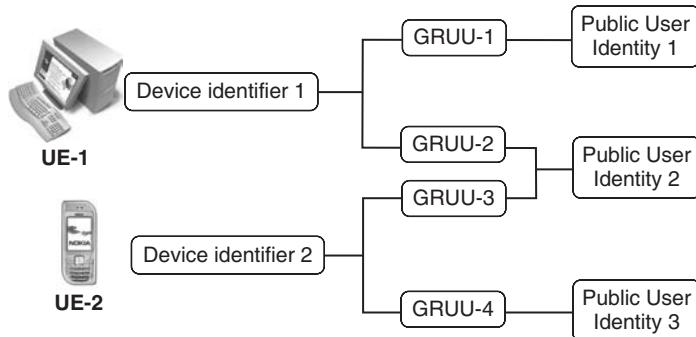
With the introduction of standardized presence, messaging, conferencing and group service capabilities it became evident that there must be identities to identify services and groups that are hosted by ASs. Identities for these purposes are also created on the fly: that is, they may be created by the user on an as-needed basis in the AS and are not registered prior to usage. Ordinary public user identities were simply not good enough; so, Release 6 introduced a new type of identity, the public service identity. Public service identities take the form of a SIP URI or are in tel URL format. Chapter 5 shows the number of use cases where user dynamically creates a new public service identity which they will use later, e.g. to initiate Push to talk Over Cellular group communication or Instant Messaging group communication or make subscription to their presence buddylist. For example, in messaging services there could be a public service identity for the messaging list service (e.g., `sip:messaginglist.joe@ims.example.com`) to which the users send messages and then the messages are distributed to other members on the messaging list by the Instant Messaging Application Server.

### 3.5.6 Identification of User’s Device

In the IMS public user identities are used to reach the recipient and single public user identity can be shared among a number of devices under single subscription (see Sections 3.5.3 and 3.7). This means that it is not possible to identify a specific device when more than one device has been registered with the same public user identity.

To reach a particular device a specific identifier called a Globally Routable User Agent URI (GRUU) must be used. For example, user Joe has a shared public user identity and his presence status indicates that he is willing to play games with UE1 and and he is willing to accept a video session with UE2 then the GRUU of UE1 (GRUU2 in Figure 3.6) can be used to establish a game session with Joe. Another typical use case for GRUU usage is session transfer from one device to a another particular device. Figure 3.6 shows relationship between UE, GRUU and Public User Identity.

Two types of GRUU are defined: temporary GRUU and public GRUU. Public GRUU in IMS is a combination of user’s public user identity and identifier of the device from



**Figure 3.6** Relationship between UE, GRUU and Public User Identities

which the public user identity is registered to the IMS. The purpose of public GRUU is to enable long lived capability to reach a particular device as it remains the same as long as public user identity-device pair exists. In contrast, temporary GRUU is an identifier that has limited life time (new temporary GRUU is created in each IMS registration request) and it keeps the user's public user identity hidden i.e. it does not contain the user's public user identity at all. In the example below `sip:joe@always.net` is the user's public user identity; 'gr' is URI parameter indicating that this SIP URI is actually GRUU, 'urn:uuid' is a Uniform Resource Name (URN) that uniquely identifies this specific device where `f81d4fae-7dec-11d0-a765-00a0c91e6bf6` is the unique value identifying particular device.

<b>Example of public GRUU</b>	<code>sip:joe@always.net; gr = urn:uuid:f81d4fae-7dec-11d0-a765-00a0c91e6bf6</code>
<b>Example of temporary GRUU</b>	<code>sip:tgruu.7hs == jd7vnzga5w7fajsc7-ajd6fabz0f8g5@example.com;gr</code>

Section 11.13.5 will explain how a device gets GRUUs and Section 12.10 describes how GRUU can be used in IMS communication.

### 3.5.7 Identification of Network Entities

In addition to users, network nodes that handle SIP routing need to be identifiable using a valid SIP URI. These SIP URIs would be used when identifying these nodes in the header fields of SIP messages. However, this does not require that these URIs be globally published in a Domain Name System (DNS) [3GPP TS 23.228]. An operator could name its S-CSCF as follows:

<b>Example of network entity naming</b>	<code>sip:finland.scscf1@ims.example.com</code>
---	---

### 3.6 IP Multimedia Services Identity Module (ISIM)

An IP Multimedia Services Identity Module (ISIM) is an application residing on the Universal Integrated Circuit Card (UICC), which is a physically secure device that can be inserted and removed from UE. There may be one or more applications in the UICC e.g. Universal Subscriber Identity Module (USIM) and ISIM. The ISIM itself stores IMS-specific subscriber data mainly provisioned by an IMS operator. This data is mainly used when a user registers a device to the IMS. The following data can be stored in ISIM e.g. when a user obtains an IMS subscription from an operator.

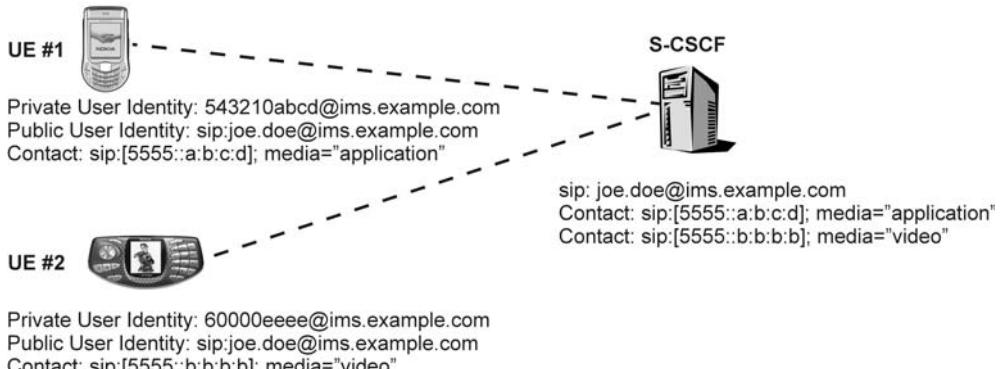
- Private user identity of the user – it is used in a registration request to identify the user's subscription (see Section 3.5.2 for further information).
- One or more public user identities of the user – it is used in a registration request to identify an identity to be registered and is used to request communication with other users (see Section 3.5.1 for further information).
- The name of the entry point of the home network (home network domain name) – it is used in a registration request to route the request to the user's home network.
- Administrative data include various data – which could be used, say, by IMS subscribers for IMS operations or by manufacturers to execute proprietary auto-tests.
- Access rule reference – it is used to store information about which personal identification number needs to be verified in order to get access to the application.
- Address of P-CSCF – it can be used when the access technology does not support dynamic P-CSCF discovery capabilities.
- Security parameters related to Generic Bootstrapping Architecture. Security parameters enable authentication to IMS.

Originally it was assumed that IMS capable devices must be equipped with ISIM but this requirement has been relaxed and currently mobile operators are dominantly allowing access to the IMS with devices equipped with SIM or USIM cards. Section 3.5.4 describes how IMS device can generate IMS identifiers and home domain name out of data stored in USIM.

### 3.7 Sharing a Single User Identity between Multiple Devices

Traditionally, in the CS every single user has their own Mobile Station International ISDN (MSISDN) number that is used to reach the user. It is not possible for a single user to use multiple terminals with the same MSISDN number simultaneously. Having two mobile stations with identical MSISDN numbers would cause significant conflicts in the network. Nowadays, users may have more than one item of UE with totally different capabilities: big/small screen, camera/no camera, full keyboard and so forth. Different UE may serve different purposes (e.g., one for gaming, another for ordinary voice and video sessions). From the user's point of view, the user should be reachable via the same identity regardless of the number of UEs that they are using simultaneously. The IMS makes this feature possible.

Release 6 IMS allows users to register the same public user identity from a number of items of UE. In addition, a user is able to indicate their preferences regarding a single



**Figure 3.7** Sharing a single user identity between multiple devices

UE at the registration phase. Different registrations can be differentiated by means of the private user identity and the used IP address. Figure 3.7 shows an example in which a user has two items of UE: one for video sessions and another for chat and gaming applications. When someone is calling the user – say, Joe – it is his S-CSCF that makes the decision as to which UE is going to be contacted in the first place when the request does not contain a specific request to reach a particular device. This decision can be done based on the preferences given at the registration phase: for example, if the incoming session contains a video component, then the S-CSCF could select UE # 2, which is Joe's primary preference for video sessions. In addition to preference-based routing, the S-CSCF may perform forking. There are two types of forking:

- sequential forking;
- parallel forking.

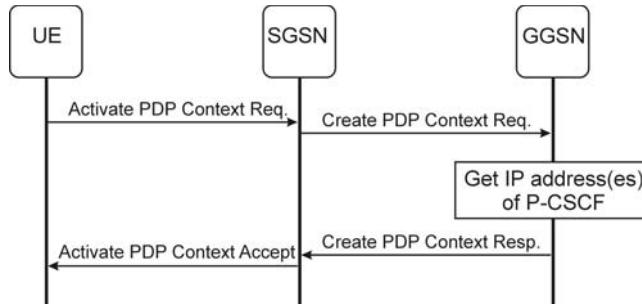
Sequential forking means that different items of UE are contacted one by one: for example, the S-CSCF first sends the request to UE # 2 and, if Joe fails to respond, within a certain time limit the S-CSCF then tries to reach Joe through UE # 1.

Parallel forking means that different items of UE are contacted at the same time: for example, when two items of UE are ringing, Joe can decide which UE to use for the incoming session; however, in the end the session can only be connected to a single item of UE.

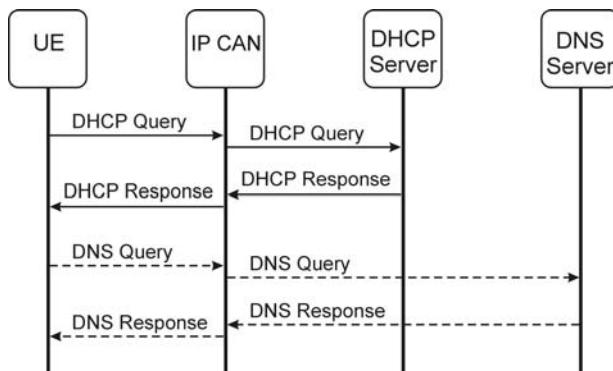
### 3.8 Discovering the IMS Entry Point

In order to communicate with the IMS, an item of UE has to know at least one IP address of the P-CSCF. The mechanism by which the UE retrieves these addresses is called “P-CSCF discovery”. Two dynamic mechanisms for P-CSCF discovery have been standardized in 3GPP: the Dynamic Host Configuration Protocol’s (DHCP) DNS procedure and the GPRS procedure. Additionally, it is possible to configure either the P-CSCF name or the IP address of the P-CSCF in the UE (see Section 3.6).

In the GPRS procedure (Figure 3.8), the UE includes the P-CSCF address request flag in the PDP context activation request (or secondary PDP context activation request) and



**Figure 3.8** A GPRS specific mechanism for discovering P-CSCF



**Figure 3.9** A generic mechanism for discovering P-CSCF

receives the IP address(es) of the P-CSCF in the response. This information is transported in the protocol configuration options information element [3GPP TS 24.008]. The mechanism the Gateway GPRS Support Node (GGSN) used to get the IP address(es) of the P-CSCF(s) is not standardized. This mechanism does not work with pre-Release 5 GGSNs.

In the DHCP DNS procedure (Figure 3.9), the UE sends a DHCP query to the IP connectivity access network (e.g., GPRS), which relays the request to a DHCP server. According to [RFC3319] and [RFC3315], the UE could request either a list of the SIP server domain names of the P-CSCF(s) or a list of the SIP server IPv6 addresses of the P-CSCF(s). When domain names are returned, the UE needs to perform a DNS query (NAPTR/SRV) to find an IP address of the P-CSCF (see Section 11.4 for details). The DHCP DNS mechanism is an access-independent way of discovering the P-CSCF.

### 3.9 S-CSCF Assignment

Section 3.8 explained how the UE discovers the IMS entry point – i.e., the P-CSCF. The next entity on a session signalling path is the S-CSCF. There are three cases when the S-CSCF is assigned:

1. When a user registers with the network.
2. When S-CSCF is needed to execute services on behalf of unregistered user.
3. When a previously assigned S-CSCF is not responding.

### *3.9.1 S-CSCF Assignment during Registration*

When a user is registering with a network the UE sends a REGISTER request to the discovered P-CSCF, which finds the user's home network entity – i.e., the I-CSCF – as described in Section 3.2. Then the I-CSCF exchanges messages with the HSS (UAR and UAA) as described in Section 2.3.5.1. As a result, the I-CSCF receives S-CSCF capabilities, as long as there is no previously assigned S-CSCF. Based on the received capabilities the I-CSCF selects a suitable S-CSCF. Capability information is transferred between the HSS and the I-CSCF within the Server-Capabilities Attribute Value Pair (AVP). The Server-Capabilities AVP contains [3GPP TS 29.228 and TS 29.229]:

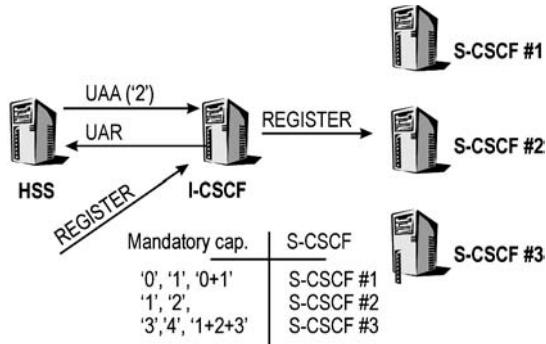
- Zero or more Mandatory-Capability AVPs – the type of this AVP is unsigned and contains a mandatory capability of the S-CSCF. Each mandatory capability available in an individual operator's network will be allocated a unique value.
- Zero or more Optional-Capability AVPs – the type of this AVP is unsigned and contains an optional capabilities of the S-CSCF. Each optional capability available in an individual operator's network will be allocated a unique value.
- Zero or more Server-Name AVPs – this AVP contains a SIP URI used to identify a SIP server.

Based on the mandatory and optional capability AVPs, an operator is able to distribute users between S-CSCFs, depending on the different capabilities (required capabilities for user services, operator preference on a per-user basis, etc.) that each S-CSCF may have. It is the operator's responsibility to define (possibly based on the functionality offered by each S-CSCF installed in the network) the exact meaning of the mandatory and optional capabilities. As a first choice, the I-CSCF will select the S-CSCF that has all the mandatory and optional capabilities for the user. If that is not possible, then the I-CSCF applies a “best-fit” algorithm. None of the selection algorithms is standardized (i.e., solutions are implementation-dependent). Figure 3.10 gives an example.

Using the Server-Name AVP, an operator has the possibility to steer users to certain S-CSCFs; for example, having a dedicated S-CSCF for the same company/group to implement a VPN service or just making S-CSCF assignment very simple.

### *3.9.2 S-CSCF Assignment to Execute Services for an Unregistered User*

Section 3.4 and Figure 3.3 explained at a high level how a session is routed from UE A to UE B. It can be seen from the figure that the I-CSCF is a contact point within an operator's network. In Section 2.3.5.1 location retrieval procedures were explained (i.e., an incoming SIP request will trigger LIR/LIA commands to find out which S-CSCF is serving User B). If the HSS knows that no S-CSCF is currently assigned and that the user has services related to the unregistered state then it returns S-CSCF capability information and the S-CSCF assignment procedure will take place in the I-CSCF as described in Section 3.9.1.



**Figure 3.10** Example of S-CSCF assignment

### 3.9.3 S-CSCF Assignment in Error Cases

3GPP standards allow S-CSCF re-assignment during registration when the assigned S-CSCF is not responding; that is, when the I-CSCF realizes that it cannot reach the assigned S-CSCF it sends the UAR command to the HSS and explicitly sets the type of authorization information element to the value registration and capabilities. After receiving S-CSCF capabilities, the I-CSCF performs S-CSCF assignment as described in Section 3.9.1.

### 3.9.4 S-CSCF De-Assignment

The S-CSCF is de-assigned when a user de-registers from the network or the network decides to de-register the user (e.g., because registration has timed out or the subscription has expired). It is the responsibility of the S-CSCF to clear the stored S-CSCF name from the HSS.

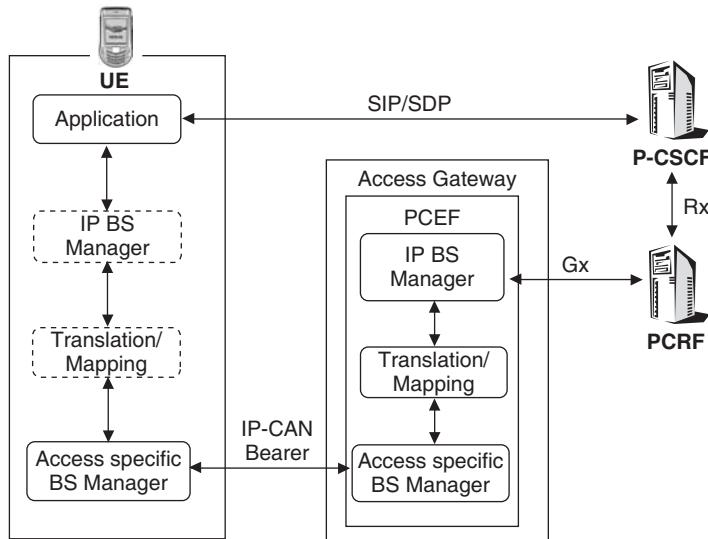
### 3.9.5 Maintaining S-CSCF Assignment

When a user de-registers from the network or a registration timer expires in the S-CSCF an operator may decide to keep the same S-CSCF assigned for the unregistered user. It is the responsibility of the S-CSCF to inform the HSS that the user has been deregistered; however, the S-CSCF could indicate that it is willing to maintain the user profile. This optimizes the load of the Cx reference point because there is no need to transfer the user profile once the user registers again or receives sessions while they have a service related to the unregistered state.

## 3.10 Mechanism for Controlling Bearer Traffic

### 3.10.1 Introduction

Separation of the control plane and the user plane was maybe one of the most important issues of IMS design. Full independence of the layers is not feasible because, without interaction between the user plane and the control plane, operators are not able to control



**Figure 3.11** Policy control entities

Quality of Service (QoS), source/destination of IMS media traffic, and when the media starts and stops. Therefore, a mechanism to authorize and control the usage of the bearer traffic intended for IMS media traffic was created; it is based on the Session Description Protocol (SDP) parameters negotiated at the IMS session. This overall interaction between the access and the IMS is called a policy control because the same architectural solution is used for ensuring coherent charging between access and IMS, therefore overall concept is typically called Policy and Charging Control (PCC). In this section we focus on the Policy Control part.

Figure 3.11 shows the functional entities involved in the PCC. In Release 5 “Service Based Local Policy (SBLP)”, policy decision function was integrated to P-CSCF and COPS protocol was used between P-CSCF and GGSN to control bearer traffic and to provide charging correlation (see Section 3.11.7) (name of the reference point was Go). Release 6 introduced standalone policy decision function that was connected using Diameter to the P-CSCF (name of the reference point was Gq) and COPS protocol was still used towards GGSN. In a separate architecture, flow based charging (FBC) was introduced in Release 6 using the Rx and Gx reference points as described in Section 3.11.5. The SBLP and FBC architecture were merged in Release 7. The Figure 3.11 depicts Release 7 based harmonized policy and charging control architecture where P-CSCF and PCRF are connected via Diameter reference point titled as the Rx reference point and the PCRF and Access Gateway are connected via Diameter reference point titled as the Gx reference point. It is fair to note that in ETSI TISPAN IMS architecture different reference points and a different name for policy control entity is given. At the time of writing 3GPP is working on harmonizing 3GPP and TISPAN policy control architecture. Here we focus on 3GPP Release 7 based architecture. The main building blocks of policy control architecture are:

- IP Bearer Service (BS) manager – manages the IP BS using a standard IP mechanism. It resides in the access gateway e.g. in GGSN and optionally in the UE.
- Translation/Mapping function – provides the interworking between the mechanism and parameters used within the Access Specific BS and those used within the IP BS. It resides in the access gateway and optionally in the UE.
- Access Specific BS manager – handles resource reservation requests from the UE. It resides in the access gateway and in the UE.
- Policy and Charging Enforcement Point (PCEF) – is a logical entity that enforces policy decisions made by the PCRF. It resides in the access gateway.
- Policy and Charging Rules Function (PCRF) – encompasses policy control decision and flow based charging control functionalities. It provides network control regarding the service data flow detection, gating, QoS and flow based charging towards the PCEF.

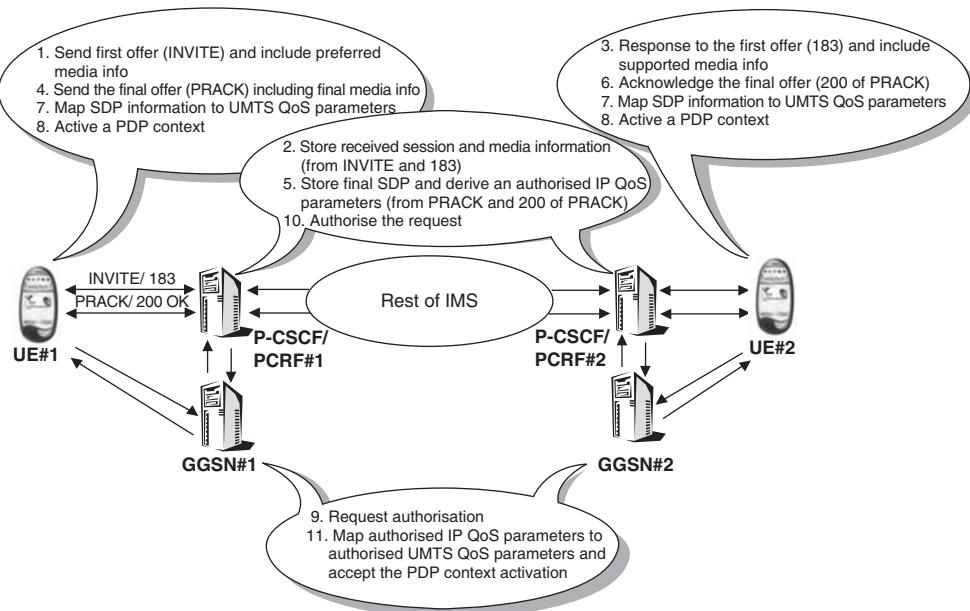
The example in this section does not explain how the P-CSCF maps SIP/SDP information to Diameter information elements nor how the information is transferred to Policy and Charging Rules Function (PCRF) using Diameter commands over the Rx and Gx reference points. On the contrary, the example focuses on the essentials: what information is needed from SIP/SDP session signalling and how it is used at the PCRF. The utilization of the Rx is covered in detail in Section 3.10.5. The following main functions can be identified.

- Gating Control: blocking or allowing of packets, belonging to a service data flow to pass through between desired endpoints;
- QoS control: the authorization and enforcement of the maximum QoS that is authorized for a service data flow or an IP-CAN bearer;
- Traffic plane event reporting: notifying events in user plane as well as the reporting of events related to the resources in the access gateway;
- IP-CAN bearer establishment for IP-CANs that support network initiated procedures for IP-CAN bearer establishment.

### 3.10.2 Gating and QoS Control

Session establishment and modification in the IMS involves an end-to-end message exchange using SIP and SDP. During the message exchange, UEs negotiate a set of media characteristics (e.g., common codec(s)). If an operator applies the policy control, then the P-CSCF will send the relevant SDP information to the PCRF. Based on this information PCRF forms IP QoS authorization data.

When the UE is activating or modifying an IP CAN bearer (e.g. PDP context) for media it has to perform its own mapping, from SDP parameters and application demands to some IP CAN QoS parameters (e.g. UMTS QoS parameters). On receiving the IP-CAN bearer activation or modification, the access gateway asks for authorization information from the PCRF. The PCRF compares the received information with the stored information and returns an authorization decision. This decision contains IP QoS parameters for the IP CAN bearer and packet classifiers to build proper gating and information on how to bound service flows to IP-CAN bearer(s).



**Figure 3.12** Bearer authorization in UE initiated model

In case of affirmative decision the access gateway maps the authorized IP QoS parameters to authorized access specific QoS parameters and, the IP-CAN bearer activation or modification will be accepted. Figure 3.12 shows this functionality. The PCRF is shown as a part of the P-CSCF for simplicity. When a standalone PCRF exists, then the P-CSCF needs to map SIP/SDP signalling information to appropriate Diameter information elements and send an appropriate Diameter request to the PCRF via the Rx reference point. This functionality is further described in Section 3.10.5.

During session setup the P-CSCF extracts information from SDP (steps 2 and 5) and delivers this information to the PCRF that forms IP QoS authorization data also known as policy control rules. The PCRF derives policy control data and charging related data. These data comprise service flow specific information encoded in so called PCC rules as well as IP CAN bearer specific QoS authorization information (QoS class and bandwidth). This data that will be sent from the PCRF to the PCEF. The PCC rules comprise:

- Rule name – used to reference a rule in the communication between the PCEF and the PCRF
- Service identifier – used to identify the service or the service component the service data flow relates to. It could, for example, contain information that the service is IMS multimedia telephony.
- Data rate – upper limit for bandwidth. It includes all the overheads coming from the IP layer and the layers above (e.g., UDP, RTP or RTCP). If multiple codecs per media are agreed to be used in a session, then the authorized data rate is set according to the codec requiring the highest bandwidth.

- QoS class – this information represents the highest class that can be used for the service data flow.
- Service data flow filters – used to select the traffic for which the rule applies. It further consists of source and destination IP addresses, port numbers and protocol.
- Gate status – indicates whether the service data flow, detected by the service data flow filter(s), may pass (gate is open) or shall be discarded (gate is closed) in uplink and/or in downlink direction.

Let's say that Tobias (UE # 1 in Figure 3.12) wants to talk to his sister Theresa (UE # 2). In addition to an ordinary voice call, Tobias wants to activate bidirectional and unidirectional video streams. Therefore, his terminal builds a SIP INVITE containing an SDP that reflects Tobias's preferences and his UE capabilities. SDP contains supported codecs, bandwidth requirements (plus characteristics of each) and assigned local port numbers for each possible media flow. Here we concentrate only on those parameters that are necessary for the policy control. Chapter 12 contains a general description for the whole session setup. SDP sent from UE # 1 would look like this:

```
v=0
o=- 3262464865 3262464868 IN IP6 5555::1:2:3:
t=3262377600 3262809600
m=video 50230 RTP/AVP 31
c=IN IP6 5555::1:2:3:4
b=AS:35
b=RS:700
b=RR:700
m=video 50240 RTP/AVP 31
c=IN IP6 5555::1:2:3:4
b=AS:32
b=RS:640
b=RR:640
a=sendonly
m=audio 3456 RTP/AVP 97 96
c=IN IP6 5555::1:2:3:4
b=AS:25.4
b=RS:500
b=RR:500
```

When PCRF in Figure 3.12 receives information based on this, it is able to formulate the authorization data for the downlink direction (from Access Gateway#1 to UE#1). When Theresa's UE responds, PCRF#1 is able to formulate the authorization data for the uplink direction (from UE#1 to Access Gateway#1). Note that Theresa is not willing to receive unidirectional video, therefore the corresponding port number is set to zero:

```
v=0
o=- 3262464865 3262464868 IN IP6 5555::1:2:3:4
t=3262377600 3262809600
```

```

m=video 60230 RTP/AVP 31
c=IN IP6 5555::5:6:7:8
b=AS:35
b=RS:700
b=RR:700
m=video 0 RTP/AVP 31
c=IN IP6 5555::5:6:7:8
b=AS:32
b=RS:640
b=RR:640
a=recvonly
m=audio 3550 RTP/AVP 0
c=IN IP6 5555::5:6:7:8
b=AS:25.4
b=RS:500
b=RR:500

```

From this information, PCRF#1 and PCRF#2 are able to construct service data flow filters. Table 3.1 shows the service data flow filters in PCRF#1.

The PCRF derives the data rate value for the media IP flow(s) from the ‘b = AS’ SDP parameter (in kilobytes). For associated Real-time Transport Control Protocol (RTCP) IP flows, the PCRF will use SDP ‘b = AS’, ‘b = RR’ and ‘b = RS’ parameters (in bytes), if present. When SDP ‘b = RR’ or ‘b = RS’ are missing the data rate for RTCP IP flows is derived from the available parameters. It is possible to assign different values for guaranteed bit rate and maximum bit rate. Table 3.3 shows maximum data rates per flow identifier as calculated in PCRF#1.

The PCRF maps media-type information to the highest QoS class that can be used for the media. The PCRF will use an equal QoS class for both uplink and downlink directions when both directions are used. Derivation of QoS class can be based on operator specific algorithm, based on service identifier, based on codec or media information. Nine different values have been specified [3GPP TS 29.213]. Table 3.2 shows how these map to UMTS QoS parameters used in 3GPP packet access.

Table 3.3 shows how the information in Table 3.2 is utilized in our example. The maximum authorized QoS class for an RTCP IP flow is the same as for the corresponding

**Table 3.1** Service data flow filters in the PCRF#1

Type of IP flows	Destination IP address/Port number of the IP flows	Rule name
RTP (Video) DL	5555::1:2:3:4 / 50230	<1,1>
RTP (Video) UL	5555::5:6:7:8 / 60230	<1,1>
RTCP DL	5555::1:2:3:4 / 50231	<1,2>
RTCP UL	5555::5:6:7:8 / 60231	<1,2>
RTP (audio) DL	5555::1:2:3:4 / 3456	<3,1>
RTP (audio) UL	5555::6:7:8:9 / 3550	<3,1>
RTCP DL	5555::1:2:3:4 / 3457	<3,2>
RTCP UL	5555::6:7:8:9 / 3551	<3,2>

**Table 3.2** IP QoS class mapping to UMTS QoS

IP QoS Class Value	UMTS QoS parameters			
	Traffic Class	Traffic Handling Priority	Signalling Indication	Source Statistics Descriptor
1	Conversational	n/a	n/a	speech (NOTE)
2	Conversational	n/a	n/a	unknown
3	Streaming	n/a	n/a	speech (NOTE)
4	Streaming	n/a	n/a	unknown
5	Interactive	1	Yes	n/a
6	Interactive	1	No	n/a
7	Interactive	2	No	n/a
8	Interactive	3	No	n/a
9	Background	n/a	n/a	n/a

NOTE: The IP QoS class values that map to “speech” should be selected for service data flows consisting of speech (and the associated RTCP) only.

**Table 3.3** The maximum data rates and QoS class per service data flow filters in the PCRF#1

	Video (RTP)	Video (RTCP)	Audio (RTP)	Audio (RTCP)
Maximum Data Rate Downlink (kbps)	35	0.7	25.4	0.5
Maximum Data Rate Uplink (kbps)	35	0.7	25.4	0.5
Maximum QoS Class	2	2	1	1

RTP media IP flow. The data rate and QoS class were created and stored in the PCRFs during Steps 2 and 5 in Figure 3.12. At the same time the PCRFs create service data flow filters, gate status and assigned name for this rule. Table 3.3 summarizes things so far (without gate status and name of the rule).

When the UE has collected enough information from SIP session signalling (see Sections 12.5 and 12.6) it can start reserving resources. The UE needs to decide what kind of QoS it desires. For GPRS access QoS parameters are listed in Table 3.4. From the policy control point of view the first three rows in Table 3.4 present values that are relevant. Interested readers can find detailed descriptions of other QoS parameters in [3GPP TS 23.107]. Here the traffic class, guaranteed bit rate and maximum bit rate are described:

- Traffic class – the four different traffic classes defined for UMTS are conversational, streaming, interactive and background. By including the traffic class, UMTS can make assumptions about the traffic source and optimize the transport for that traffic type.

**Table 3.4** Requested QoS parameters

Traffic class	Maximum bit rate for downlink
Guaranteed bit rate for downlink	Maximum bit rate for uplink
Guaranteed bit rate for uplink	
SDU format information	Maximum SDU size
SDU error ratio	Residual BER
Delivery of erroneous SDUs	Traffic Handling Priority
Transfer delay	Allocation/Retention priority
Source statistics descriptor	Delivery order

- Guaranteed Bit Rate (GBR) – describes the bit rate the UMTS bearer service will guarantee to the user or application.
- Maximum Bit Rate (MBR) – describes the upper limit a user or application can accept or provide. This allows different rates to be used for operation (e.g., between GBR and MBR).

The traffic class values – GBR for downlink/uplink and MBR for downlink/uplink – should not exceed the derived values of maximum authorized bandwidth and maximum authorized traffic class per flow identifier. The maximum authorized bandwidth in the UE is derived from SDP in the same way as was done in the PCRF. The maximum authorized traffic class is derived according to Table 3.5. The exact derivation rules for both parameters are described in [3GPP TS 29.213].

[3GPP TS 26.236] gives recommendations on how other requested QoS parameters for conversational codec applications could be set. Correspondingly, [3GPP TS 26.234] gives recommendations on how other requested QoS parameters for streaming codec applications could be set.

Table 3.6 shows the maximum authorized UMTS QoS parameters per service data flow as calculated by the UE.

Next, the UE needs to decide how many IP-CAN bearers (PDP contexts in GPRS access) are needed. The key factors are the nature of media streams (i.e., required traffic class) and the received grouping indication from the P-CSCF.<sup>2</sup> In our example there are two different types of bidirectional media streams: video and audio. Both media would require high QoS (low delay and preserved time relation); therefore, a single conversational traffic class PDP context would be suitable. Here we assume that the P-CSCF has not given any media grouping instructions therefore UE # 1 should activate only one PDP context or use an existing PDP if already available. Finally, Table 3.7 presents the maximum authorized UMTS QoS parameters as calculated by UE #1.

The UE has now completed Step 7 in Figure 3.12. After deriving and choosing the suitable QoS parameters the UE activates the necessary PDP context(s).

<sup>2</sup> Pre Release 6 GGSNs were able to produce only one GGSN Call Detail Record for a PDP context. Therefore, it is impossible to separate traffic for each media component with the same PDP context. That's why P-CSCF has a capability to force the UE to open separate PDP contexts for each media component. For this purpose a keep-it-separate indication was defined [3GPP TS 24.229] [RFC3524].

**Table 3.5** The maximum authorized traffic class per media type in the UE

Media type (m-line in SDP)	UMTS traffic class
Bidirectional audio or video	Conversational
Unidirectional audio or video	Streaming
Application	Conversationaly
Data	Interactive
Control	Interactive
Others	Background

**Table 3.6** The values of the maximum authorized UMTS QoS parameters per service data flow as calculated by UE #1 (Tobias) from the example

	Video (RTP)	Video (RTCP)	Audio (RTP)	Audio (RTCP)
Maximum data rate DL (kbps)	35	0.7	25.4	0.5
Maximum data rate UL (kbps)	35	0.7	25.4	0.5
Maximum QoS class	Conversational	Conversational	Conversational	Conversational

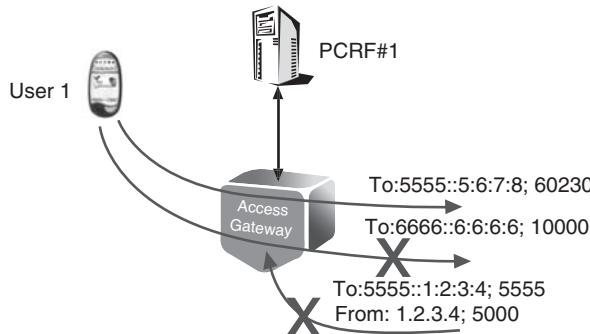
**Table 3.7** The values of the maximum authorized UMTS QoS parameters as calculated by UE #1 from the example

Maximum authorized bandwidth DL (kbps)	61.6
Maximum authorized bandwidth UL (kbps)	61.6
Maximum authorized traffic class	Conversational

When an access gateway (e.g. GGSN) receives IP-CAN bearer activation (e.g. a secondary PDP context activation request to an Access Point Name (APN)) for which the Gx reference point is enabled, the PCEF within the access gateway identifies the correct PCRF holding authorization data for a requesting user. Selection will be done based on requesting UE's IP address. Once the decision point is found it sends user identification information (e.g. IP address of the UE), requested QoS information (e.g. desired bandwidth, wanted QoS class), bearer information (source and destination IP address, source and destination port numbers, protocol, packet filter information) and other additional information to the resolved PCRF. This corresponds to Step 9 in Figure 3.12.

At Step 10 in Figure 3.12 the PCRF receives a Diameter request from the PCEF and the PCRF validates that the corresponding SIP session exists and authorization data containing rules matches with the received request. If this is not the case the PCRF rejects the request completely. Otherwise it communicates the decision (PCC rules) to the PCEF.

After getting the authorization data the PCEF maps the authorized IP QoS to the access specific QoS (e.g. UMTS QoS) and the PCEF accepts IP CAN bearer activation or modification.



**Figure 3.13** Example of IMS based gating in the Access Gateway

Based on received service data flow filters PCEF constructs a gate description. The gate description allows a gate function to be performed. The gate function enables or disables the forwarding of IP packets. When the gate is open, then the packets of the related IP flows are allowed to be forwarded. The opening of the gate may be part of the authorization decision or may be a standalone decision. If a standalone decision is used, then an operator can ensure that user-plane resources are not used before the IMS session is finally accepted (i.e., when a SIP 200 OK message is received). In this case end users may lose e.g. all announcements which are to be delivered before completing the session, as the PCEF will drop all incoming user-plane IP packets.

PCEF is also able to instruct the PCEF to close any open gate whenever desired or instructed by the P-CSCF based on information in SIP signalling messages. Once the PCEF gets this type of request it will drop incoming and outgoing packets subject for this gate.<sup>3</sup> This function is used, for example, when a session is put on hold. Typically gate and associated policy control rule(s) is removed when the UE releases service data flows that relates to a particular gate. However, in some cases UE (e.g. misbehaving or malfunctioning UE) might not remove a bearer resource(s) in synch with the state of the SIP session therefore P-CSCF can instruct PCRF to revoke bearer resources. For example, UE that has ended a SIP session but has not released associated user plane resources within a pre-defined time set by an operator then P-CSCF can initiate bearer resource release.

The example in this section has created two gates (for incoming and outgoing service data flows) at the Access Gateway: IMS user plane traffic between addresses 5555::1:2:3:4 and 5555::5:6:7:8 and associated ports are allowed when the gates are open. If it is assumed that the device uses the same IP address for sending and receiving traffic then it is possible to block all traffic outside these gates. For example if the UE#1 attempts to send traffic towards IP address 6666::6:6:6:6 the Access Gateway will discard the packet and it is not delivered further. Similarly if somebody is sending IP packets to the UE1 with valid port and IP address but source address is wrong the Access Gateway drops the packet. This is depicted in Figure 3.13.

<sup>3</sup> RTP traffic associated to RTP stream will not be blocked in order to ensure proper behaviour in the end points (RTCP can be used to detect whether the link is alive or not).

### 3.10.3 Traffic Plane Event Reporting

The previous section covered how bearers get authorized and policed. This section explains how the IMS can get up to date information regarding different events in the user plane. Using the policy control capabilities the P-CSCF is able to track status of IMS signalling and user plane bearers (new bearer created, bearer is lost, lost bearer recovered), which IP-CAN(s) the device is currently using (e.g. UTRAN/GERAN, WLAN, GAN, HSPA evolution), to get notification when some service data flows (e.g. video stream) or all service data flows (i.e. all media streams of particular SIP session) are deactivated. By utilizing this information the IMS could provide improved service execution. For example, the P-CSCF could initiate a SIP session release on behalf of the served user when it gets information that all service flows related to the ongoing SIP session are released, but the P-CSCF does not get any appropriate SIP message (i.e. SIP BYE) from the UE itself. The second example could be that if the P-CSCF has learned that there are no user plane bearers towards the UE and the UE has not deregistered from the IMS then the P-CSCF can immediately reject an incoming terminating SIP request with an appropriate error code instead of sending the request towards the UE and waiting until a timer would expire in the P-CSCF. The third example could be that the P-CSCF uses PCRF provided IP-CAN type to verify that the UE has inserted correct access type in P-Access-Network-Info SIP header. This could be useful when an operator applies access specific charging or wishes to provide access specific service handling in the IMS. To enable this type of functionality the P-CSCF sets proper Diameter AVP in Diameter AA-Request command (see Section 3.5.4.1) when it creates a Diameter session between itself and PCRF. Likewise the PCRF sets proper Diameter AVP in Diameter command (e.g. Re-Auth Request (RAA)) when the PCRF desired to get bearer event reporting from PCEF.

Figure 3.14 depicts a case when the P-CSCF wants to know status of IMS signalling bearer. In Steps 1 and 2 the UE activates signalling bearer and the bearer gets authorized by the PCRF. In Step 3 the UE performs an IMS registration and once the registration is accepted the P-CSCF contacts the PCRF to install subscription for IMS signalling bearer status (Steps 4, 5 and 6). In Steps 7 and 8 the PCRF makes similar subscription to the PCEF (if not done already as part of Step 2). After this procedure the PCEF reports changes to the PCRF which in turn reports changes to the P-CSCF.

The Gx reference point between PCRF and Access Gateway supports even more granular information than the Rx reference point between P-CSCF and PCRF. 3GPP Release 7 supports reporting change of SGSN, QoS, radio access type, GPRS traffic flow template, PLMN, IP-CAN type, Routing Area identity, user's location. Moreover it is possible to get notifications when a bearer is lost, when the lost bearer is recovered and a number of other event reports regarding failures in access gateway side [3GPP TS 29.212]. Based on this information the operator is able to build additional logic around policy control rules. For example, the PCRF could enforce a different QoS class depending on used PLMN or could allow wideband multimedia codecs only when user is close to home or office.

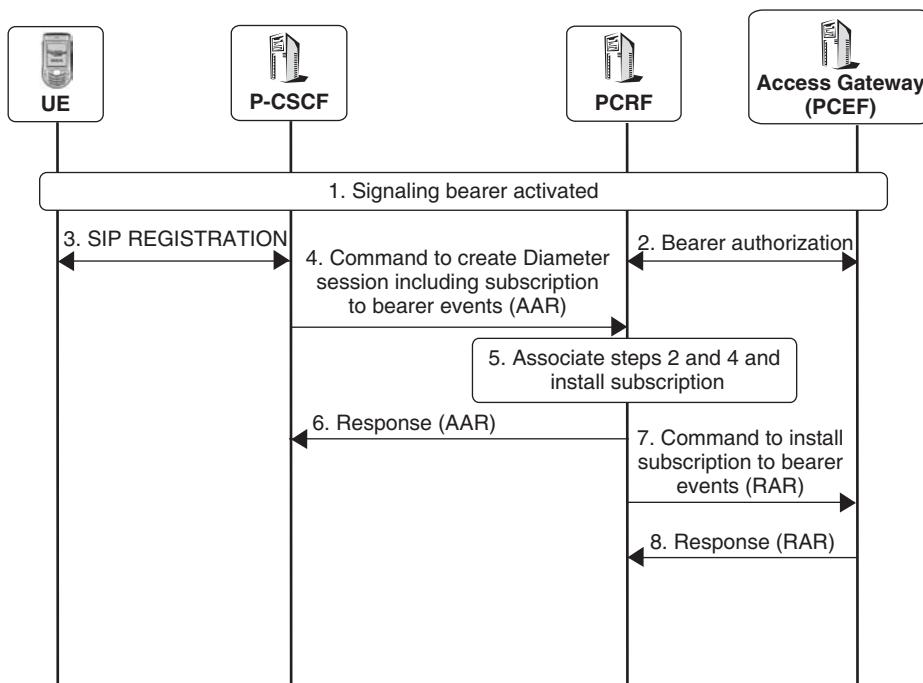
### 3.10.4 Network Initiated Bearer Activation

Widely deployed packet bearer for mobile telecommunication systems was introduced in Release 97 when GPRS was launched. In the GPRS system, mobile station (MS) has

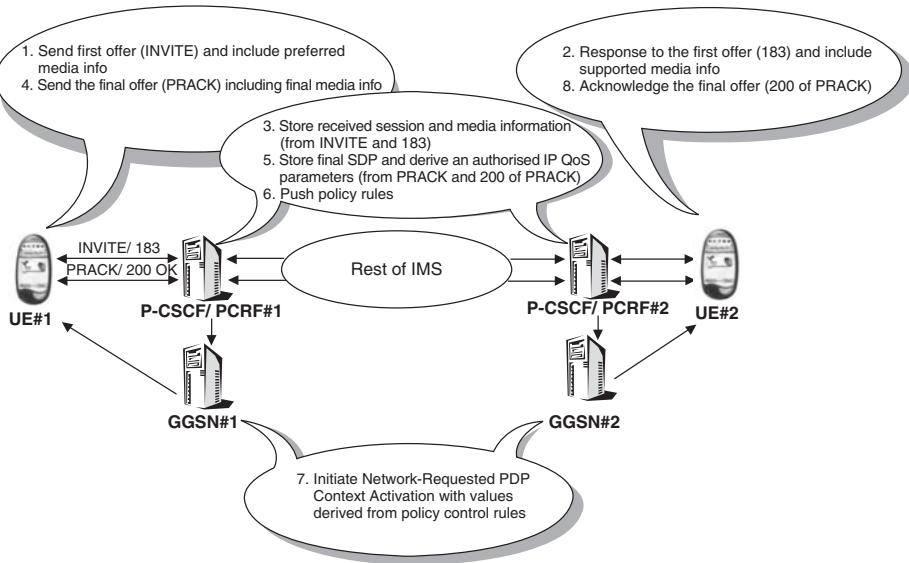
controlled bearer activation (establishment) since day one. In other words, the MS has decided how many PDP contexts are needed and what kind of traffic class and QoS parameters it wants to use with each PDP context. This model is likely to change in the future. At the time of writing 3GPP is defining a next generation GPRS system in Release 8 so called Evolved Packet System. Currently it is expected that it is primarily the network which decides what kind of bearers UE needs when communicating using the Evolved Packet System architecture. An initial step towards network initiated bearer activation was already taken in Release 7 as an optional capability to use network initiated bearer activation when IMS user plane traffic in GPRS access network was specified. It is expected that the UE initiated bearer activation is the dominant design prior to Release 8 deployments.

Using the policy control architecture GGSN (PCEF) is able to inform PCRF if the UE, SGSN and GGSN all supports network initiated bearer activation. Based on this information the PCRF is able to detect whether it waits for authorization requests from PCEF as shown in Step 9 in Figure 3.12 or whether it should push policy control rules immediately after learning sufficient SIP session details from P-CSCF.

Let's revisit examples we have presented so far in the context of policy control architecture. Figure 3.14 depicts how P-CSCF is able to monitor the status of IMS signalling bearer, but it also reveals that the IMS signalling bearer gets authorized by the PCRF. When the UE and GPRS network both support network initiated bearer activation this can be advertised in Step 2 in this figure. Figure 3.12 shows how bearer authorization works for UE initiated bearers. Now, when the network initiated bearer activation is used



**Figure 3.14** Subscription to IMS signaling bearer status



**Figure 3.15** Bearer authorization in network initiated model

the PCRF#1 could push policy control rules once it gets 183 Session Progress response from UE#2. Finally the PCEF(GGSN) gets necessary information about the needed bearer and it can activate suitable bearers for the UE. Advantage of this approach is: QoS handling and service flow bundling is centralized to the network. Figure 3.15 shows bearer authorization in network initiated mode.

### 3.10.5 Usage of Rx Reference Point

When policy and charging control is used in the network the P-CSCF sends information obtained from SIP/SDP session setup signalling to the PCRF via the Rx reference point. This information enables the PCRF to form authorized IP QoS data and appropriate policy control rules that will be delivered to the access gateway as described in Section 3.10.2. Similarly, the PCRF utilizes the Rx to send notifications of bearer events to the P-CSCF (see Section 3.10.3). For passing the information, the P-CSCF and PCRF use Diameter protocol as defined in 3GPP TS 29.214. Four Diameter request and answer pairs are used in the Rx reference point (see Table 3.8 as well):

- AA-Request/AA-Answer (AAR&AAA);
- Session-Termination-Request/Session-Termination-Answer (STR&STA);
- Re-Auth-Request/Re-Auth-Answer (RAR&RAA);
- Abort-Session-Request/Abort-Session-Answer (ASR&ASA).

#### 3.10.5.1 Interactions from P-CSCF to PCRF

To convey session information and IMS charging correlation identifier from P-CSCF to PCRF AAR command is used and it contains among other basic Diameter command

**Table 3.8** Rx commands

Command-Name	Purpose	Abbreviation	Source	Destination
AA-Request	P-CSCF uses AAR to push SIP session information and IMS charging correlation identifier towards PCRF.	AAR	P-CSCF	PCRF
AA-Answer	AAA delivers acknowledgement to AAR, access charging correlation identifier to P-CSCF and address of access charging entity.	AAA	PCRF	P-CSCF
Re-Auth-Request	RAR delivers bearer event reports (e.g. bearer loss/recovery, some component(s) of session released).	RAR	PCRF	P-CSCF
Re-Auth-Answer	RAA is used to acknowledge the RAR command.	RAA	P-CSCF	PCRF
Session-Termination-STR Request	STR is used to ensure that bearer resources are released together with a SIP session release.	STR	P-CSCF	PCRF
Session-Termination-Answer	STA is used to acknowledge the STR command.	STA	PCRF	P-CSCF
Abort-Session-Request	ASR is used to inform P-CSCF that all bearers related to a particular SIP session have been released.	ASR	PCRF	P-CSCF
Abort-Session-Answer	ASA is used to acknowledge the ASR command	ASR	P-CSCF	PCRF

elements information about the UE's IP address, media stream information, reporting policy, IMS charging identifier, service-URN, priority and information about SIP forking. Media stream information further express details about media stream(s) e.g. direction of traffic, source/destination IP address and port number, maximum requested bandwidth, status of each media component (enabled/disabled per uplink/downlink direction), type of media (audio, video, data, application, control, text, message, other).

Following our example, the P-CSCF # 1 in Figure 3.12 will first issue an AAR command when it receives a SIP INVITE request from Tobias's UE. This command will carry the necessary information to construct downlink information. Media stream information conveys direction of traffic, source IP address and port number, maximum requested bandwidth, status of each media stream (assumed to be disabled for both uplink and downlink direction for the time being), type of media stream (audio and video in this case). Reporting policy could contain a request for the PCRF#1 to report if the IP-CAN bearer to transport these media streams is dropped or if service data flow (e.g. video)

is deactivated. The P-CSCF # 1 then issues an AAR command when it receives a 183 Session Progress response from Teresa's UE. The second AAR command will carry the necessary information to construct uplink information (destination IP address and port number, maximum requested bandwidth, status of each media stream (assumed to be disabled for both uplink and downlink direction), type of media stream (audio and video in this case)). Based on this information the PCRF is capable of forming authorized IP QoS and communicate appropriate gate description to the access gateway.<sup>4</sup> It is an operator's decision whether to mark media stream status as disabled or enabled at this stage. If marked enabled then the PCRF opens the gates. If the media stream(s) are marked disabled the P-CSCF should activate media stream(s) when it received final SIP 200 OK response (i.e. SIP session is accepted). The AAR command is acknowledged with an AAA command that contains information to enable charging correlation (see Section 3.11.5 for more details) and information regarding used IP-CAN type and possible information about the used radio access type (UTRAN/GERAN, WLAN, GAN, HSPA evolution).

When the IMS session is released the P-CSCF uses Session-Termination-Request to inform the PCRF that session that was previously authorized is now ended and the PCRF shall enforce that associated service data flows will be released in the access gateway to prevent bearer misuse after SIP session termination. This command is acknowledged with an STA command.

### 3.10.5.2 Interactions From PCRF to P-CSCF

Re-Auth-Request (RAR) command is used to deliver notifications on bearer event which P-CSCF has previously asked. The following events can be reported: changes of IMS signalling and user plane bearer status (new bearer created, bearer is lost, lost bearer recovered), IP-CAN type changes (UTRAN/GERAN, WLAN, GAN, HSPA evolution) and some service data flows (e.g. video stream) are deactivated. This command is acknowledged with an RAA command.

After receiving a service data flow release notification from access gateway via the Gx reference point, the PCRF needs to investigate if all service data flows (media streams) related to a particular SIP session are now releases and if yes the PCRF will report this to the P-CSCF using ASR command including a reason that causes the session release. This command is acknowledged with an ASA command.

## 3.11 Charging

### 3.11.1 Introduction

Flat-free and volume-based charging schemas are traditional charging models in current IP-based communication networks. The IMS enables new charging models which, in turn, enables different business models for IMS operators. The capability to charge based on session or event or service is one of the key advantages that the IMS enables for operators. There are foreseen benefits for end users as well. For instance, an operator is able to

---

<sup>4</sup> In our example single speech codec and video codec is assumed therefore the final service information can be derived at this point of time. If multiple codec options were available final service information could be derived from SIP PRACK request and 200 OK of PRACK response.

offer peer-to-peer games as a pre-paid service (i.e., a user needs to have money in their account before consuming services) and other multimedia sessions as a post-paid service (i.e., a user pays for services periodically such as once per month), or instant messages could be available as a flat-free service and session-based messaging charged differently (e.g., based on duration of the session or based on transferred bytes).

In order to offer a post-paid service, the IMS needs to support a mechanism for offline charging. Offline charging is a charging process where charging information is mainly collected after the session and the charging system does not affect in real time the service being used. In this model a user typically receives a bill on a monthly basis, which shows the chargeable items during a particular period. A pre-paid service requires online charging support. This means that IMS network entities need to consult the Online Charging System (OCS) before allowing users to use services. The OCS is responsible for interacting in real time with the user's account and for controlling or monitoring the charges related to service usage.

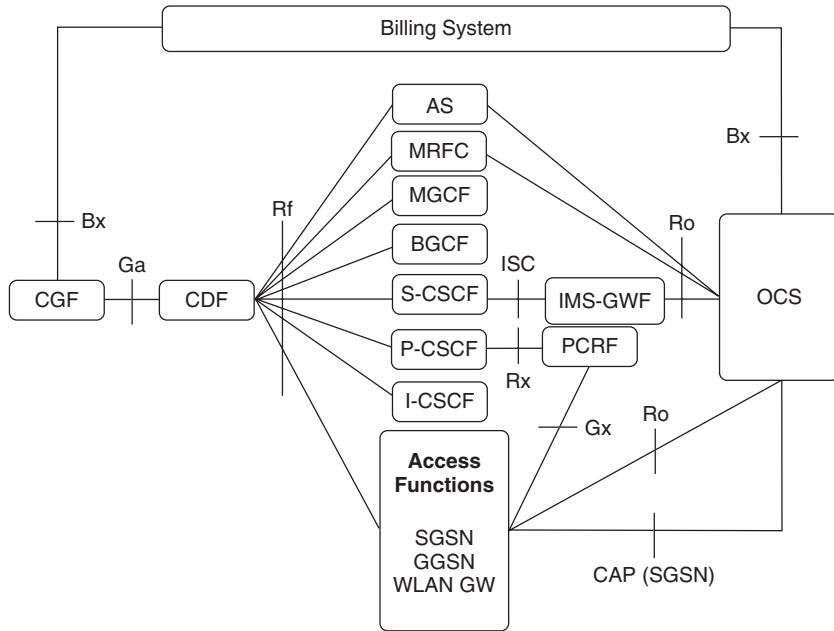
IMS network entities are configured to detect when a chargeable trigger condition is met. After detection, the entity collects the necessary information from a SIP request and either requests permission from a charging system (online charging) to continue processing the SIP request or sends relevant information to a charging system for creating a CDR for post-processing (offline charging), and allows the SIP request to continue. A chargeable trigger could be session initiation, session modification, session termination request (session-based charging) or it could be on any SIP transaction – e.g., MESSAGE, PUBLISH, SUBSCRIBE requests (event-based charging). Moreover, a trigger could be the presence of some SIP header or SDP information. Based on the received information the charging system either takes credit from the user's account (online charging) or transfers CDR(s) to the billing system. This sounds rather easy, but standardizing a charging solution has been very painful.

### *3.11.2 Charging Architecture*

Due to the different nature of charging models different architecture solutions for offline and online are defined. Figure 3.16 shows the high-level IMS charging architecture. The left side of the figure depicts offline charging and the right side shows online charging.

From the figure you can see that all IMS entities handling SIP signalling are able to communicate with the offline charging entity – i.e., Charging Data Function (CDF) – using a single Diameter-based Rf reference point [3GPP TS 32.299]. The CDF receives a Diameter request also from access network entities and based on the information provided from various entities it creates CDRs which are delivered to the Charging Gateway Function (CGF) via the Ga reference point [3GPP TS 32.295]. Finally, the CGF processes the received CDRs and transfers the final CDR(s) to the billing system using the Bx reference point [3GPP TS 32.240].

In contrast, only three IMS entities (AS, MRFC and S-CSCF) are involved in online charging. Moreover, the S-CSCF cannot communicate directly with the OCS due to bad design in the Release 5 timeframe. An IMS-Gateway Function (IMS-GWF) is used to perform the necessary protocol conversion. The OCS supports two reference points from other network entities. SGSN uses the CAMEL Application Part (CAP) and the rest of the entities use the Diameter-based Ro reference point. Like CGF in offline charging the OCS



**Figure 3.16** IMS charging architecture

is also able to create CDRs in addition to credit control handling (approving resources in real time).

### 3.11.3 Offline Charging

IMS signalling traverses through various IMS entities and, as stated earlier, all entities are able to generate offline charging information. In fact, each offline charging-capable entity contains an integrated function called a Charging Trigger Function (CTF). The CTF is aware of charging triggers (such as the beginning of IMS sessions, IMS session modification, IMS session termination, sending of message, subscribing to an event, publishing presence information) and is able to decide when it needs to contact the CDF, the central point in the offline charging system. When a trigger condition is met the CTF collects charging information from the signalling message and sends the offline charging information to the CDF using Diameter Accounting Requests (ACRs) via the Rf interface. The request contains much information about the event that launched the trigger (e.g., type of request INVITE/MESSAGE/SUBSCRIBE, calling party address, called party address, time stamps). The CDF uses a Diameter Accounting Answer (ACA) to acknowledge the received request. In the case of an IMS session, at least two ACR/ACA pairs are sent (at the start of the session and at the end of the session). Further ACR(s) may be used as well if the session properties are changed (e.g., media components added or removed, the codecs of the media component and bandwidth have changed, session is placed on hold). In the case of a single end user to network transaction (e.g., sending an instant message) a single ACR/ACA is sufficient. The usage of Diameter requests is further described in Section 3.11.6.1(Rf reference point).

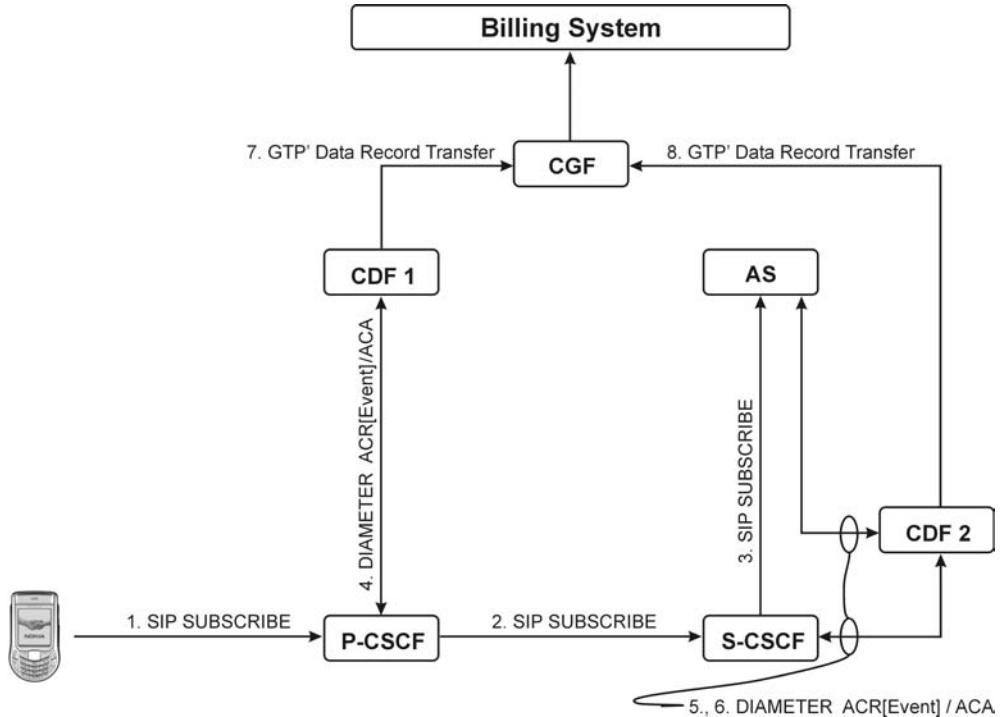
So far we have described how charging information is delivered from an IMS entity to the CDF. From Figure 3.16 we can see that there are further steps ahead before a billing system can send an actual bill to a user. The next step is to transfer CDRs from the CDF on towards the CGF. The CGF is needed as there can be multiple CDFs involved in a single session/nonsession event because different IMS entities may send charging information to different CDFs (e.g., due to roaming or configuration reasons). The CGF validates, consolidates, pre-processes the incoming CDR (e.g., filters unnecessary fields and adds operator-specific information), and may correlate different CDRs before passing them to the billing system. Table 3.9 summarizes the key procedures supported by different offline charging functions.

In Figure 3.17 an offline charging example is given. In this example a user sends a subscription request to an AS. It is assumed that the S-CSCF and the AS are using the same CDF (CDF#2 in the figure) and the P-CSCF is using a different CDF (CDF#1 in the figure). The subscription request could, for example, be a request to discover who are the members of a particular Push to talk Over Cellular group. In Steps 1–3 a SUBSCRIBE request traverses from the UE to the AS. In Steps 4–6 CTFs inside the IMS entities detect a chargeable event, create an ACR and send it to the CDF. In Steps 7–8 CDFs send appropriate CDRs to the CGF which, in turn, delivers the CDR(s) to the billing system. CDRs from the CDF to the CGF are transferred using a Data Record Transfer request in GPRS Tunnelling Protocol which includes functions for charging (GTP') [3GPP TS 32.295]. Please note that it is possible for CDF#2 to create a single CDR from the charging information received. Moreover, it is possible for a CGF to consolidate the received CDRs and send a single CDR to the billing system.

Although in this example it was assumed that multiple CDFs are in use, it is possible that a single CDF address is distributed to all IMS entities involved in a SIP session or transaction. This enables sending charging information to a single CDF. How this can be done is explained in Section 12.7.

**Table 3.9** Summary of offline charging functions

Offline charging function	Key procedures
Charging Triggering Function (CTF)	Monitors SIP signalling Detects trigger condition Extracts information from SIP signalling and assembles charging information Sends charging information to CDF
Charging Data Function (CDF)	Constructs CDRs Delivers CDRs to CGF
Charging Gateway Function (CGF)	Correlates, consolidates, filters unnecessary fields and adds operator-specific information to the received account information CDR error handling and storage Delivers CDRs to billing system Pre-processes CDRs.
Billing system	Creates the actual bill



**Figure 3.17** Example of offline charging

### 3.11.4 Online Charging

The purpose of online charging is to perform credit control before usage of IMS services/resources. Two different models exist: direct debiting and unit reservation. In direct debiting, an IMS network entity contacts the OCS and asks permission to grant the usage of services/resources. The OCS uses an internal rating function to find the appropriate tariff for an event based on the received information, if the cost was not given in the request. After resolving the tariff and the price, the OCS checks whether the user has enough credits in their account. If so, the OCS deducts a suitable amount of money from the user's account and grants the request from the IMS entity. In the unit reservation model, the OCS receives a credit control request from an IMS entity and uses an internal rating function to determine the price of the desired service according to the service-specific information provided by the IMS entity, if the cost was not given in the request. Then the OCS reserves a suitable amount of money from the user's account and returns the corresponding number of resources to the requesting IMS entity. Among the number of resources could be, for example, time or allowed data volume. When resources granted to the user have been consumed or the service has been successfully delivered or terminated, the IMS entity informs the OCS of the number of resources consumed. Finally, the OCS deducts the used amount from the user's account [3GPP TS 32.240, TS 32.296]. It is also possible for the OCS to receive subsequent requests from the IMS entity during service

execution if all granted resources are consumed. In this case the OCS needs to perform a new credit authorization.

The direct debiting model is appropriate when the IMS entity knows that it could deliver the requested service to the user itself. For example, a game AS could send a credit control request and inform the OCS of the service (say, a game of rally) and the number of items (say, 2) to be delivered. Then the OCS uses the rating function to resolve the tariff (€0.3) and to calculate the price based on the number of delivered units €0.6). Finally, 0.6 is deducted from the user's account and the OCS informs the game AS that 2 units have been granted within the credit control answer. 3GPP's definition for this online charging model is Immediate Event Charging [3GPP TS 32.240].

The unit reservation is suitable when the IMS entity is unable to determine beforehand whether the service could be delivered or when the required number of resources are not known prior to the use of a specific service (e.g., duration of multimedia session). The unit reservation model is usually applied to sessions (3GPP's term for this is Session Charging with Unit Reservation) but it is also possible to apply nonsession related requests (3GPP's term for this is Event charging with unit reservation).

In this book we do not probe the OCS further but readers who are interested to learn more about the internal functions of the OCS are recommended to read 3GPP's specification: Online Charging System (OCS) applications and interfaces [3GPP TS 32.296].

### *3.11.5 Flow-Based Charging*

The flow-based charging model introduces the ability to charge for service data flows identified by service flow filters according to certain charging rules.<sup>5</sup> Flow based charging is part of the PCC functionality as described in Section 3.10. Charging rules contain information that allows the filtering of traffic to identify the packets belonging to a particular service data flow (e.g., IMS, FTP, browsing), and allows definition of how the service data flow is to be charged (e.g. different media streams within single PDP context). Charging rules are normally requested by a Policy and Charging Enforcement Point (PCEF) at the bearer establishment, upon a specified trigger event and upon bearer termination. This request is made using the Gx reference point towards a Policy and Charging Rule Function (PCRF). The PCRF is further connected to an application function via the Rx reference point (in the IMS case it is usually the P-CSCF or AS). The Rx reference point enables transport of information (e.g., dynamic media stream information) from the application function to the PCRF. An example of such information would be filter information to identify an IMS session and its connection parameters (e.g. end points, media description).

### *3.11.6 Charging Reference Points*

For charging purposes four IMS-related reference points, Ro (online charging), Rf (offline charging), Rx exist and Gx exist. All reference points are based on the Diameter protocol developed by Internet Engineering Task Force (IETF).

---

<sup>5</sup> In Release 5 this capability did not exist and granularity of bearer charging was on PDP context level. Release 6 introduced flow-based charging but the architecture was slightly different.

### 3.11.6.1 Rf Reference Point (Offline Charging)

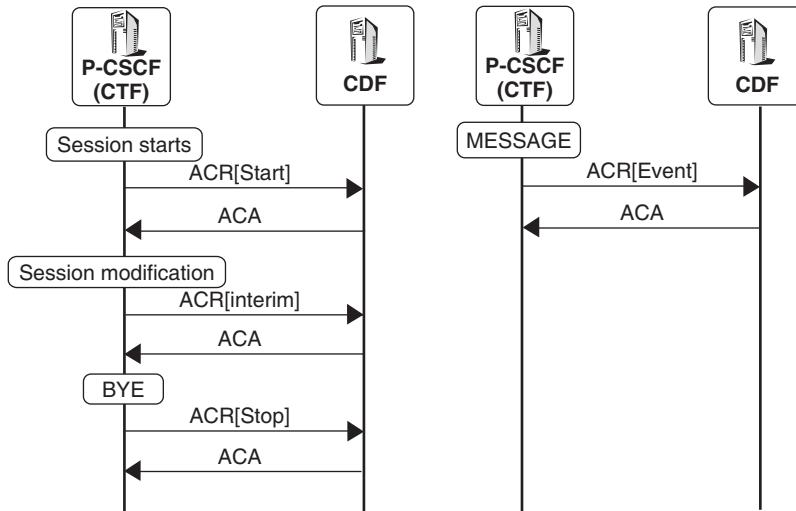
At the beginning of this chapter we explained that the CTF inside the IMS entity is responsible for detecting when it is necessary to report a chargeable event to the CDF (charging system in general). This critical task is achieved by sending a Diameter ACR via the Rf reference point to the CDF. The CDF replies back using another Diameter command, an Accounting Answer (ACA).

The basic Diameter functionality is described in [RFC3588] and forms the base of the Rf reference point. In addition to the Diameter base protocol, 3GPP has defined its own extensions as a 3GPP Diameter accounting AVP to meet the charging requirements of 3GPP. A 3GPP-specific AVP contains information which is considered valuable especially in the 3GPP environment but not necessarily in broad Internet. For instance, it includes media description of the session (audio, video, message, chat), authorized Quality of Service, involved ASs. The ACR used in 3GPP contains suitable Diameter protocol AVPs and 3GPP Diameter accounting AVPs. The use of AVPs is specified per IMS entity and ACR type: for example, ACRs generated by the S-CSCF could contain information about the contacted AS.

The offline charging system needs to support both session- and event-based charging. IMS entities know the request type that need to be indicated to the CDF. This is achieved using a suitable value in the Accounting-Record-Type AVP ('event' for events and 'start', 'interim', 'stop' for sessions) in the ACR. IMS session-related ACRs are called start, interim and stop and are sent at the start, during and at the end of a session, as the name implies. Nonsession related ACRs are called event ACRs. Event ACRs cause the CDF to generate corresponding CDRs, while session ACRs cause the CDF to open, update and close corresponding CDRs. Let's take two examples on usage of the Rf reference point, IMS session and instant message (nonsession related), to explain the usage of Diameter ACR.

In an IMS session three different phases can be detected (session initiation, session modification and session release). At beginning of a session (200 OK, acknowledging an INVITE is received), the CTF inside an IMS entity monitors the signalling traffic and detects a triggering point defined for reception of 200 OK, acknowledging an INVITE. When the triggering point is met the CDF collects information from the signalling messages (e.g., calling party address, called party address, time stamps, 'audio' SDP media component), assembles the charging information that matches the detected chargeable event and forwards the charging information towards the CDF via the Rf reference point using an ACR[Start] request (see Figure 3.18). Usage of an ACR[Start] prompts the CDF to open a CDR for this session. When the same session is modified (RE-INVITE or UPDATE is received) – e.g., a video component is added – the CTF could again trigger this event and collect the necessary information again (e.g., calling party address, called party address, time stamps, 'audio + video' SDP media component). This changed charging information is sent again to the CDF, but this time an ACR[Interim] request is used (see Figure 3.18). Finally, when the session ends (BYE is received) the CTF constructs the ACR[Stop] request to indicate session termination (see Figure 3.18). Based on these three charging events, the CDF can create a single CDR including total session time, audio session time and video session time.

After receiving a nonsession related request (here MESSAGE) a trigger point can be met again. The CTF collects the necessary information from the request (e.g., calling party address, called party address, time stamps, content length) and this time it constructs an



**Figure 3.18** Session- and event-based offline charging example

ACR[Event] request to indicate event-based charging (see Figure 3.18). As a result of this ACR[Event] the CDF knows to apply event-based charging, generates a CDR immediately and passes it on to the CGF.

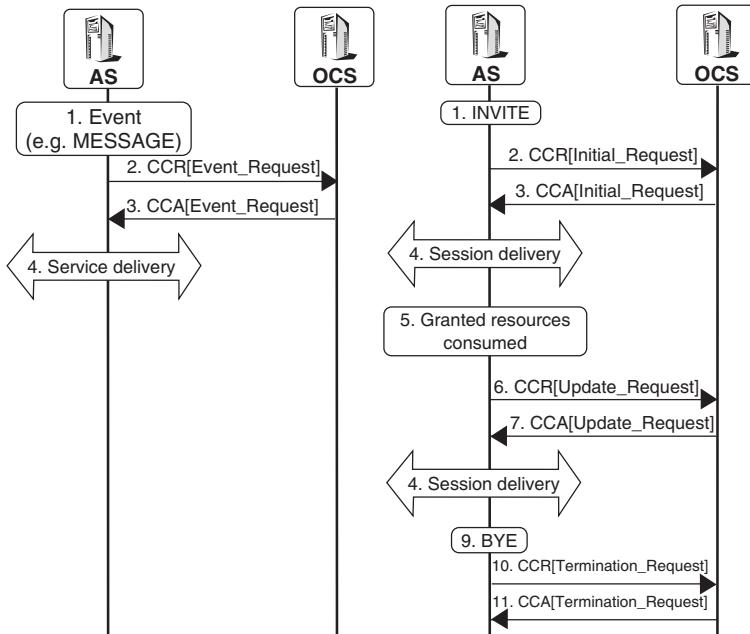
It is the operator's choice which SIP method, ISDN User Part (ISUP) or Bearer Independent Call Control (BICC) message triggers sending of the ACR. However, two mandatory items have been defined:

- Whenever SIP 200 OK, acknowledging an initial SIP INVITE, is received or the MGCF receives an ISUP/BICC answer, ACR[Start] will be sent to the CDF.
- Whenever SIP BYE is received or the Media Gateway Control Function (MGCF) receives an ISUP/BICC release, ACR[Stop] will be sent to the CDF.

### 3.11.6.2 Ro Reference Point (Online Charging)

In order to perform online charging the OCS needs to get the necessary information from the requesting IMS entity. For this purpose the Ro reference point is defined. It transfers credit control requests and answers between the OCS and three different IMS entities which are able to perform online charging (AS, MRFC and S-CSCF via the IMS-GWF). Credit Control Request and Credit Control Answer commands from RFC4006 are used for this purpose. In addition, 3GPP has defined 3GPP Credit Control AVPs to enhance IETF's solution [3GPP TS 32.299] to meet the charging requirements of 3GPP.

To enable direct debiting (see Section 3.11.4 for details) the IMS entity sends a Credit Control Request to the OCS and uses the value 'EVENT\_REQUEST' in CC-Request-Type AVP and the value 'DIRECT\_DEBITING' in Requested-Action AVP. For example, a messaging AS may receive a request from a user to send an instant message to somebody (Step 1 in Figure 3.19). The messaging server knows that the user is a pre-paid user and, therefore, it needs to seek permission from the OCS. It constructs a Credit Control



**Figure 3.19** Session- and event-based online charging example

Request, sets the CC-Request-Type AVP, Requested Action AVP and any other required AVP correctly, and then sends the request to the OCS (2). The OCS gets the request and, if it does not contain information about the price of service, the OCS uses a rating function before consulting the user's account. This example is depicted in Figure 3.19. When the user has enough credits in the account the OCS grants the request with a Credit Control Answer (3). Finally, the messaging server allows the service and sends the instant message towards the destination (4). This example is depicted in the left hand side of Figure 3.19.

To enable session charging with unit reservation (see Section 3.11.4 for definition) the IMS entity sends a Credit Control Request to the OCS and then uses values 'INITIAL\_REQUEST', 'UPDATE\_REQUEST' and 'TERMINATION\_REQUEST' in the CC-Request-Type AVP as follows:

- 'INITIAL\_REQUEST' value is used when the IMS entity receives the first service delivery request.
- 'UPDATE\_REQUEST' value is used when the IMS entity request reports the number of units used and indicates a request for additional units.
- 'TERMINATION\_REQUEST' value is used when the IMS entity reports that the content/service delivery is complete or the final allocated units have been consumed.

To enable event charging with unit reservation (see Section 3.11.4 for definition) the IMS entity sends a Credit Control Request to the OCS and then uses values 'INITIAL\_REQUEST' and 'TERMINATION\_REQUEST' in the CC-Request-Type AVP as follows:

- ‘INITIAL\_REQUEST’ value is used when the IMS entity receives the first service delivery request.
- ‘TERMINATION\_REQUEST’ value is used when the IMS entity reports that content/service delivery is complete.

For example, an operator who offers a pre-paid service for its customers needs to route all signalling traffic via an AS or IMS-GWF in order to apply online charging. In this example, the AS-based approach is assumed. In Step (1) in Figure 3.19 the AS receives a SIP session request (INVITE). The SIP INVITE triggers a Credit Control Request including the value ‘INITIAL\_REQUEST’ in the CC-Request-Type AVP as this is the first request in this session. The OCS receives the request (2) and uses the information provided to decide whether or not to grant this request. The Credit Control Answer (3) contains the number of service units granted and based on this the AS is able to allow the SIP session to continue (4). When the number of units granted are used up or there is a need for additional units (e.g., due to addition of a media component) the AS sends a new Credit Control Request but this time the value in the CC-Request-Type AVP is different (6). Again the OCS makes a credit control decision and communicates it, based on which the SIP session continues (7, 8). When the session is terminated or all units are used up the AS sends a third Credit Control Request indicating termination of the session and uses an appropriate value in the CC-Request-Type AVP. This example is depicted in the right-hand side of Figure 3.19.

### 3.11.6.3 Rx Reference Point for Charging

When flow-based charging is used in the network the IMS entities (in practice, the P-CSCF and AS) assist the PCRF by transferring media information of the IMS session to the PCRF over the Rx reference point. The PCRF uses the provided information to generate dynamic charging rules which are communicated to the access network (PCEF) using the Gx reference point, see Section 3.11.6.4. Moreover P-CSCF uses the Rx reference point to pass IMS charging identifier to PCRF and similarly the PCRF uses the Rx reference point to pass access charging identifier to P-CSCF and to report traffic plane events (e.g. bearer releases, bearer lost, bearer recovered) which can cause P-CSCF to perform additional operations towards IMS layer charging entities over Rf or Ro reference points. Section 3.11.7 explains how these charging identifiers are used to correlate access CDRs and IMS CDRs. Section 3.11.8 shows how P-CSCF distributes the obtained access charging identifier to other IMS entities.

### 3.11.6.4 Gx Reference Point for Charging

The Gx reference point resides between PCRF and Access Gateway as shown in Figure 3.16. This reference point enables the following charging related functions:

- exchange of IMS charging identifier and access charging identifier;
- distribution of primary and secondary addresses of offline and online charging entity addresses to the access gateway (i.e. OCS and CDF);
- activation of online and offline charging in access gateway (enabled/disabled);

- metering method to be applied in access gateway (duration, volume or both);
- rating group information (e.g. 0.1 per minute);
- desired reporting level in access gateway (based on given service or based on given service and rating-group).

### 3.11.7 Charging Information Correlation

Due to the layering design (see Section 2.1.11), IMS entities are not aware of the user-plane traffic volumes related to IMS sessions, and IP connectivity network entities (e.g., SGSN and GGSN) are not aware of the status of control-plane signalling (i.e., the status of IMS sessions). From the operator's perspective, it is desirable to have a possibility to correlate charging information created at the user plane and the control plane. Exchanging charging identifiers – the IMS Charging Identifier (ICID) and the access network charging identifier e.g. GPRS Charging Identifier (GCID) – through the Gx reference point enables charging correlation between the IMS and the access networks.

During the SIP session establishment phase the necessary IP-CAN bearer(s) e.g. PDP context(s) in GPRS is activated. During the IP-CAN bearer authorization process the access gateway and the PCRF exchange charging identifiers as follows:

1. The PCRF passes the ICID to the access gateway.
2. The access gateway passes the access network charging identifier e.g. GCID to the PCRF.

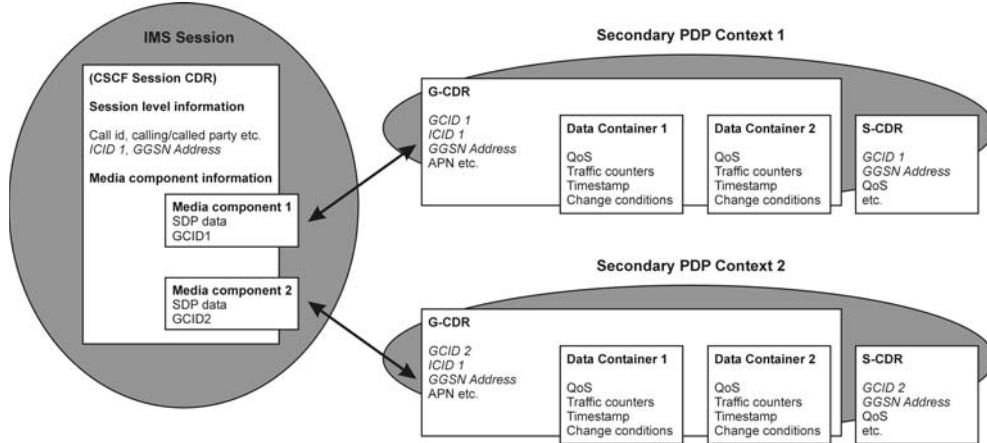
The PCRF also passes the access network charging identifier to the P-CSCF, which forwards it to the IMS entities in its own network where it is included on IMS CDRs. The access gateway (e.g. GGSN) includes the ICID on its CDRs (e.g. in GGSN Charging Data Record). In the case of GPRS the following applies: if single IMS session requires several secondary PDP contexts, one or more GCIDs are mapped to one ICID. In addition, the GGSN is responsible for updating GCID information at the IMS level when secondary PDP context or media flows are removed or added during the session. As a last link, the SGSN creates an S-CDR (i.e., an SGSN Charging Data Record) that includes GCID and GGSN addresses. This is a unique identifier for each PDP context. Figure 3.20 shows an example of an IMS session that contains two media components which are transported in separate PDP contexts.

As seen from the example above, the 3GPP IMS architecture defines the ICID and GCID for charging data correlation and a mechanism for exchanging the identifiers between the IMS and the PS domain.

### 3.11.8 Charging Information Distribution

Section 3.11.7 explained how charging information is correlated. The present section gives an overview of how charging information is distributed between different IMS entities. An example of SIP details is given in Section 12.7.

The first IMS entity within the SIP signalling path generates an ICID. This ICID is passed along the SIP signalling path to all entities involved, except the UE: that is, the P-CSCF in the terminating network will remove the ICID. The ICID is used for correlating charging data between IMS components. The ICID applies for the duration of the event



**Figure 3.20** IMS charging correlation

with which it is associated: for example, an ICID assigned for session establishment is valid until session termination, etc. We can see from Figure 3.21 that IMS and GPRS charging identifiers are exchanged when the bearer is authorized. In addition, Figure 3.21 indicates when accounting requests are sent to the CDF. The address of the CDF is distributed during registration or, alternatively, it is configured in IMS entities.

## 3.12 User Profile

### 3.12.1 Introduction

A user profile is a collection of user-specific information that is permanently stored in the HSS and downloaded to the S-CSCF when the S-CSCF needs to execute service for registered or un-registered user. The user profile contains at least one private user identity and single service profile. Figure 3.22 depicts the general structure of a user profile [3GPP TS 29.228]. The private user identity is described in Section 3.5.2, but it should be understood that a user profile may contain more than one private user identity, if e.g. a user is using a shared public user identity as described in Section 3.7. Figure 3.4 shows that a single IMS subscription may contain multiple service profiles; this allows different treatment for different public user identities as explained in Section 3.5.3.

Operator assigns a user profile when a user obtains an IMS subscription from an operator. The profile is transferred from the HSS to an assigned S-CSCF in two user data-handling operations – Server-Assignment-Answer (SAA) and Push-Profile-Request (PPR) – as described in Sections 2.3.5.1 and 2.3.5.2. The service profile is carried in one Diameter AVP, where it is included as an Extensible Markup Language (XML) document. The service profile is further divided into four parts:

- public identification;
- core network service authorization;

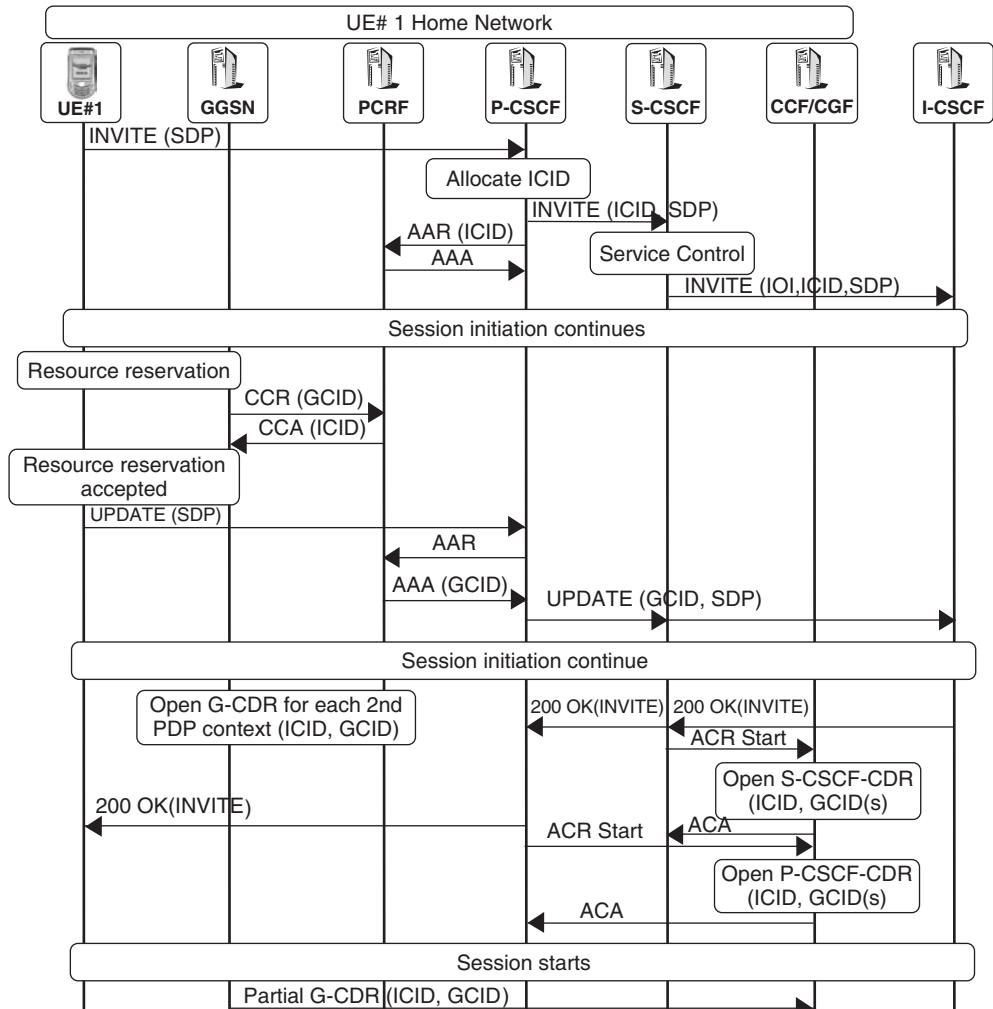


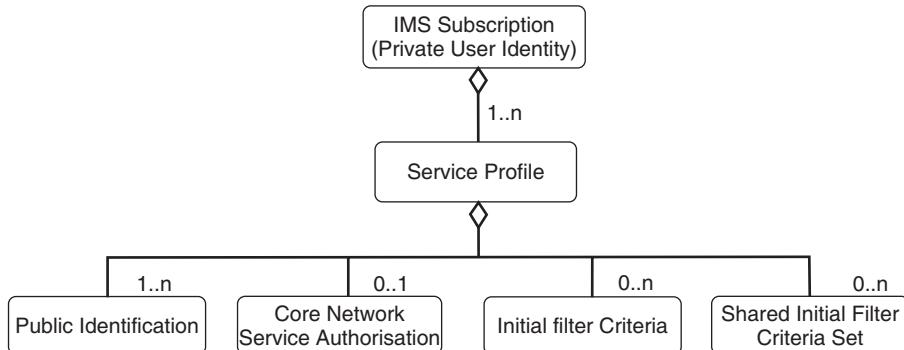
Figure 3.21 Distribution of charging information

- initial filter criteria;
- shared initial filter criteria set.

### 3.12.2 Public Identification

Public Identification identifies one or more identities for which a particular service profile will be executed. Identity can be either public user identity or public service identity.

Each public user identity contains an associated barring indication and optional display name (e.g. Nokia Corporation). If the barring indication is set, then the S-CSCF will prevent that public identity (e.g., a temporary public user identity) from being used in



**Figure 3.22** Structure of IMS user profile

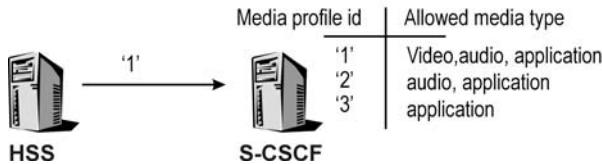
any IMS communication other than registrations and re-registrations. If the display name is included the S-CSCF will distribute it to UE and P-CSCF during registration and the IMS network will confirm its usage during session setup i.e. far end can trust also the content of display name. In addition each public user identity can be marked to belong to a particular alias identity group. Two or more identities are aliases if the identities belong to the same implicit registration set, are linked to the same service profile and have the same service data configured for each and every service. Practically alias identities are expected to be treated in similar manner when operator executes services e.g. applying IM/PoC service settings.

For public service identity Public Identification contains either exactly the public service identity (e.g. createconference@conferenceserver1.com) or wildcarded public service identity (sip:chatlist!.\*!@example.com) that matches to URIs which begin with sip:chatlist and end with @example.com (e.g. sip:chatlist1@example.com and sip:chatlist.userx@example.com). Display name can be bind to public service identities but alias grouping and barring indications do not have a meaning in the context of public service identities.

### 3.12.3 Core Network Service Authorization

Two different capabilities for service authorization have been defined: media policy and IMS communication service identifier policy.

Media policy information contains an integer that identifies a subscribed media profile in the S-CSCF (e.g., allowed SDP parameters). This information allows operators to define different subscriber profiles in their IMS networks. They may define different customer classes, such as gold, silver and bronze. Gold could mean that a user is able to make video calls and all ordinary calls. Silver could mean that a user is able to use wideband Adaptive Multi-Rate (AMR) as a speech codec, but they are not allowed to make video calls and so on. Transferring just the integer value between the HSS and the S-CSCF saves the storage space in the HSS and optimizes the usage of the Cx reference point.



**Figure 3.23** Media authorization in S-CSCF

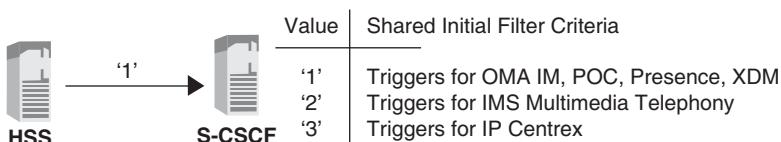
The S-CSCF needs to have a static database that contains the mapping between the integer value and the subscribed media profile. The meaning of the integer value is not standardized (i.e., it is operator specific). Figure 3.23 gives an illustrative example.

IMS communication service identifier policy contains a list of service identifiers that identifies which IMS communication service user is entitled to use. Based on the provided list the S-CSCF enforces usage of identifiers in SIP signalling. Details of IMS communication service identifier are covered in Section 11.9.

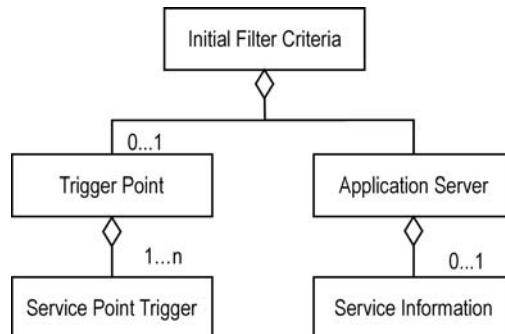
#### 3.12.4 Service-Triggering Information

Service-triggering information is presented in the form of initial filter criteria. Initial filter criteria describe when an incoming SIP message is further routed to a specific AS. User profile may contain both user specific service-triggering information which is coded as Initial Filter Criteria and a reference value to initial filter criteria which are locally administrated and stored in S-CSCF. The latter one is called Shared Initial Filter Criteria and it is encoded as an integer value where the integer value has only meaning inside single operator's network. For example, value 1 could point triggers that take care of routing requests to OMA IM, PoC, Presence and XDM applications and value 2 could point triggers that take care of routing requests to IMS multimedia telephony application and value 3 could point triggers that take care of routing requests to IP Centrex (see Figure 3.24).

Figure 3.25 shows that user specific Initial Filter Criteria are composed of either zero or one instance of a trigger point and one instance of an AS [3GPP TS 29.228]. Each initial filter criterion within the service profile has a unique priority value (integer) that is utilized in the S-CSCF. When multiple initial filter criteria are assigned the S-CSCF assesses them in numerical order: that is, an initial filter criterion with a higher priority number will be assessed after one with a smaller priority number.



**Figure 3.24** Shared initial filter criteria



**Figure 3.25** Structure of initial filter criteria

### 3.12.4.1 Trigger Point

The trigger point describes conditions that should be checked to discover whether the indicated AS should be contacted. The absence of a trigger point will indicate unconditional triggering to an AS. Each trigger point contains one to multiple instances of the Service Point Trigger. Service Point Triggers may be linked by means of logical expressions (AND, OR, NOT). Section 3.13 will give a more detailed explanation of how trigger points are used.

### 3.12.4.2 Application Server (AS)

The Application Server (AS) defines the AS that is contacted if the trigger points are met. The AS may contain information about the default handling of the session if contact with the AS fails. Default handling will either terminate the session or let the session continue based on the information in the initial filter criteria. In addition, the AS contains zero or one instance of the service information. Service information enables provisioning of information that is to be transferred transparently via the S-CSCF to an AS when the conditions of initial filter criteria are satisfied during registration.

## 3.13 Service Provision

### 3.13.1 Introduction

Strictly speaking the IMS is not a service in itself; on the contrary, it is a SIP-based architecture for enabling an advanced IP service and application on top of the PS network. IMS provides the necessary means for invoking services; this functionality is called ‘service provision’. IMS service provisioning contains three fundamental steps:

1. Define possible service or service sets.
2. Create user-specific service data in the format of initial filter criteria when a user orders/modifies a subscription.
3. Pass an incoming initial request to an AS.

Item (1) is not addressed in this book because it is up to operators and service providers to define what kind of services they are willing to offer their subscribers. The other two steps are described next.

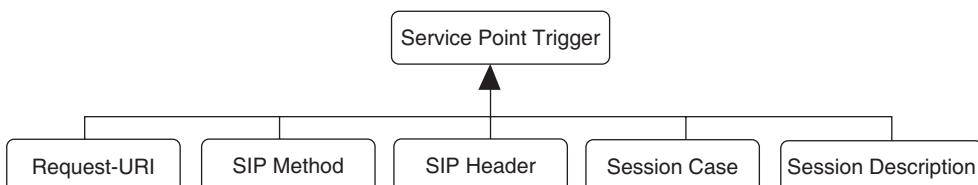
### 3.13.2 Creation of Filter Criteria

Whenever a user obtains an IMS subscription and their subscription contains some value-added services or an operator is willing to utilize ASs as part of its IMS infrastructure, they need to create service-specific data. These service-specific data are part of the user's user profile. More precisely, service-specific data are represented as initial filter criteria. Hereafter, we only concentrate on initial filter criteria. Section 3.12 describes how initial filter criteria fit into a user profile. When constructing initial filter criteria an operator needs to consider these questions:

- What is a trigger point?
- What is the correct AS when the trigger point is met?
- What is the priority of an initial filter criterion?
- What should be done if the AS is not responding?

The trigger point is used to decide whether an AS is contacted. It contains one to multiple instances of a Service Point Trigger [3GPP TS 29.228]. The Service Point Trigger comprises the items shown in Figure 3.26:

- Request-URI – identifies a resource that the request is addressed to (e.g., sportnews@ims.example.com).
- SIP Method – indicates the type of request (e.g., INVITE or MESSAGE).
- SIP Header – contains information related to the request. A Service Point Trigger could be based on the presence or absence of any SIP header or the content of any SIP header. The value of the content is a string that is interpreted as a regular expression. A regular expression could be as simple as a proper noun (e.g., John) in the FROM header that indicates the initiator of the request.
- Session Case – can be any one of four possible values, Originating, Terminating, Originating\_Unregistered or Terminating\_Unregistered, that indicate whether the filter should be used by the S-CSCF that is handling the originating service, terminating service or originating/terminating for an unregistered end user service. An originating case refers to when the S-CSCF is serving the calling user. A terminating case refers to when the S-CSCF is serving the called user.



**Figure 3.26** Structure of service point trigger

- Session Description – defines a Service Point Trigger for the content of any SDP field within the body of a SIP Method. Regular expressions can be used to match the trigger.

Based on the above an operator could build, for example, initial filter criteria to handle unregistered users, such as an IMS user who has not registered any of their public user identities. The following initial filter criterion routes an incoming session to a voice mail server (sip:vmail@ims.example.com) when the user is not registered. To make this happen the operator has to set a SIP Method to match INVITE and a session case to match the value of Terminating\_Unregistered (value 2). If the voice mail server cannot be contacted, then the default handling should be that the session is terminated (value 1). Initial filter criteria are coded in XML, as shown below (see [3GPP TS 29.228] for the exact coding rules of initial filter criteria):

```
<?xml version="1.0" encoding="UTF-8"?>
<testDatatype xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="D:\CxDataType.xsd">
<IMSSubscription>
  <PrivateID>privatexzyjoe@ims.example.com </PrivateID>
  <ServiceProfile>
    <PublicIdentity>
      <Identity>sip: joe.doe@ims.example.com </Identity>
    </PublicIdentity>
    <PublicIdentity>
      <Identity>tel:+358503334444</Identity>
    </PublicIdentity>
    <InitialFilterCriteria>
      <Priority>0</Priority>
      <TriggerPoint>
        <ConditionTypeCNF>0</ConditionTypeCNF>
        <SPT>
          <ConditionNegated>0</ConditionNegated>
          <Group>0</Group>
          <Method>INVITE</Method>
        </SPT>
        <SPT>
          <ConditionNegated>0</ConditionNegated>
          <Group>0</Group>
          <SessionCase>2</SessionCase>
        </SPT>
      </TriggerPoint>
      <ApplicationServer>
        <ServerName>sip:vmail@ims.example.com</ServerName>
        <DefaultHandling>1</DefaultHandling>
      </ApplicationServer>
    </InitialFilterCriteria>
  </ServiceProfile>
</IMSSubscription>
</testDatatype>
```

### 3.13.3 Selection of AS

Initial filter criteria are downloaded to the S-CSCF on user registration or on a terminating initial request for an unregistered user. After downloading the user profile from the HSS, the S-CSCF assesses the filter criteria for the initial request alone, according to the following steps [3GPP TS 24.229]:

1. Check whether the public user identity is barred; if not, then proceed.
2. Check whether this request is an originating request or a terminating request.
3. Select the initial filter criteria for a session case (originating, terminating, or originating/terminating for an unregistered end user).
4. Check whether this request matches the initial filter criterion that has the highest priority for that user by comparing the service profile with the public user identity that was used to place this request:
  - if this request matches the initial filter criterion, then the S-CSCF will forward this request to that AS, check to see whether it matches the next following filter criterion of lower priority and apply the filter criteria on the SIP Method received from the previously contacted AS;
  - if this request does not match the highest priority initial filter criterion, then check to see whether it matches the following filter criterion's priorities until one does match;
  - if no more (or none) of the initial filter criteria apply, then the S-CSCF will forward this request based on the route decision.

There exists one clear difference in how the S-CSCF handles originating and terminating initial filter criteria. When the S-CSCF realizes that an AS has changed the Request-URI in the case of terminating initial filter criteria, it stops checking and routes the request based on the changed value of the Request-URI. In an originating case the S-CSCF will continue to evaluate initial filter criteria until all of them have been evaluated.

If the contacted AS does not respond, then the S-CSCF follows the default-handling procedure associated with initial filter criteria: that is, either terminate the session or let the session continue based on the information in the filter criteria. If the initial filter criteria do not contain instructions to the S-CSCF regarding the failure to contact the AS, then the S-CSCF will let the call continue as the default behaviour [3GPP TS 24.229].

According to our initial filter criteria example, incoming INVITE requests will be routed to a voice mail server, vmail@ims.example.com, when Joe is not registered in the network. In exceptional cases, when the voice mail server is not responding, the S-CSCF is instructed to release a session attempt.

### 3.13.4 AS Behaviour

Section 3.13.3 described how the request is routed to an AS. After receiving the request the AS initiates the actual service. To carry the service out the AS may act in three different modes:

- Terminating User Agent (UA) – the AS acts as the UE. This mode could be used to provide a voice mail service.

- Redirect server – the AS informs the originator about the user's new location or about alternative services that might be able to satisfy the session. This mode could be used for redirecting the originator to a particular Web page.
- SIP proxy – the AS processes the request and then proxies the request back to the S-CSCF. While processing, the AS may add, remove or modify the header contents contained in the SIP request according to the proxy rules specified in [RFC3261].
- Third-party call control/back-to-back UA – the AS generates a new SIP request for a different SIP dialog, which it sends to the S-CSCF.

In addition to these modes, an AS can act as an originating UA. When the application is acting as an originating UA it is able to send requests to the users: for example, a conferencing server may send SIP INVITE requests to a pre-defined number of people at 9 am for setting up a conference call. Another example could be a news server sending a SIP MESSAGE to a soccer fan to let him know that his favourite team has scored a goal. See more examples of AS service execution in Chapters 4,6,7,8 and 9.

## 3.14 Connectivity between Traditional CS Users and IMS Users

### 3.14.1 Introduction

For the time being, most users utilize traditional CS UE: that is, fixed line telephones and all kinds of cellular terminals. Therefore, it is desirable for the IMS to interwork with legacy CS networks to support basic voice calls between IMS users and CS network users. This requires interworking both at the user plane and the control plane because the used protocols are different in both planes. Control-plane interworking is tasked to the MGCF. It performs mapping from SIP signalling to the BICC or ISUP used in CS legacy networks, and vice versa. The IMS Media Gateway (IMS-MGW), in turn, translates protocols at the user plane. It terminates the bearer channels from the CS (PSTN/ISDN/GSM) networks as well as media streams from IP or ATM-based PS networks and provides the translation between these terminations. Additional functions, such as codec interworking, echo cancellation and continuity check, can also be provided. The terminations are controlled by the MGCF. Network configurations for handling both IMS and CS-originated calls are explained next.

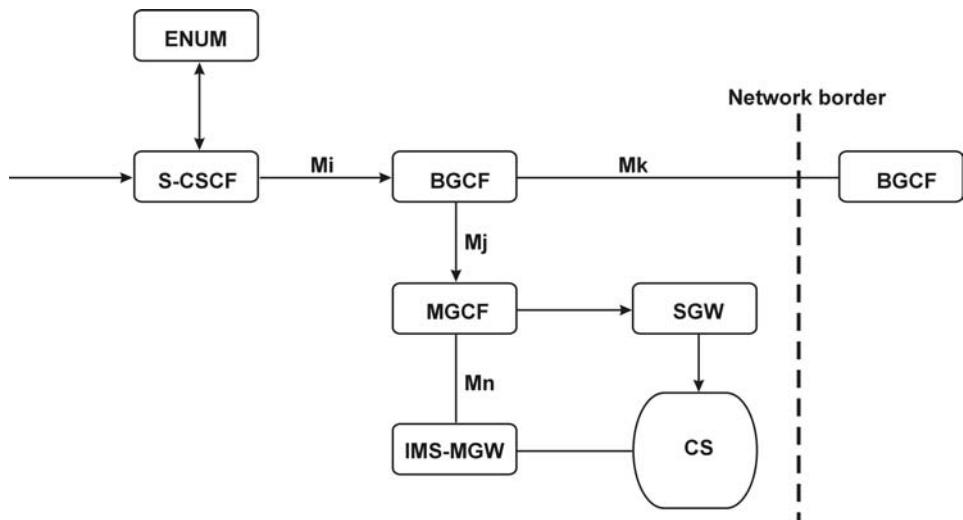
### 3.14.2 IMS-Originated Session Toward a User in the CS Core Network

When an IMS user initiates a session she does not need to bother about whether the called user is an IMS user or a CS user. She simply makes a call and the IMS takes care of finding the called party. The session request from the calling user will always arrive at the S-CSCF serving the calling user, based on a route learned during IMS registration. When the S-CSCF receives a session request using a tel URL type of user identity (tel:+358501234567), it has to perform an ENUM query to convert the tel URI to a SIP URI, as IMS routing principles do not allow routing with tel URIs. If the S-CSCF is able to convert the identity to SIP URI format it will route the session further to the target IMS network, and when this conversion fails the S-CSCF will try to reach the user in the CS network. To break out to the CS network, the S-CSCF routes the session

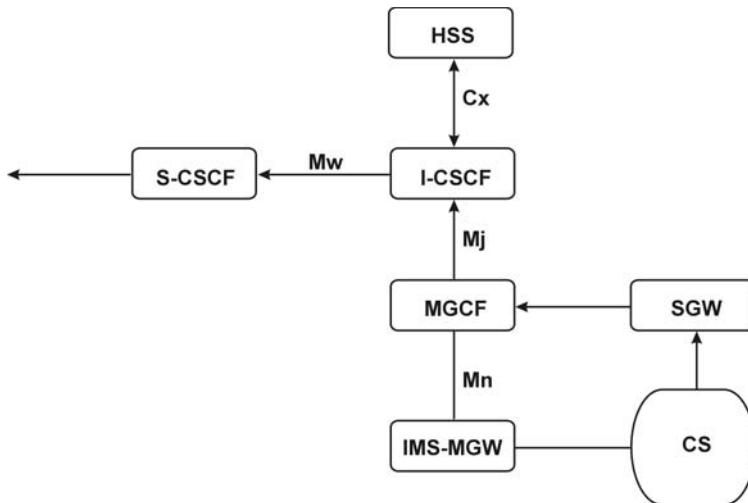
request further to the Breakout Gateway Control Function (BGCF) in the same network. The selected BGCF has two options: either selecting the breakout point in the same network or selecting another network to break out to the CS network. In the former case the BGCF selects an MGCF in the same network in order to convert SIP signalling to ISUP/BICC signalling and control the IMS-MGW. In the latter case the BGCF selects another BGCF in a different IMS network to select an MGCF in its network for handling breakout. The MGCF acts as an end point for SIP signalling; so, it negotiates media parameters together with the IMS UE and, similarly, negotiates media parameters together with the CS entity (e.g., with an MSC server). Figure 3.27 visualizes the interworking concept when an IMS-originated session is terminated in the CS network. The arrows in the figure show how the first signalling message traverses from the S-CSCF to the CS network.

### 3.14.3 CS-Originated Session Toward a User in IMS

When a CS user dials an E.164 number that belongs to an IMS user, it will be handled in the CS network like any other E.164 number; however, after routing analysis it will be sent to an MGCF in the IMS user's home network. After receiving the ISUP/BICC signalling message, the MGCF interacts with the IMS-MGW to create a user-plane connection, converts ISUP/BICC signalling to SIP signalling and sends a SIP INVITE to the I-CSCF, which finds the S-CSCF for the called user with the help of the HSS (as described in Section 2.3.5.1). Then the S-CSCF takes the necessary action to pass the SIP INVITE to the UE. Thereafter, the MGCF continues communication with the UE and the CS network to set the call up. Figure 3.28 shows how the functions interwork when a CS-originated call is terminated by the IMS network. The arrows in the figure show how the first signalling message traverses from the CS to the IMS user.



**Figure 3.27** IMS-CS interworking configuration when an IMS user calls a CS user

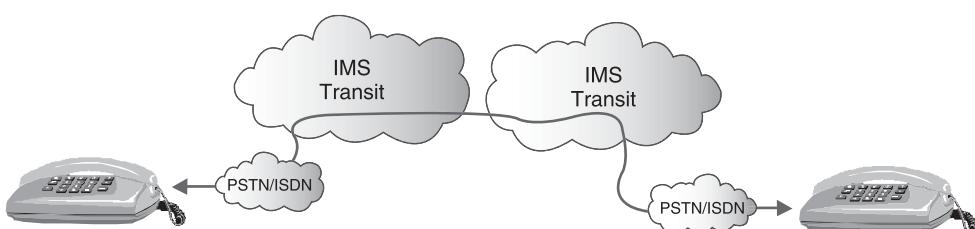


**Figure 3.28** IMS-CS interworking configuration when a CS user calls an IMS user

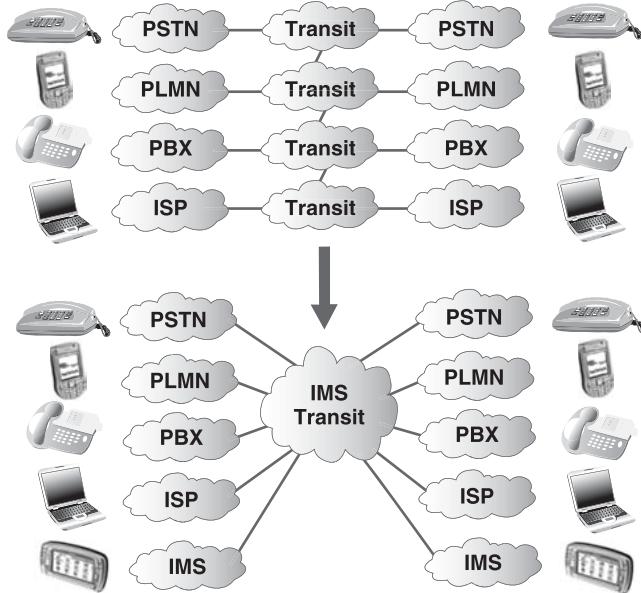
### 3.15 IMS Transit

In current telephony and data networks operators and service providers do not have direct connections with all the networks in the world. They rather use a third party provider to manage and operate interconnection networks, so called transit networks. These transit networks typically contain a number of transit nodes that interface with nodes in the originating and terminating side. Moreover, usually for each technology (e.g. PSTN, PLMN, Enterprise and Internet Service Providers (ISP)) different transit networks are used.

History of IMS transit feature dates back to 2005 when TISPAN was working on fixed IMS standard, NGN release 1.0. Intention was to use IMS routing capabilities to route traffic from one PSTN/ISDN network to other PSTN/ISDN network (as shown in Figure 3.29). As an example, an operator that manages both PSTN/ISDN (or directly back to the PSTN/ISDN if routing further via the IMS is not possible) and IMS routes all outgoing calls from PSTN/ISDN to MGCF of its network which converts incoming ISUP signalling



**Figure 3.29** IMS transit solution for PSTN/ISDN



**Figure 3.30** IMS as a general transit network

to SIP signalling<sup>6</sup> and routes the request further to the IMS transit function. The IMS transit analyzes the destination and routes traffic to the IMS terminating operator that is also managing PSTN/ISDN (or directly back to the PSTN/ISDN if routing further via the IMS is not possible) so the request gets routed to the I-CSCF which finds out the termination domain is PSTN/ISDN e.g. using IMS transit function or using HSS and routes the request via BGCF/MGCF to the PSTN/ISDN.

Later on 3GPP took over the IMS transit work from TISPAN and generalized the feature and created architecture for IMS transit function in Release 7. IMS transit function itself is rather simple; it gets a request and analyzes the destination address and decides where to route the request based on nonstandardized means. It could be based on DNS/ENUM lookup, private database lookup or configurable data. In IMS architecture the IMS transit functionality can be a standalone entity or it might be combined with the functionality of MGCF, I-CSCF, S-CSCF or IBCF. With this type of architecture IMS could support different transit scenarios e.g. transit functionality for its own non-IMS subscribers, transit functionality for other operators and service providers and transit functionality for enterprise networks (PBX in the Figure 3.30). In principle, IMS could provide general interconnection for all kind of networks with the IMS transit feature as shown in Figure 3.30.

<sup>6</sup>This conversion is a bit problematic as one of the transit network requirements is that information should not be lost and currently SIP does not contain headers to carry all possible information included in the request. Therefore, 3GPP and TISPAN have discussed using specific XML containers attached to the SIP requests or encapsulating ISUP message inside the SIP requests.

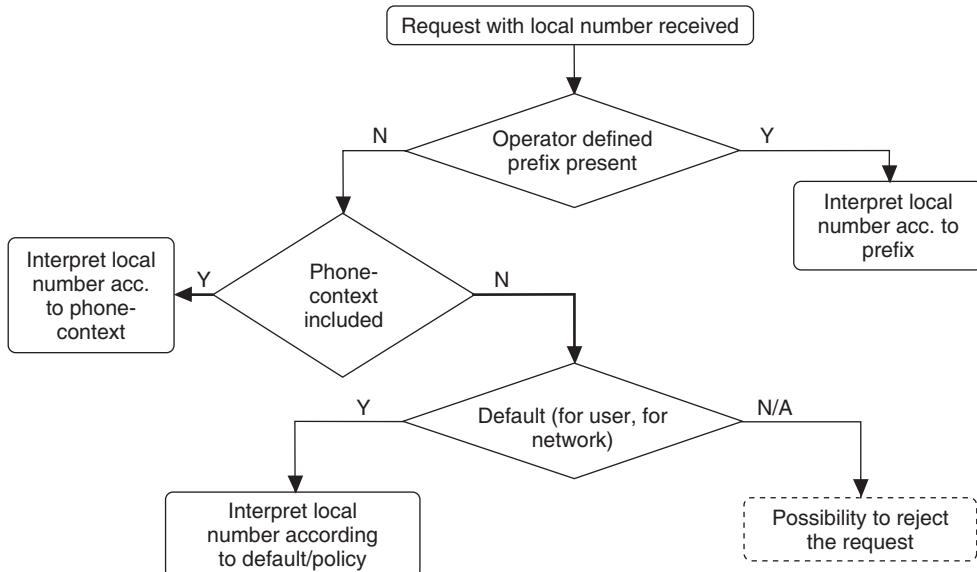


Figure 3.31 Derivation rules for local dialling plans

### 3.16 Support for Local Dialling Plans

Users in circuit switched mobile networks get used to dialling b-party numbers in a particular way because the service is provided by the local network where users are located with. As an example, when a GSM user in Finland dials 1234567 it gets routed to the owner of that number in Finland. But while the GSM user is roaming in the UK the call is connected to the user in the UK. A roaming user is assumed to dial the called party number in international format e.g. +358 50 1234567 when the user wants to reach the called user irrespective of the location of the calling party.

Users in fixed PSTN/ISDN networks also get used to dialling in a particular way. For dialling a number in a particular area the user can directly dial the wanted number like 444-9999 in that particular area but to reach a number in a different area an area code (209) should be added (209-444-9999) and maybe even a suitable prefix for a wanted interconnection carrier. When the user wants to make an international call the country code in addition to the so called national destination code is needed.

As IMS is becoming the common communication platform for users of both fixed and mobile network it is desirable to make all kinds of dialling schemas interchangeable. This is a far from easy task and actually the first two 3GPP IMS releases only supported numbers in international format including a leading ‘+’ sign e.g. tel:+358 50 1234567. Without this limitation a very bad user experience would have occurred. Let’s elaborate this by visiting our first example in this section, but this time the user has dual-mode mobile (CS and IMS capable). A Finnish user is roaming in the UK and calls 1234567; if the call is made via the CS domain then the user in the UK is reached, but if the

same call is made via IMS domain then it would be connected to the user in Finland due to different service control models (see Section 2.1.9). Another problematic example is overlapping service numbers at the visited and home network. For example, a Finnish tourist in Japan dials 118 which is the emergency number in Japan and directory enquiries in Finland. What kind of decision should be taken in this case? Solutions to overcome these problems were finally standardized in 3GPP Release 7.

The fact is that neither the IMS UE nor IMS network can know the user's intention so the solution to handle the so called local dialling to support local numbering plans is based on explicit indication from the user or default policy in the UE or default operator policy at the network. To convey either user's desire or UE's decision to the network two possibilities exist.

1. An operator can define a prefix that could be used to explicitly indicate whether user wishes to connect to a local number in the visited network or a local number in the visited network (e.g. prefix '\*' could mean intention to use visited network dialling plan).
2. UE could include a 'phone-context' parameter to Request-URI field according to the user's preference. Exact coding to indicate visited/home dialling plan has been defined by 3GPP (3GPP TS 24.229):
  - (a) when the 'phone-context' parameter contains access technology information or the home domain name prefixed by the 'geo-local' string in 'phone-context' parameter it means the dialling plan in the visited network;
  - (b) when the 'phone-context' parameter contains the home domain name it is interpreted as a home-local number;
  - (c) when the 'phone-context' parameter contains any other value general procedures for translation apply as defined in RFC 3966.

Like any other request from the UE it gets routed to S-CSCF via the P-CSCF. The S-CSCF routes to request to the application server that analyzes the dialling string as shown in Figure 3.31. If home operators has defined a prefix string(s) to enable subscribers to differentiate dialling using visiting network dialling plan and/or home network dialling plan and if such a prefix is included in the Request URI, the AS interpret the received number in a noninternational format as a number in the visiting network or as a number in the home network according to the prefix. When the prefix is not present but phone 'phone-context' parameter is included in the Request URI, the AS learns the wanted dialling plan by inspecting the content of the parameter (it is an operator's policy in which order previous two checks (prefix vs. phone-context) are performed). Decoding rules are as presented in Step 2 above and examples of local dialling string handling at the network are given in Table 3.10. After finding out the wanted dialling plan the AS translates the dialling string to globally routable SIP URI or an international tel URI and sends the request back to the S-CSCF for routing the request further using ordinary IMS routing mechanisms.

**Table 3.10** Examples of local dialling strings

SIP Request-URI set by the UE	Meaning in the network
tel:545455;phone-context = geo-local.home.net	'geo-local' is an indication for visited numbering plan – > A session attempt to be delivered to number in the visited network.
tel:1234567; phone-context = 216.01.gprs.home1.net	'216.01.gprs.home1.net' is used to indicate that this call is to be treated in the network which country code is 216 and its network code is 01. Remaining part indicates the used access and user's home network.
tel:1234567; phone-context = home1.net	'home1-net' is used to indicate that number should be treated according to home network number plan – > A session attempt to be delivered to number in the home network.
tel:!*1234567	'*' is a prefix to indicate visited numbering plan. – > A session attempt to be delivered to number in the visited network.

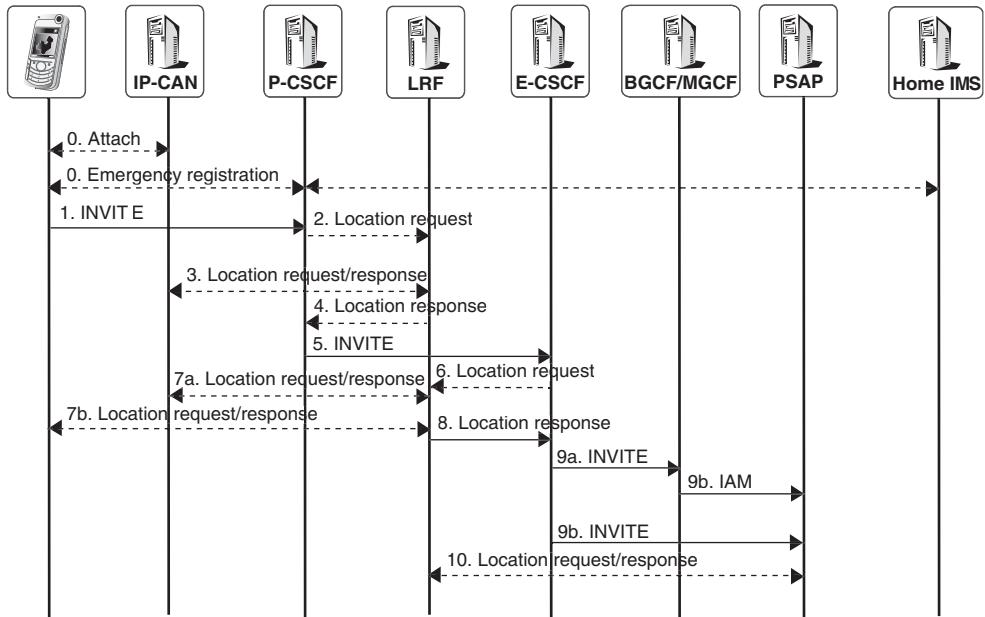
## 3.17 IMS Emergency Sessions

### 3.17.1 Introduction and Architecture

Before 3GPP release 7 the IMS was only able to detect an emergency session attempt and to guide a requesting UE to use an alternative system (i.e. CS domain) to reach Public Safety Answering Point (PSAP). This type of solution was sufficient as the majority of available IMS UEs were mobile phones supporting CS technology. Obviously this level of emergency support was not sustainable because it would prevent deploying the IMS as a standalone service platform. Introduction of fixed broadband access (e.g. ADSL, Packetcable) and WLAN as legally competent IMS accesses caused enough pressure to 3GPP community to standardize IMS emergency calls. Description here is based on 3GPP Release 7 standards as they were at the time of the writing. It is fair to note that IMS emergency session procedures are being revisited again in 3GPP Release 8 and it may trigger some changes to even stable Release 7 solution.

For supporting the IMS emergency session a new CSCF role was introduced in Release 7, the Emergency CSCF (E-CSCF). The main task of E-CSCF is to route the emergency call to the appropriate PSAP or emergency centre based on the location of the UE as indicated by the UE in the session setup signalling. The E-CSCF itself needs to obtain necessary location information to select an appropriate PSAP in case the UE was not able to provide accurate enough location information or when country specific regulations require the network to verify UE provided information. In addition, emergency registration and P-CSCF, S-CSCF and MGCF functions were modified to support emergency sessions).

Figure 3.32 attempts to show a complete flow of an IMS emergency session setup procedure therefore it contains several optional steps (dotted line) which are applied if and only if certain conditions are met.



**Figure 3.32** IMS emergency session setup

Once the UE detects an emergency session request from the end user it first has to decide whether it needs to attach to IP-CAN or not. UE is forced to attach if it does not have IP-CAN connectivity or it is roaming and using home network P-CSCF, as the emergency service is a service of the visited network. In all other cases UE can use existing IP-CAN connection.

### 3.17.2 Emergency Registration

UE is required to perform an IMS emergency registration if it is not located in the home network or is not already registered to the IMS (the home network may also require the UE in the home network to do emergency registration). The emergency registration follows ordinary IMS registration procedures (see Chapter 11) with the following additions: no subscriptions to registration event package (see Section 11.13.6), no UE initiated nor network initiated deregistration (to enable PSAP callback, registration timer is subject to national regulation and possible roaming agreements). The emergency registration is independent from all other user's registrations.

### 3.17.3 Emergency Session Setup

Once an end user needs to reach an emergency dispatcher they are expected to dial one of common emergency numbers such as 911, 112. Actually this dialled number is not sent to the IMS when the IMS UE detects the dialled number to be an emergency number. The UE is expected to translate the dialled emergency number to an emergency service Uniform Resource Name (URN) as specified in RFC5031. This URN can take, for example, the

following values urn:service:sos, urn:service:sos.ambulance, urn:service:sos.fire and the UE places this URN in the Request-URI field of the INVITE request. There are cases when the user is roaming where the UE is unable to detect an emergency session request. In these cases the UE will execute normal IMS session attempt and places the dialled number in the Request-URI field of INVITE request.

When an INVITE request arrives to P-CSCF it needs to analyze the received Request-URI in order to detect all emergency session attempts. For this reason the P-CSCF shall store a configurable list of local emergency service identifiers, i.e. emergency numbers and the emergency service URN, which are valid for the operator to which the P-CSCF belongs. In addition, the P-CSCF shall store a configurable list of roaming partners' emergency service identifiers. If policy requires P-CSCF to reject the request (e.g. IMS emergency sessions are not supported by the operator, user is roaming and emergency session is attempted via home P-CSCF, or an unregistered user makes an emergency session and the operator requires emergency registration) then P-CSCF sends 380 Alternative Service error response containing 3GPP IMS specific XML body stating what actions UE should take. Two actions are specified so far: emergency session indication and emergency registration indication. The first one (indication of emergency session) could be used to inform the UE that network does not support IMS emergency sessions and it should use alternative network i.e. CS for emergency request. The home network P-CSCF can use the second one (emergency registration indication) to re-direct UE to use a visited IMS network when the UE is roaming or to do emergency registration in the home network. When P-CSCF supports and local policy allows the emergency session to continue the P-CSCF selects E-CSCF to handle the request. Before forwarding the request to the E-CSCF the P-CSCF may optionally request and/or verify UE's (e.g. fixed device) location information using Location Retrieval Function (LRF) and it marks UE's undetected emergency sessions by placing emergency service URN in the Request-URI field.

The E-CSCF is responsible for routing an emergency session towards the most appropriate PSAP. PSAP selection is done based on user's location and type of emergency. If the request does not contain location information or it is not accurate enough or the network must verify location information the E-CSCF asks assistance from LRF functionality. The LRF may also perform routing determination function and in this case the LRF responds with the correct PSAP routing address instead of more accurate location information. When correct PSAP is discovered the E-CSCF routes the request via BGCF/MGCF if the PSAP is located in CS domain and directly to PSAP if it is located in PS domain. Figure 3.32 also shows that PSAP may query LRF if it needs more accurate location information or wants updated location information.

## 3.18 SIP Compression

### 3.18.1 Introduction

The IMS supports multimedia services using the SIP call control mechanism. SIP is a client server, text-based signalling protocol used to create and control multimedia sessions with two or more participants. The messages also contain a large number of headers and header parameters, including extensions and security-related information. Setting up a

SIP session is a tedious process involving codec and extension negotiations as well as QoS interworking notifications. In general, this provides a flexible framework that allows sessions with differing requirements to be set up. However, the drawback is the large number of bytes and the many messages exchanged over the radio interface. The increased message size means that:

- Call setup procedures using SIP may take more time to be completed compared with those using existing cellular-specific signalling, which means that the end user will experience a delay in call establishment that will be unexpected and likely unacceptable.
- Intra-call signalling will, in some way or another, adversely affect voice quality/system performance.

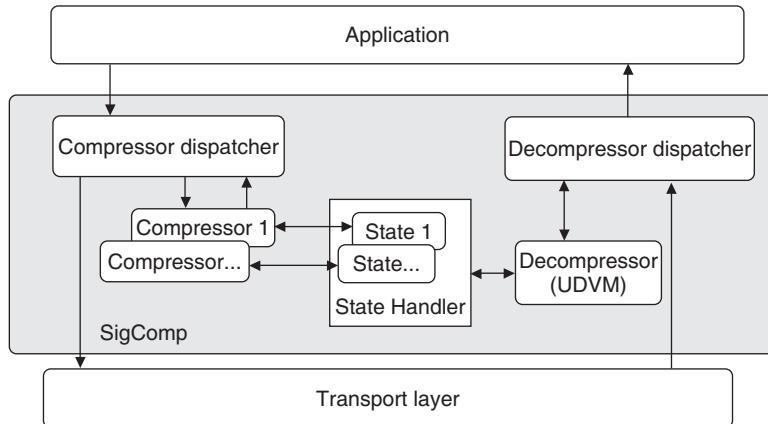
Therefore, support for real-time multimedia applications requires particular attention when SIP call control is used. To speed up session establishment, 3GPP has mandated the support of SIP compression by both the UE and the P-CSCF [3GPP TS 23.221]. Although the support of compression is mandatory, 3GPP was not happy to mandate its usage because in certain accesses the need for SIP compression is not apparent, e.g. in Wireless Local Area Network (WLAN) network.

The IMS uses IETF defined solution called Signalling Compression (SigComp). It is a mechanism that application protocols use to compress messages before sending them across the network. It is presented to applications as a layer between application protocols and transport protocols [RFC3320 and RFC4896]. SigComp uses a Universal Decompressor Virtual Machine (UDVM) to decompress messages. A message that is compressed using SigComp is referred to as a SigComp message.

### 3.18.2 *SigComp Architecture*

SigComp's architecture can be broken down into five entities (see Figure 3.33). The entities are described as follows:

- Compressor dispatcher – this is the interface between the application and the SigComp system. It invokes a compressor, indicated by the application using a compartment identifier. The compressor dispatcher forwards the returned compressed message to its destination.
- Decompressor dispatcher – this is the interface between the SigComp system and the application. It invokes a UDVM that decompresses the message. The decompressor dispatcher then passes the decompressed message to the application. If the application wishes the decompressor to retain the state of the message, it returns what is called a ‘compartment identifier’.
- Compressors – this entity compresses the application message. It uses a compartment that is identified using the compartment identifier. The compressed message is passed to the compressor dispatcher. DEFLATE is an example of a compression algorithm.
- UDVM – This entity decompresses a compressed message. A new instance is invoked for every new SigComp message. UDVM uses the state handler to create a state for a new message or make use of an existing state.
- State handler – this holds information that is stored between SigComp messages (referred to as the ‘state of the messages’). It can store and recover the state.



**Figure 3.33** Signalling compression architecture

### 3.18.3 Compressing a SIP Message in IMS

During the registration phase the UE and the P-CSCF announce their willingness to perform compression by providing details about their compression capabilities, such as memory size and processing power, upload states and compression instructions (see Section 11.10). Due to the strong security requirements, announcements and state creations are only allowed after a security association has been established. Otherwise, a malicious user could upload false states that would make compression vulnerable.

When the UE or P-CSCF wants to send a compressed SIP message, it follows the framework described in [RFC3320], which states that a SIP application in the UE should pass a message to a compressor dispatcher. The compressor dispatcher invokes a compressor, fetches the necessary compression states using a compartment identified by its ID and supplied by the application, and uses a certain compression algorithm to encode the message. Finally, the compressor dispatcher relays the compressed message to the transport layer to be delivered to the remote end (the P-CSCF).

The compressor is responsible for ensuring that the remote end can decompress the generated message. It is possible to include all the needed information in every SigComp message (i.e., every bytecode) to decompress the message. However, this would reduce the archived compression ratio significantly; so, it is better to ask the other end to create states. The information saved in these state items can then be accessed for future SigComp message decompression for messages that arrive from the same source and are related, avoiding the need to upload the data on a per-message basis.

When a decompressor dispatcher receives a message it inspects the prefix of the incoming message. As all SigComp messages contain a prefix (the five most significant bits of the first byte are set to 1) the decompression dispatcher is able to identify that the message is compressed. This prefix does not occur in UTF-8-encoded text messages. The decompressor dispatcher forwards the message to the UDVM, which requests previously created states from the state handler and uses the provided states (or provided bytecode in the message if no states exist) to decompress the message. After decompression the

UDVM returns the uncompressed message to the decompression dispatcher which further passes the message to an application.

## 3.19 Combination of CS and IMS Services – Combinational Services

### 3.19.1 Introduction

Whilst the IMS is designed as a standalone system to provide multimedia services to its users, it can also be used to provide additional functionalities within existing telephony networks. During the initial phase of IMS deployment mobile network users will make use mainly of the CS domain of the mobile networks for audio sessions. In such CS calls the IMS is not required for provision of the basic service – i.e., the signalling and media connection between the two users, as well as the QoS.

But also such CS calls should benefit from the flexible and broad service provisioning capabilities of IMS. New work has been started by 3GPP recently, to enable such so-called ‘combinational services’. As the work is still in progress at the time of writing, only the basic principles can be shown here.

In order to enable such a service combination, the two UEs first need to become aware of the supported capabilities at the other end. Section 3.19 shows how IMS-related capabilities can be exchanged after a CS call has been established between two users. In Section 3.19 it is shown how, based on that capability exchange, an IMS service can be invoked that enhances the ongoing CS call.

### 3.19.2 Capability Exchange

In the following example we will assume that two users are connected over the CS domain – i.e., they have a CS connection between them. In this example Tobias called his sister via her international phone number. In order to allow the combinational services to work, several conditions have to be fulfilled:

- within CS signalling, the calling network needs to provide Tobias’s phone number (the MSISDN) in full international format to Theresa’s UE – e.g., +44123456789;
- Tobias’s MSISDN also needs to be registered in Tobias’s home IMS network as a tel URL – e.g. tel:+44123456789;
- also, Theresa’s MSISDN must be sent via CS signalling to Tobias’s UE. Theresa’s MSISDN must be indicated in international number format and must be registered in Theresa’s IMS home network as a tel URL – e.g. tel:+36987654321; and
- both Tobias and Theresa need to be registered with their tel URLs from the very same UEs they are using to establish CS calls with each other.

In order to simplify the example, we assume that both users, Tobias and Theresa, are each registered only from one UE. This is done in order to avoid forking cases. Theresa and Tobias both have an IMS-based application installed on their UEs that allows them to share videostreams in real time in addition to an existing CS voice call. This peer-to-peer application will only work if the application is present on both UEs. In order to give a consistent user experience, the menu item for ‘video sharing’ will only be shown to

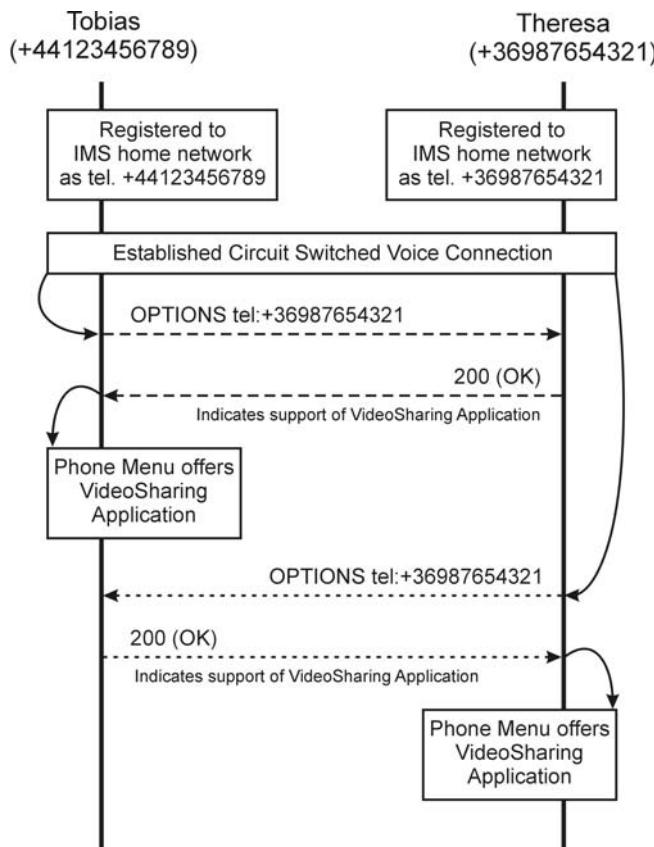
Theresa and Tobias, if their UEs have verified that the application is also supported at the remote end.

In order to query the capabilities of the remote end the SIP OPTIONS request is used (see Fig 3.34). After establishing the CS call, both UEs send an OPTIONS request to the tel URL of the remote user. The tel URL will be derived from the MSISDN of the remote user in the CS call. In our example, Tobias's UE sends an OPTIONS request to the following address:

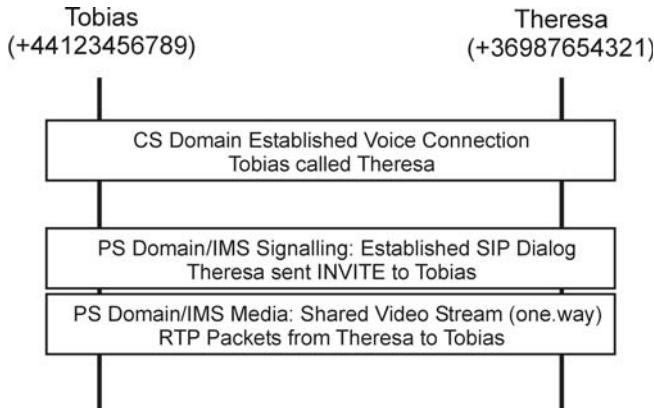
```
OPTIONS tel:+36987654321 SIP/2.0
```

Theresa's UE will send a 200 (OK) response to this OPTIONS request, indicating the UE's capabilities. One of these capabilities will be support for the video-sharing application. The same OPTIONS/200 (OK) exchange is done in the other direction. After these exchanges have been finished, both Theresa and Tobias will be offered the possibility of sharing a videotostream with the other user in the CS call.

As said above, the example given here is very basic and only shows how capabilities between UEs can be exchanged in principle.



**Figure 3.34** Capability exchange during an ongoing CS call



**Figure 3.35** Example for parallel connections when combining IMS and CS services

### 3.19.3 Parallel CS and IMS Services

After the exchange of capabilities, one of the users can now start to share a live video. In our example Theresa wants to send the video to Tobias. The session setup between the two users will work as described in Section 11.10 (Alternative session setups). After this session has been established, Theresa can send a videotostream via GPRS to Tobias. Figure 3.35 shows the connections that exist in parallel.

The SIP-based connection between the two users can either be released whilst the CS call is still ongoing or the video-sharing application will automatically release it once the CS call has ended. This is purely an application-specific behaviour.

## 3.20 Voice Call Continuity

### 3.20.1 Introduction

Widely deployed radio networks (GERAN, EDGE, WCDMA) and GPRS packet core are not currently either capable or optimized for providing VoIP as a primary end user speech service and most likely this will stay this way. New 3GPP radio technologies such as HSPA and LTE will provide higher spectral efficiency and lower battery consumption than current CS Voice and once these new radio technologies are widely available and deployed it is fair to assume that mobile operators will gradually offer VoIP as a primary speech service. Meanwhile it is assumed that VoIP in mobile devices are going to be offered with non 3GPP specific radio technologies such as WLAN. Typically WLAN is available in hotspots such as homes, offices, coffee houses, airports etc and this does not fulfil end user expectations in terms of universal reachability. In other words, if you initiate a voice call over WLAN and you move outside of WLAN coverage (hotspot) then you will experience service discontinuity because your voice call is dropped. In contrast in some countries (like the US) indoor coverage for wide area networks may be limited meaning that once you enter office or home ordinary GSM call may drop. To overcome this challenge a feature called Voice Call Continuity was specified in Release 7. It is a feature that enables the user to continue a voice call as they move between the

3GPP circuit switched (CS) domain and IMS VoIP. For example, it allows continuing VoIP calls when leaving the WLAN coverage in the office or at home or public hotspot. Current VoIP solutions from IT and Internet VoIP providers are restricted to office/home due to the location of the VCC functionality in either a business premises or a WiFi access point. A VCC service could provide a clear competitive advantage for a mobile operator allowing them to differentiate their VoIP offering. On the other hand, VCC allows fixed network operators to play the mobile game by offering VoIP at home and, in conjunction with a roaming agreement, GSM speech service outside. In addition to the capability to transfer an ongoing call from one access to other access the VCC functionality provides terminating domain selection when a call is made. In other words, the VCC functionality decides whether the incoming communication request is delivered via IMS or via CS.

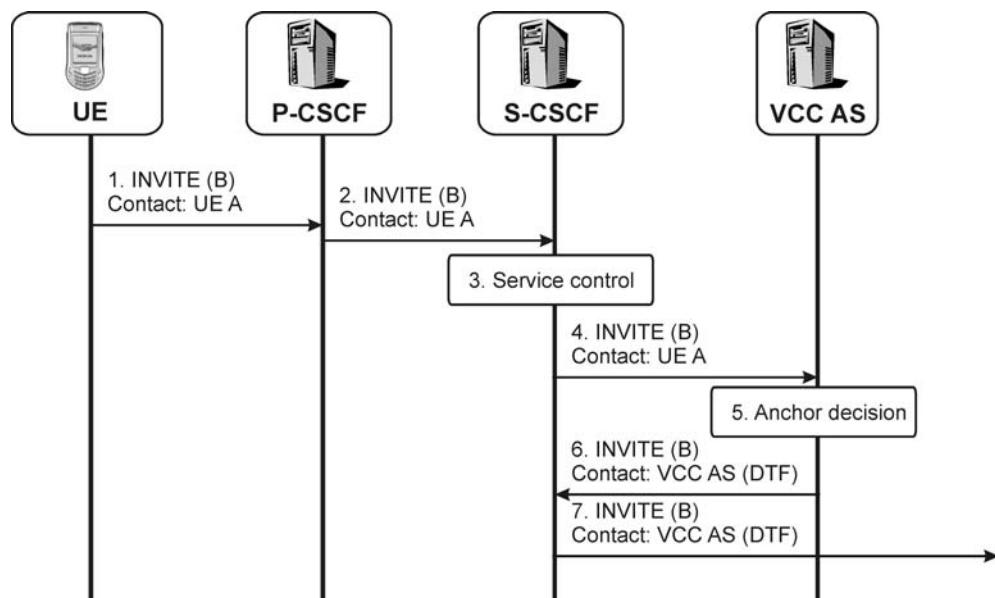
### *3.20.2 Voice Call Continuity Functionality*

To enable VCC functionality it is necessary to have a specific UE, the VCC UE, and a specific application in the network, the VCC application, in addition CS core needs to have a mechanism by which to route mobile originated and mobile terminated calls to the VCC application. VCC UE implements the following additional functions compared to ordinary IMS UE: domain selection function (DSF) for mobile originated calls and the domain transfer function (DTF). Domain selection function uses both end user preferences, stored operator preferences and of course the given radio access capability to select whether voice call is made via CS or IMS. The domain transfer function moves an ongoing call from one domain to another when it detects conditions requiring a transfer (e.g. WLAN coverage gets weaker or UE enters WLAN coverage area). The VCC application comprises a set of functions required for a VCC UE to establish voice calls and control (allow or reject transfer requests) the switching of the VCC UE's Access Leg between the CS domain and IMS whilst maintaining the active session. VCC application has correspondingly DSF and DTF, but it also contains necessary functions to enable routing back and forth from CS to IMS and vice versa. In standards (3GPP TS 23.206, 24.206) terms CS Adaptation Function and CAMEL Service hosted by a gsmSCF are used to describe these routing functionalities. At the time of writing interactions between these functions are not specified and therefore we focus here on describing the VCC application itself not its sub functions.

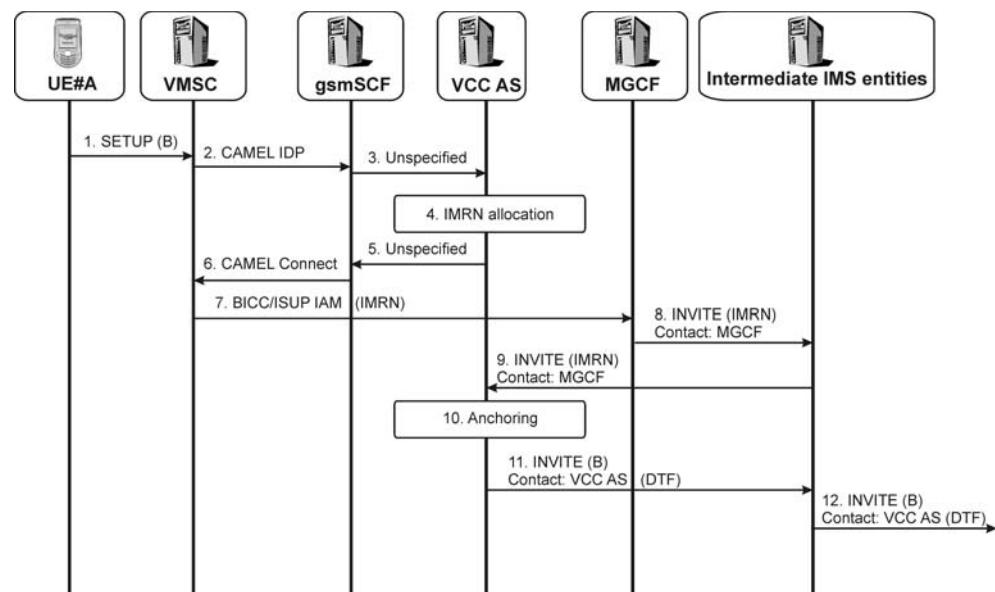
### *3.20.3 Voice Call Continuity Session Initiation and Termination*

When a user makes a voice call with her VCC enabled UE the domain selection function at the UE first makes a decision which domain is used to setup a call (CS or IMS). After domain selection it sends a setup request to the network. Note: There is nothing VCC specific in this request nor in SIP or in CS signalling request. IMS originated session setup is shown in Figure 3.36 and CS originated call setup is shown in Figure 3.37.

From Figure 3.36 we can see that the INVITE request in Steps 1–2 follows ordinary originating IMS routing principles i.e. the request traverses through P-CSCF and S-CSCF. At Steps 3–4 the originating S-CSCF executes originating initial filter criteria and for VCC users it contains a trigger to send INVITE requests to the VCC application for further processing. At Step 5 the VCC application acts as SIP B2BUA and anchors (inserts itself



**Figure 3.36** Voice call continuity and IMS originated call

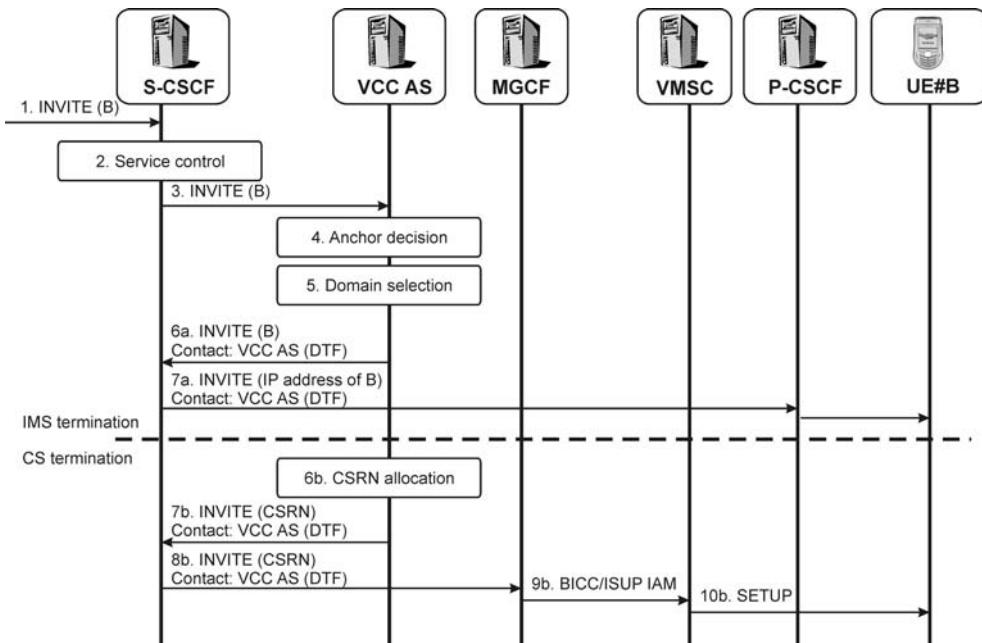


**Figure 3.37** Voice call continuity and CS originated call

in the call) the call and creates a new INVITE towards the B-party including its own contact information. From Step 6 onwards normal IMS procedures continue.

Required functionality in CS originated calls is a bit more complicated than in previously explained IMS originated calls. This is due to the fact that entity in CS core, visited MSC (VMSC), must send all VCC user originated call request to the IMS for anchoring the calls. For VCC user an operator needs to define appropriate CAMEL triggers. It means that an originating call attempt at VMSC causes the VMSC to invoke signalling towards gsmSCF (step 2). The gsmSCF instructs the VMSC to route the call towards the IMS (step 6). After obtaining routing instructions from gsmSCF the VMSC sends request further towards obtained IMRN (step 7). As it points to the user's home IMS network it hits MGCF like all other CS originated requests towards IMS. The MGCF converts incoming request to SIP INVITE requests and sends it to I-CSCF. From I-CSCF to VCC application the routing follows PSI routing principles (explained in Section 12.11). At Step 9 the request arrives to the VCC application and it performs call anchoring (step 10) like it does for IMS originated calls. In addition it must restore original called party address to the SIP Request-URI header. After all this VCC application returns the request back to the IMS and normal IMS originating session handling continues which either means routing towards the visited IMS operator or routing towards CS if called party is CS user (step 11).

Figure 3.38 reveals how session termination works when VCC is enabled at the IMS network. The request gets routed to the terminating S-CSCF as per normal IMS routing principles as explained in Section 3.4. For VCC subscriber an operator needs to define



**Figure 3.38** Voice call continuity and terminated call

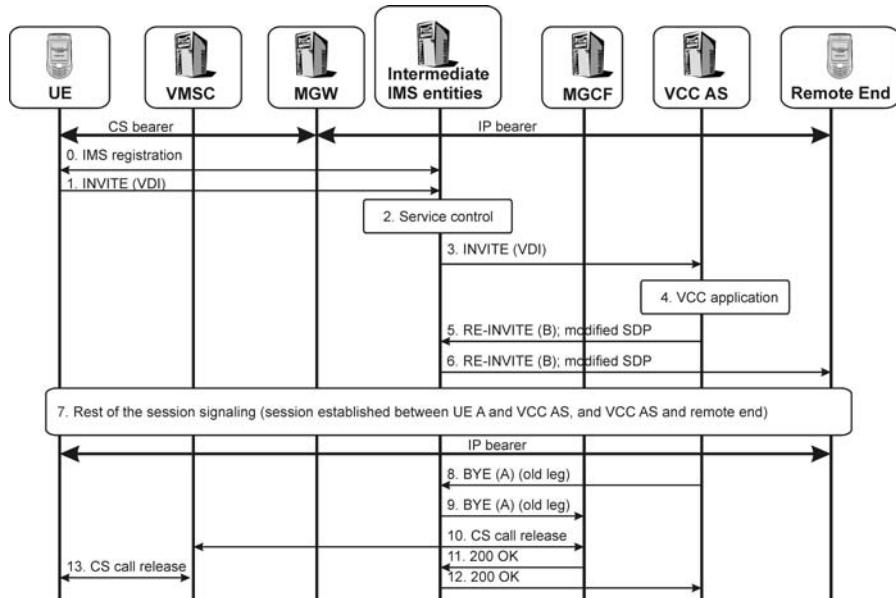
appropriated initial filter criteria that routes all incoming INVITE requests with speech media to a VCC application (Steps 2 and 3). Once the VCC application gets the request it must anchor the call in order to prepare for possible handover request later on and it needs to select which termination option is to be used. There are a number of different conditions that can be taken into account when selecting either CS or IMS as a termination domain. Possible dynamic conditions are, for example, UE's registration status in CS and/or IMS, the last known access network type (e.g. GERAN or WLAN), offered media components in an incoming IMS session, a domain used by an existing call already anchored. More static conditions are, for example, registered UE's capability for VoIP (user may have more than one UE and not all of them necessarily support VoIP), user preference and/or operator policy. When VCC application chooses termination via IMS domain then Steps 6a and 7a occur i.e. VCC application adds itself as a contact point and sends the modified INVITE request back to the S-CSCF which then performs normal IMS session termination. When VCC application chooses termination via CS domain then Steps 6b–10b occur. First VCC application obtains CS domain Routing Number (CSRN),<sup>7</sup> then it modifies the INVITE request by inserting the CSRN as a target address and itself as a contact point and finally it sends the modified request back to the S-CSCF. The S-CSCF analyzes the target address and learns that the request must break to the CS and to do so it passes the request to MGCF via BGCF (not shown in the figure due to simplicity). Once the MGCF gets the request it performs protocol translation and based on CSRN number analysis finds correct VMSC serving the UE. Finally it sends Initial Address Message (IAM) to the VMSC for executing normal CS call termination.

### 3.20.4 Voice Call Continuity Domain Transfer Procedure

Domain transfer denotes the ability to change the access domain from IMS to CS and vice versa, while a voice call is ongoing. This is realized by third party call control execution in VCC application (domain transfer function). During session initiation/termination the VCC application has placed itself in the signalling path and it has bound two dialogs together. The first dialog is between the served UE and the second dialog is between VCC application and the other UE. When UE wishes to switch between domains it makes a domain transfer request (session) from the new domain. This call is routed to the VCC application as any other call of the VCC subscriber. The VCC application recognizes that this attempt was made towards a VCC specific identifier and it understands how to find the call which this request relates to. As a consequence the VCC application realizes that this new call will replace the existing dialog between the requesting UE and itself and it also re-invites the remote end of the original call with new session descriptions copied from the domain transfer request. This results in the bearers between the far-end device and the domain transferred UE being interconnected. When the domain transfer has been completed successfully the old dialog between the requesting UE and the VCC application will be released by the VCC application.

---

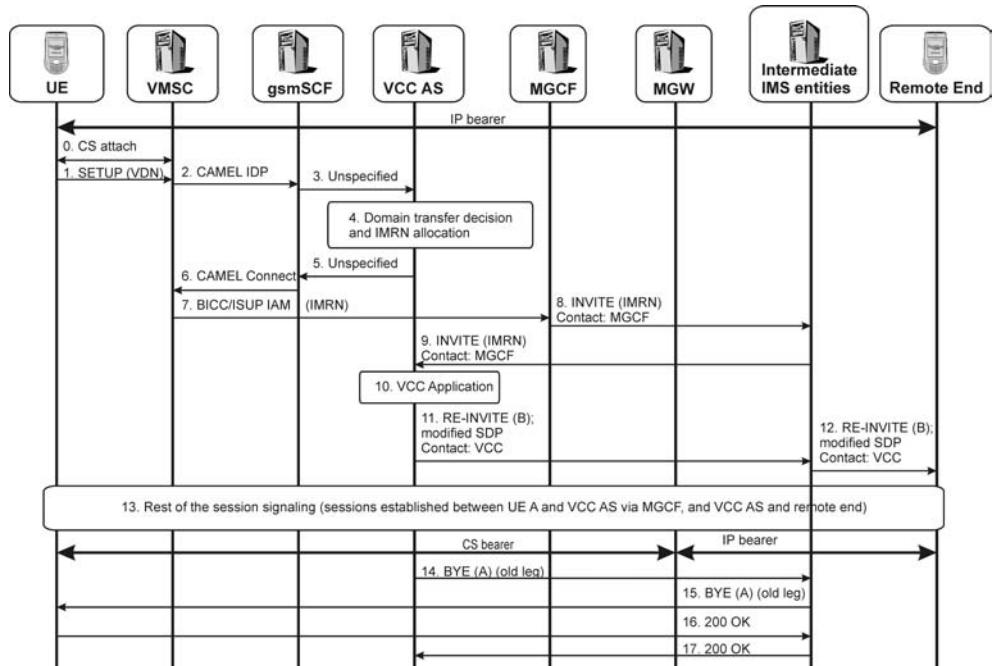
<sup>7</sup> The CSRN is a dynamic routing number used for routing into CS domain, i.e. to find the outgoing MGCF and then the terminating user's VMSC.



**Figure 3.39** Domain transfer from CS to IMS

Figure 3.39 shows how domain transfer from CS to IMS happens. Once the UE has registered itself to the IMS it can request domain transfer. It does it by sending SIP INVITE request with wanted medias targeted to VCC Domain Transfer URI (VDI) e.g. `sip:domain.xfer@dtf1.home1.net`. This VDI is treated like PSI in the IMS and it gets routed to the VCC application that executes domain transfer by sending a SIP RE-INVITE with a modified media description (e.g. new user-plane IP address and port pointing to the UE instead of MGW). The remote end accepts the session modification request and stops sending media to the MGW and after completing session modification it starts sending and receiving media directly with the UE. Once all this is successfully completed the VCC application decides to tear down the existing dialog between itself and the requesting UE and it sends a BYE request towards to the UE. This request will be routed via MGCF which converts the request to appropriated ISUP/BICC signalling message. In addition the MGCF releases resources from MGW.

Figure 3.40 respectively explains how domain transfer from IMS to CS happens. Steps 1–7 are almost identical with Figure 3.37 and its description. The only difference is instead of called party number a specific VCC Domain Transfer Number (VDN) is used. At Step 8 the VCC application executes domain transfer by sending a SIP RE-INVITE with a modified media description (e.g. new user-plane IP address and port pointing to the MGW instead of UE). The remote end accepts the session modification request and stops sending media to the UE and after completing session modification it starts sending and receiving media via the MGW. Once all this is successfully completed the VCC application decides to tear down the existing dialog between itself and the requesting UE by sending a BYE request to the UE.



**Figure 3.40** Domain transfer from IMS to CS

### 3.20.5 Supplementary Services

All basic supplementary services during session setup and during ongoing call can be provided e.g. call barring, call diversion. However, VCC functionality usage imposes restrictions when domain transfer is executed. According to 3GPP Release 7 architecture the following supplementary services cannot be maintained between domains if the domain transfer procedure occurs: call hold, call waiting, conference call, multiparty call. 3GPP is improving the solution in Release 8 under a work item called IMS centralized services.

## 3.21 Security Services in the IMS

This section is intended to explain how security works in the IMS. It is intentionally thin in cryptography and, thus, will not discuss algorithms and key lengths in depth, nor will it perform any cryptanalysis on IMS security. There are many other books and sources giving detailed information on this issue.<sup>8</sup>

This chapter will give a high-level view of the security architecture and explain the components of that architecture, including the models and protocols used to provide the required security features. After reading this chapter the reader should be familiar with

<sup>8</sup> See, for example, V. Niemi and K. Nyberg (2003) *UMTS Security*, John Wiley & Sons Ltd., Chichester, UK.

the main concepts in the IMS security architecture and understand the underlying models, especially those related to trust and identity that shape IMS security as a whole.

### 3.21.1 *IMS Security Model*

The IMS security architecture consists of three building blocks, as illustrated in Figure 3.41. The first building block is Network Domain Security (NDS) [3GPP TS 33.210], which provides IP security between different domains and nodes within a domain.

Layered alongside NDS is IMS access security [3GPP TS 33.203]. The access security for SIP-based services is a self-sustaining component in itself, with the exception that the security parameters for it are derived from the UMTS Authentication and Key Agreement (AKA) Protocol [3GPP TS 33.102]. AKA is also used for bootstrapping purposes – namely, keys and certificates are derived from AKA credentials and subsequently used for securing applications that run on the Hypertext Transfer (or Transport) Protocol (HTTP) [RFC2616], among other things – in what is called the Generic Authentication Architecture (GAA) [3GPP TS 33.220].

Intentionally left out of this architectural model are those security layers that potentially lie on top of the IMS access security or run below the NDS. For example, in the UMTS the radio access layer implements its own set of security features, including ciphering and message integrity. However, the IMS is designed in a way that does not depend on the existence of either access security or user-plane security.

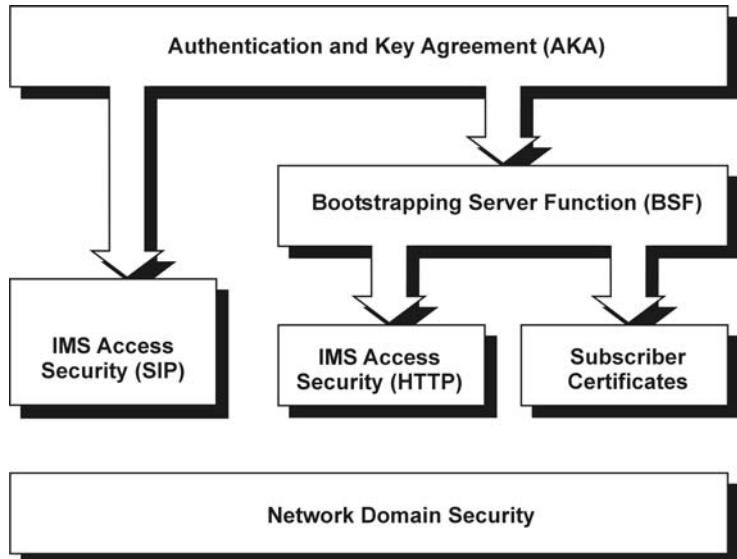
### 3.21.2 *Authentication*

#### 3.21.2.1 **Overview of IMS Authentication Methods**

Whenever a user wants to access the IMS network of his or her home operator, the user will be authenticated, this means that the network will make sure that it is really the user who is accessing the network rather than a malicious user.

Authentication in IMS is achieved in different ways within IMS, mostly dependant on the access network technology used and the network operator's preferences. The following authentication mechanisms are currently defined in IMS:

- 3GPP Authentication and Key Agreement (AKA), which makes use of the general 3GPP AKA mechanism that is also used within third generation CS and GPRS networks. 3GPP AKA relies on a shared secret between the user (stored in the UICC card) and the network (stored in the HSS) and is performed automatically without any user interaction – see Section 3.21.2.2;
- Network Access Subsystem (NASS)-IMS-bundled authentication (NBA), which is an authentication method that is mostly used by fixed/TISPAN networks and which relies on the available security of lower layers in order to authenticate the user. In the case of NBA no IMS or SIP specific authentication procedure is required – see Section 3.21.2.3;
- GPRS-IMS-bundled Authentication (GIBA), formerly called ‘early IMS security’, which is deployed within 3GPP based networks which does not provide as yet the infrastructure for full IMS security, e.g. networks with early IMS deployments that did not make use of IPsec and 3GPP AKA for IMS. GIBA relies on the security of the GPRS layer and therefore no specific IMS or SIP authentication procedures are



**Figure 3.41** Security architecture of the IMS

required. GIBA is described in detail in Section 11.16, where the differences between 3GPP AKA and GIBA will be outlined;

- From 3GPP Release 8 onwards, HTTP digest will be an authentication method within IMS. HTTP digest is the defined by IETF in [RFC 2617]. It is based on a shared username and password between the user and the network (where it is stored in the HSS). In the case of HTTP digest, the user needs to remember the username and password and has to provide them during the authentication procedure, i.e. there is not UICC-like storage, which would allow automated authentication.

### 3.21.2.2 Authentication and Key Agreement (AKA)

Security in the IMS is based on a long-term secret key, shared between the ISIM and the home network's Authentication Centre (AUC). The most important building block in IMS security is the ISIM module, which acts as storage for the **shared secret ( $K$ )** and accompanying **AKA algorithms**, and is usually embedded on a smartcard-based device called the Universal Integrated Circuit Card (UICC). Access to the shared secret is limited. The module takes AKA parameters as input and outputs the resulting AKA parameters and results. Thus, it never exposes the actual shared secret to the outside world.

The device on which the ISIM resides is tamper resistant, so even physical access to it is unlikely to result in exposing the secret key. To further protect the ISIM from unauthorized access, the user is usually subject to user domain security mechanisms. This, in essence, means that in order to run AKA on the ISIM, the user is prompted for a PIN code. The combination of ownership – i.e., access to a physical device (UICC/ISIM) and knowledge of the secret PIN code – makes the security architecture of the IMS robust. An

**Table 3.11** Authentication and key agreement parameters

AKA Parameter	Length (bits)	Description
K	128	Shared secret; authentication key shared between the network and the mobile terminal
RAND	128	Random authentication challenge generated by the network
AUTN	128	(Network) authentication token
SQN	48	Sequence number tracking the sequence of the authentication procedures
AUTS	112	Synchronization token generated by the ISIM upon detecting a synchronization failure
RES	32–128 <sup>9</sup>	Authentication response generated by the ISIM
CK	128	Cipher key generated during authentication by both the network and the ISIM
IK	128	Integrity key generated during authentication by both the network and the ISIM

attacker is required to have possession of both ‘something you own’ and ‘something you know’, which is difficult, as long as there is some level of care taken by the mobile user.

AKA accomplishes mutual authentication of both the ISIM and the AUC, and establishes a pair of cipher and integrity keys. The authentication procedure is set off by the network using an authentication request that contains a random challenge (RAND) and a network authentication token (AUTN). The ISIM verifies the AUTN and in doing so verifies the authenticity of the network itself. Each end also maintains a sequence number for each round of authentication procedures. If the ISIM detects an authentication request whose sequence number is out of range, then it aborts the authentication and reports back to the network with a synchronization failure message, including with it the correct sequence number. This is another top-level concept that provides for anti-replay protection.

To respond to the network’s authentication request, the ISIM applies the secret key on the RAND to produce an authentication response (RES). The RES is verified by the network in order to authenticate the ISIM. At this point the UE and the network have successfully authenticated each other and as a by-product have also generated a pair of session keys: the Cipher Key (CK) and the Integrity Key (IK). These keys can then be used for securing subsequent communications between the two entities. Table 3.11 lists some of the central AKA parameters and their meaning.

### 3.21.2.3 NASS-IMS-Bundled Authentication (NBA)

Within a TISPAN NGN the IMS UE is usually attached via a direct link to a Network Access Subsystem (NASS), which provides the lower layer transport (such as IP) between the UE and the network.

By which methods the UE is authenticated by the NASS on the lower layer will not be treated here in detail, as there are several different possibilities. For example, it could

---

<sup>9</sup> Note that the short key lengths are to accommodate backwards compatibility with 2G authentication.

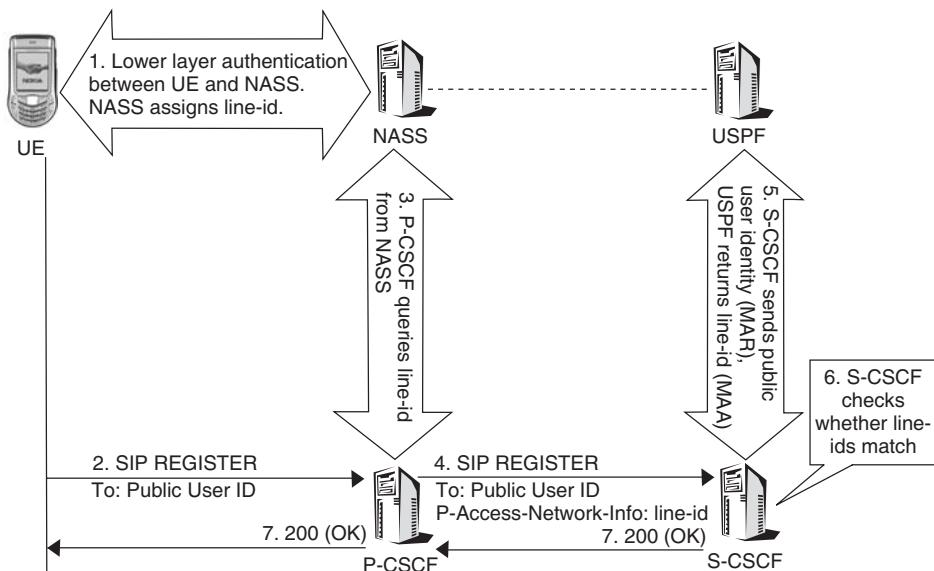
be that the UE is authenticated by means of HTTP digest as described in [RFC 2617]. Another example could be, that it is actually a legacy PSTN UE accessing the NASS and that a gateway, located between the UE and the NASS, is translating the PSTN and IP procedures.

The NASS belongs to the access network (i.e. it is located between the UE and the P-CSCF) and consists of a variety of network elements (which form the subsystem), which are not looked at more closely here.

The UE within a TISPAN NGN is assumed to have a fixed connection to the NGN, which is not changing permanently, such as in mobile networks (e.g. in 3GPP UMTS networks). Once the UE is switched on, it requests an IP address from the NASS and whilst this address is assigned, the UE gets authenticated on the lower layer. The UE is identified from now on by the configured line identification (line-id) which is stored in the NASS and is used as a handle for the authenticated user.

Once the UE is attached to the NGN it can perform IMS registration, during which the user authentication will take place (see Figure 3.42). The UE sends a SIP REGISTER request towards the P-CSCF. The REGISTER request will not include any authentication related information. The P-CSCF becomes aware of the IP address of the UE and knows, based on local configuration and operator policy, that this address was assigned by a specific NASS. Based on this, the P-CSCF queries the NASS, which returns the line-id. The P-CSCF then includes the line-id in the P-Access-Network-Info header (see Section 11.11.1) in the REGISTER request and sends the request towards the I-CSCF and S-CSCF, based on normal IMS procedures.

Once the REGISTER request reaches the S-CSCF, the S-CSCF needs to take care that the user gets authenticated. It therefore sends a Diameter Multimedia-Auth Request (MAR) to the HSS, indicating the public user identity under registration – see Section 11.6.4.



**Figure 3.42** NASS bundled authentication

The HSS also has a connection towards the NASS, which has not been standardized, i.e. it is left to the individual implementation of HSS and NASS, how this connection can be achieved. In any case, the HSS is aware that the UE, which is currently registering, has attached to the HSS. The HSS also knows the details of the UEs lower layer authentication and therefore replies to the S-CSCF in a Multimedia-Auth Answer (MAA) including the line-id as well as the user profile of the user.

The S-CSCF now compares the line-id that it received in the REGISTER request from the UE within the P-Access-Network-Info header and the line-id that it received from the USPF in the MAA. If both match, the user is authenticated and the S-CSCF will send back a 200 (OK) response to the REGISTER request immediately, without further challenging the UE.

With this the NBA procedures do not authenticate the user or the UE itself, they fully rely on the authentication that already took place on the lower layers when the UE attached to the NASS.

### 3.21.3 Network Domain Security (NDS)

#### 3.21.3.1 Introduction

One of the main identified weaknesses of 2G systems is the lack of standardized security solutions for core networks. Even though radio access from the mobile terminal to the base station is usually protected by encryption, nodes in the rest of the system pass traffic in the clear. Sometimes these links even run over unprotected radio hops, so an attacker that has access to this medium can fairly easily eavesdrop on communications.

Having learned from these shortcomings in 2G, 3G systems have set out to protect all IP traffic in the core network. Network Domain Security (NDS)<sup>10</sup> accomplishes this by providing confidentiality, data integrity, authentication and anti-replay protection for the traffic, using a combination of cryptographic security mechanisms and protocol security mechanisms applied in IP security (IPsec).

#### 3.21.3.2 Security Domains

Security domains are central to the concept of NDS. A security domain is typically a network operated by a single administrative authority that maintains a uniform security policy within that domain. As a result, the level of security and the installed security services will, in general, be systematically the same within a security domain.

In many cases a security domain will correspond directly to an operator's core network. It is, however, possible to run several security domains, each pertaining to a subset of the operator's entire core network. In the NDS/IP the interfaces between different security domains are denoted as Za, while interfaces between elements inside a security domain are denoted as Zb. While use of the Zb interface is, in general, optional and up to the security domain's administrator, use of the Za interface is always mandatory between different security domains. Data authentication and integrity protection is mandatory for both interfaces, while the use of encryption is optional for the Zb, as opposed to being recommended for the Za.

---

<sup>10</sup> To specifically indicate that the protected traffic is IP based, NDS is usually denoted by the abbreviation NDS/IP.

The IMS builds on the familiar concept of a home network and a visited network. Basically, two scenarios exist, depending on whether the IMS terminal is roaming or not. In the first scenario the UE's first point of contact to the IMS, called the P-CSCF, is located in the home network. In the second scenario the P-CSCF is located in the visited network, meaning that the UE is, in fact, roaming in such a way that its first point of contact to the IMS is not its home network. These two scenarios are illustrated in Figure 3.43.

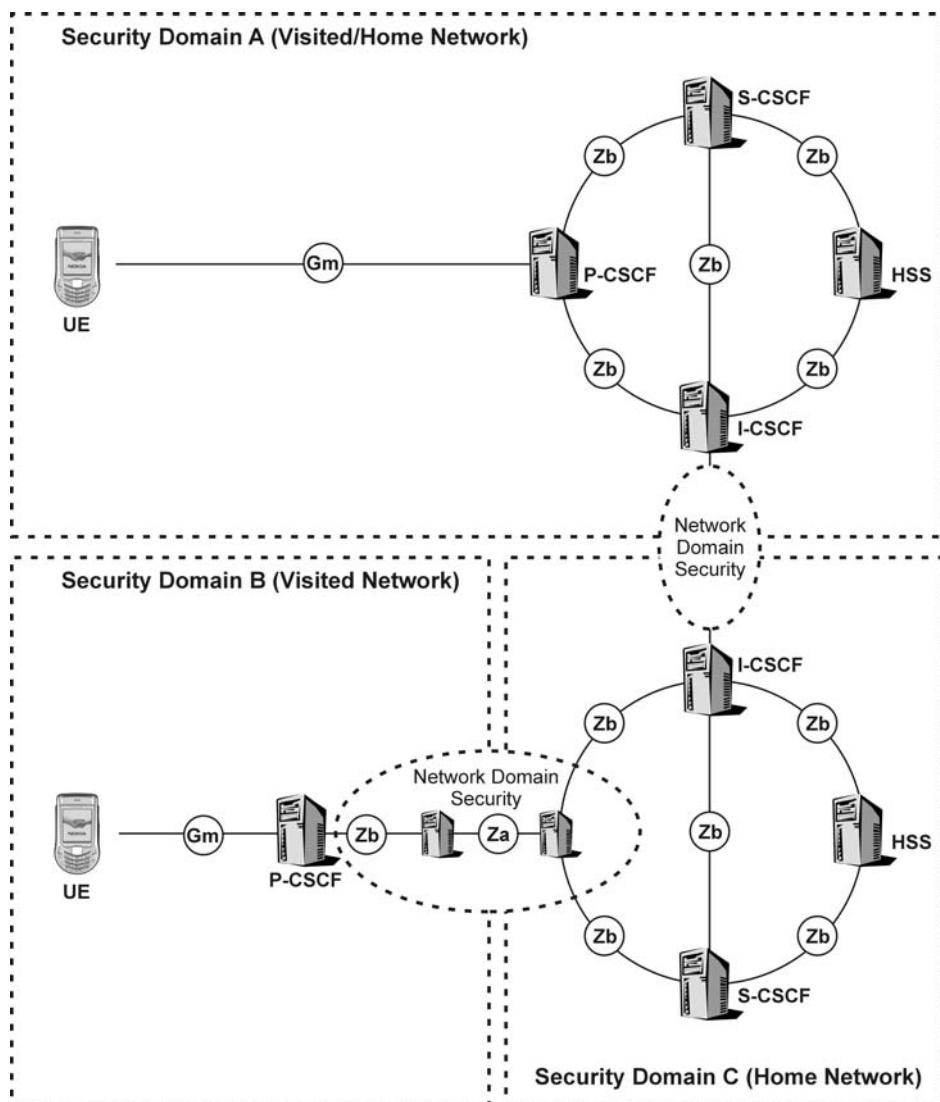


Figure 3.43 Security domains in the IMS

Quite often, an IMS network corresponds to a single security domain and, therefore, traffic between the operator's IMS networks is protected using the NDS/IP. The same also applies in the above-mentioned second scenario, where traffic between the visited and the home network is also protected using the NDS/IP.

In the IMS the NDS/IP only protects traffic between network elements in the IP layer; so, further security measures are required. These will be covered in subsequent chapters. Most importantly, the first hop<sup>11</sup> in terms of SIP traffic is not protected using the NDS/IP, but is protected using IMS access security measures [3GPP TS 33.203]. As will be explained in subsequent chapters, the above scenario in which the IMS elements are split across the home network and the visited network and, therefore, different security domains, requires some special care in terms of authentication and key distribution.

### 3.21.3.3 Security Gateways

Traffic entering and leaving a security domain passes through a Security Gateway (SEG). The SEG sits at the border of a security domain and tunnels traffic toward a defined set of other security domains. This is called a hub-and-spoke model; it provides for hop-by-hop security between security domains. The SEG is responsible for enforcing security policy when passing traffic between the security domains. This policy enforcement may also include packet filtering or firewall functionality, but such functionality is the responsibility of the domain administrator.

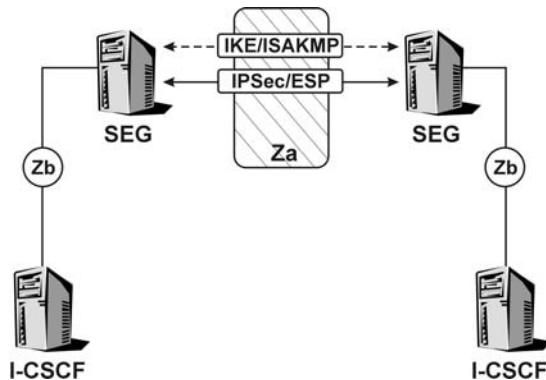
In the IMS all traffic within the IMS core network is routed via SEGs, especially so when the traffic is inter-domain, meaning that it originates from a different security domain from the one where it is received. When protecting inter-domain IMS traffic, both confidentiality as well as data integrity and authentication are mandated in the NDS/IP.

### 3.21.3.4 Key Management and Distribution

Each SEG is responsible for setting up and maintaining IPsec Security Associations (SAs) [RFC2401] with its peer SEGs. These SAs are negotiated using the Internet Key Exchange (IKE) [RFC2409] protocol, where authentication is done using long-term keys stored in the SEGs. A total of two SAs per peer connection are maintained by the SEG: one for inbound traffic and one for outbound traffic. In addition, the SEG maintains a single Internet Security Association and Key Management Protocol (ISAKMP) SA [RFC2408], which is related to key management and used to build up the actual IPsec SAs between peer hosts. One of the key prerequisites for the ISAKMP SA is that peers are authenticated. In the NDS/IP, authentication is based on Pre-shared Secret Keys (PSKs). In 3GPP Release 6, NDS is extended so that other authentication mechanisms than the PSK can also be used between the SEGs. The NDS Authentication Framework (NDS/AF) [3GPP TS 33.310] defines the Public Key Infrastructure (PKI), trust model and mechanisms for authentication

---

<sup>11</sup> Referring here to the interface between the UE and the P-CSCF, denoted as Gm.



**Figure 3.44** NDS/IP and SEGs

of SEGs using public key certificates and RSA signatures. The trust model for the PKI required by NDS/Af includes two modes for cross-certification:<sup>12</sup>

- Manual cross-certification: in this mode, the authorities decide to trust another domain authority. This mode has difficulty in scaling, since each remote security domain needs a valid, locally trusted (signed) certificate. This limitation is similar to the limitations found in the baseline NDS/IP, using PSKs.
- Cross-certification using a bridge Certificate Authority (CA): in this mode, a broker entity or a clearinghouse applies 1-to-1 certification, and the different security domains trust each other via transitive trust (see Section 3.19.4.2 for a discussion of transitive trust). This limits the amount of certification required, since a single domain authority only needs cross-certification with the bridge CA.

Figure 3.44 illustrates the overall NDA/IP model.

The security protocol used in the NDS/IP for encryption, data integrity protection and authentication is the IPsec Encapsulating Security Payload (ESP) [RFC2406] in tunnel mode. In tunnel-mode ESP the full IP datagram including the IP header is encapsulated in the ESP packet. For encryption, the 3DES [RFC1851] algorithm is mandatory, while for data integrity and authentication both MD5 [RFC1321] and SHA-1 [RFC2404] can be used. For the specifics of IPsec/IKE and ESP please refer to Chapter 22, where these protocols are discussed more extensively.

### 3.21.4 IMS Access Security for SIP-Based Services

#### 3.21.4.1 Introduction

SIP is at the core of the IMS, as it is used for creating, managing and terminating various types of multimedia sessions. The key thing to accomplish in securing access to the IMS

<sup>12</sup> Cross-certification is a process for establishing trust between two domain authorities. When Domain Authority A is cross-certified with Domain Authority B, both are able to trust each others' certificates –i.e., they are able to authenticate one another.

is to protect the SIP signalling in the IMS. As noted previously, in the IMS core network this is accomplished using the NDS/IP. But the first hop, meaning the interface for SIP communications between the UE and the IMS P-CSCF denoted as Gm, needs additional measures since it is outside the scope of the NDS/IP.

The security features and mechanisms for secure access to the IMS are specified in [3GPP TS 33.203]. This defines how the UE and network are authenticated as well as how they agree on the security mechanisms, algorithms and keys used.

### 3.21.4.2 Trust Model Overview

The IMS establishes a trust domain, as described in [RFC3325], that encompasses the following IMS elements:

- P/I/S-CSCF;
- BGCF;
- MGCF/MRFC;
- all ASs that are not in third-party control.

The main component of trust is identity: in order to trust an entity accessing the IMS there needs to be an established relationship with that entity (i.e., its identity is known and verified). In the IMS this identity is passed between nodes in the trust domain in the form of an asserted identity. The UE can state a preference for this identity if multiple identities exist; but, it is ultimately at the border of the trust domain (namely, in the P-CSCF) that the asserted identity is assigned. Conversely, the P-CSCF plays a central role in authenticating the UE.

The level of trust is always related to the expected behaviour of an entity. For example, Alice may know Bob and trust him to take her children to school. She expects and knows that Bob will act responsibly, drive safely and so on. But she may not trust Bob enough to give him access to her bank account.

Another important property of the IMS trust model is that it is based on transitive trust. The existence of pairwise trust between a first and a second entity as well as a second and a third entity automatically instils trust between the first and the third entity: for example, Alice knows and trusts Bob, who in turn knows Celia and trusts her to take his children to school. Now, according to transitive trust, Alice can also trust Celia to take her children to school without ever actually having met Celia in person. It is enough that Alice trusts Bob and knows that Celia is also part of the trust domain of parenthood. The fact that Alice and Bob are both parents assures Alice that Bob has applied due diligence when judging whether Celia is fit to take children to school. In essence, the trust domain of parenthood forms a network of parents, all compliant with the pre-defined behaviour of a mother or a father.

In [RFC3325] terms, both the expected behaviour of an entity in a trust domain and the assurance of compliance to the expected behaviour needs to be specified for a given trust domain T in what is called a ‘Spec(T)’. The components that make up a Spec(T) are:

- Definition of the way in which users entering the trust domain are authenticated and definition of the used security mechanisms that secure the communications between

the users and the trust domain. In the IMS this entails authentication using the AKA protocol and related specifications on Gm security in [3GPP TS 33.203].

- Definition of mechanisms used for securing the communications between nodes in a trust domain. In the IMS this bit is documented in the NDS/IP [3GPP TS 33.210].
- Definition of the procedures used in determining the set of entities that are part of the trust domain. In the IMS this set of entities is basically represented by the set of peer SEGs, of which a SEG in a security domain is aware.
- Assertion that nodes in a trust domain are both compliant with SIP and SIP-asserted identity specifications.
- Definition of privacy handling. This definition relies on SIP privacy mechanisms and the way they are used with asserted identities (Section 3.19.4.3 will discuss these issues more deeply).

### 3.21.4.3 User Privacy Handling

The concepts of trust domain and asserted identity enable passing a user's asserted identity around, potentially to entities that are not part of the trust domain. This creates obvious privacy issues, since the user may, in fact, require that their identity be kept private and internal to the trust domain.

In the IMS the user can request that their identity is not revealed to entities outside the trust domain. This is based on SIP privacy extensions [RFC3323]. A UE inserts its privacy preferences in a privacy header field, which is then inspected by the network. Possible values for this header are:

- User – indicates that user-level<sup>13</sup> privacy functions should be provided by the network. This value is usually set by intermediaries rather than user agents.
- Header – indicates that the UA is requiring that header privacy be applied to the message. This means that all privacy-sensitive headers be obscured and that no other sensitive header be added.
- Session – indicates that the UA is requiring that privacy-sensitive data be obscured for the session (i.e., in the SDP payload of the message).
- Critical – indicates that the requested privacy mechanisms are critical. If any of those mechanisms is unavailable, the request should fail.
- ID – indicates that the user requires their asserted identity be kept inside the trust domain. In practice, setting this value means that the P-Asserted-Identity header field must be stripped from messages that leave the trust domain.
- None – indicates that the UA explicitly requires no privacy mechanisms to be applied to the request.

### 3.21.4.4 Authentication and Security Agreement

Authentication for IMS access is based on the AKA protocol. However, the AKA protocol cannot be run directly over IP; instead, it needs a vehicle to carry protocol messages

---

<sup>13</sup> By 'user-level privacy' we mean privacy functions that the SIP UA itself is able to provide (e.g., using an anonymous identity in the From field of a request).

between the UE and the home network. Obviously, as the entire objective of IMS access authentication is to authenticate for SIP access, SIP is a natural choice for such a vehicle. In practice, the way in which the AKA protocol is tunnelled inside SIP is specified in [RFC3310]. This defines the message format and procedures for using AKA as a digest authentication [RFC2617] password system for the SIP registration procedure. The digest challenge originating from the network will contain the RAND and AUTN AKA parameters, encoded in the server nonce value. The challenge contains a special algorithm directive that instructs the client to use the AKA protocol for that particular challenge. The RES is used as the password when calculating digest credentials, which means that the digest framework is utilized in a special way to tunnel the AKA protocol in IMS access security.

Concurrent with authentication of the user, the UE and the IMS also need to negotiate the security mechanisms that are going to be used in securing subsequent SIP traffic in the Gm interface. The protocol used for this security agreement is again SIP, as specified in [RFC3329]. The UE and the P-CSCF exchange their respective lists of supported security mechanisms and the highest commonly supported one is selected and used. At a minimum, the selected security mechanism needs to provide data integrity protection, as that is required to protect actual security mechanism negotiation. Once the security mechanism has been selected and its use started, the previously exchanged list is replayed back to the network in a secure fashion. This enables the network to verify that security mechanism selection was correct and that the security agreement was not tampered with. An example of an attack that would be possible without this feature is a ‘bidding-down attack’, where an attacker forces peers into selecting a known weak security mechanism. The important benefit from having secure negotiation of the used security mechanism is that new mechanisms can be later added and old ones removed. The mechanisms can coexist nicely as each UE always uses the strongest mechanism it has available to it.

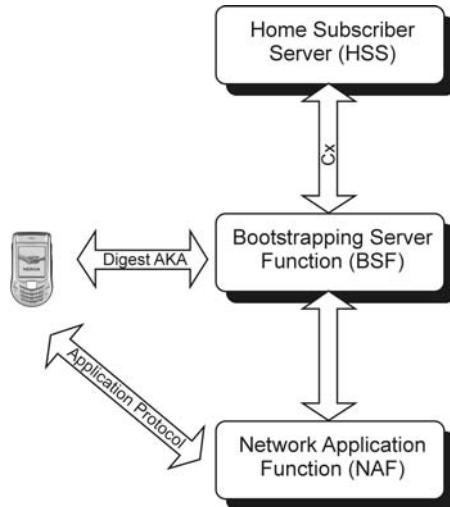
### **3.21.4.5 Confidentiality and Integrity Protection**

In IMS access security, both confidentiality as well as data integrity and authentication are mandatory. The protocol used to provide them is IPsec ESP [RFC2406], explained in further detail in Chapter 22.

AKA session keys are used as keys for ESP SAs. The IK is used as the authentication key and the CK as the encryption key. Naturally, depending on the key lengths required by the cipher algorithms used in ESP, certain key expansion functions may be used on AKA session keys.

### **3.21.4.6 Key Management and Distribution**

As described in previous chapters, the P-CSCF may also reside in the visited network. By virtue of the AKA protocol, the shared secret is only accessible in the home network, which means that, while authentication needs to take place in the home network, a certain delegation of responsibility needs to be assigned to the P-CSCF, as IPsec SAs exist between the P-CSCF and the UE. In practice, while IMS authentication takes place in the home network, the session keys that are produced in AKA authentication and used in ESP are delivered to the P-CSCF piggybacked on top of SIP registration messages.



**Figure 3.45** Generic bootstrapping architecture

To renew the SAs the network has to re-authenticate the UE. This means that the UE has to re-register as well, which may be either network initiated or due to the registration expiring. The net effect is the same: the AKA protocol is run and fresh keys are delivered to the P-CSCF.

### 3.21.5 IMS Access Security for HTTP-Based Services

#### 3.21.5.1 Introduction

Parallel to SIP traffic, there is a need for the UE to manage data associated with certain IMS applications. The Ut interface, as explained in Section 2.3.15, hosts the protocols needed for that functionality. Securing the Ut interface involves confidentiality and data integrity protection of HTTP-based traffic [RFC2616]. As previously mentioned, user authentication and key establishment for the Ut interface are also based on AKA.

#### 3.21.5.2 The Generic Bootstrapping Architecture (GBA)

As part of the GAA, the 3GPP IMS defines the Generic Bootstrapping Architecture (GBA) [3GPP TS 33.220], illustrated in Figure 3.45. The Bootstrapping Server Function (BSF) and the UE perform mutual authentication based on AKA, allowing the UE to bootstrap session keys from the 3G infrastructure. Session keys are the result of AKA and enable further applications provided by a Network Application Function (NAF). One such example is a NAF that issues subscriber certificates<sup>14</sup> using an application protocol secured by the bootstrapped session keys.

<sup>14</sup> Such an entity is usually referred to as a PKI portal.

### 3.21.5.3 Authentication and Key Management

Authentication in the Ut interface is performed by a specialized element, called the ‘authentication proxy’. In terms of the GBA the authentication proxy is another type of NAF. Traffic in the Ut interface goes through the authentication proxy and is secured using the bootstrapped session key.

### 3.21.5.4 Confidentiality and Integrity Protection

The Ut interface employs Transport Layer Security (TLS) for both confidentiality and integrity protection [3GPP TS 33.222]. TLS is discussed more thoroughly in Chapter 21.

## 3.22 Interworking between IPv4 and IPv6 in the IMS

### 3.22.1 Introduction

In the Internet at large, deployment of IPv6 has not progressed as fast as was hoped or expected when the initial concept of the IMS first came about. Many factors have contributed and continue to contribute to this, but in general any change in the core routing infrastructure of the Internet is likely to take a much longer time than one adopted at the edges of the network.

Network Address Translators (NATs) constitute a change that falls into the latter category. NATs are widely acknowledged as detrimental to new services as well as existing ones (e.g., SIP-based VoIP), but in terms of investment they offer a very affordable and easily deployable option to extending the fast-depleting address space of IPv4,<sup>15</sup> thus slowing down deployment of the long-term solution in the form of IPv6.

In many ways, what is happening across the Internet is descriptive of what is happening in the mobile domain. While IPv6 is seen as technically superior to IPv4 for the IMS, mandating IPv6 support represents quite a big barrier to deployment of any system. Using IPv6 exclusively also requires that roaming partners and service providers migrate to IPv6. This is why many early deployments have chosen to work with existing GPRS access networks, as well as a security infrastructure (more on this in Section 3.21), and often this existing network infrastructure does not support IPv6. In addition, the 3GPP2 IP multimedia system, called Multimedia Domain (MMD), operates both over IPv4 and IPv6. This means that any interworking between the IMS and the MMD naturally involves IPv4–IPv6 interworking as well.

Hence, it has become apparent that some deployment of the IMS based on IPv4 is inevitable, although originally specified as supporting IPv6 only. Usually, these IMS deployments based on IPv4 are called ‘early IMS implementations’, and either the UE, the IMS core network, or both exclusively use IPv4.

There are two main consequences:

- IPv4 implementations will need to deal with having to use private address space, since not all mobile devices can be afforded an address from the public address space.

---

<sup>15</sup> Actually, there are many other use-cases for deploying NATs, which have to do with security (firewall functionality), naming (e.g., private naming of hosts in a home-grown network), as well as addressing, just to name a few.

- Once IPv6-capable implementations are introduced alongside IPv4 implementations, they need the ability to work together in a seamless way.

This chapter illustrates the issues and solutions for IPv4 and IPv6 interworking in IMS. In Release 6, 3GPP has worked to provide guidelines to the implementers of early IMS systems in the form of a technical report [TR 23.981]. This means that, while early IMS implementation is still not endorsed, at least some form of guidelines is given to address some of the most pressing concerns.

### 3.22.2 *Network Address Translation*

#### 3.22.2.1 General

Network address translation for SIP and especially within IMS is a highly complex issue that could fill a book on its own. There is a large number of possible scenarios that can occur and also very specific solutions to each of them.

This section describes the principles of network address translation within IMS and gives an overview of the involved mechanisms. Nevertheless, this is only meant as an overview and a first introduction and does not cover all possible scenarios and protocol details.

#### 3.22.2.2 Network Address Translator (NAT)

The basic component for doing address translation is called a Network Address Translator (NAT). NATs allow a single public IP address to be shared between a number of hosts. The hosts behind the NAT device are given IP addresses that are in a private address space (i.e., in 192.168.0.0/16 or 10.0.0.0/24), and therefore not routable in the Internet. The NAT device then creates temporary bindings between the public and private address space, on a per-connection basis. A binding is simply a mapping between the public IP address and port to a private address and port associated with a specific transport (UDP or TCP). The binding lasts only as long as the session lasts, or until the binding times out. Usually, these timeouts are quite aggressive in order to save resources.

Another type of NAT is specifically designed to translate between protocol versions. A Network Address Translator–Protocol Translator (NAT-PT) translates between IPv6 and IPv4. In effect, it takes the IP datagram and replaces the IPv6 header with an IPv4 header and vice versa.

#### 3.22.2.3 Network Address Translation for SIP and IMS

For SIP and IMS, the existence of NATs alone is not enough. SIP and SDP both contain plain IP addresses, which are related to each other and therefore need to be translated in a consistent way.

For example the IP address in the SIP Contact header and the SDP connection address (c-line) are identical in most of the IMS scenarios, as the UE is sending SIP from the same IP address as it wants to receive the media at. But within IMS, SIP signalling and the media session are not traversing the same network nodes, they are transported end-to-end independently of each other.

A normal NAT is not aware of such complex relations between different protocols and therefore does not take these relations into account when performing the address translation. Most of the NATs do not even work on application protocol level, such as SIP, SDP and RTP.

There are basically two ways to get around this problem and allow network address translation for SIP and IMS:

- include an intelligent NAT, that is aware of the application protocols (SIP, SDP, RTP, etc.) and their relation amongst each other – such an entity is called Application Layer Gateway (ALG);
- enable the UE to find out whether it is behind a NAT, to get aware of its own public IP addresses as well as of the public IP addresses of the remote side. Based on this knowledge, the UE can modify SIP and SDP messages in order to make the traversal of signalling and media through NATs possible.

### 3.22.2.4 Application Layer Gateway

Within IMS an ALG can be co-located with the P-CSCF and controls a NAT that is located in the IMS Access Gateway (see Figure 3.46). This is a so called hosted solution, as the network operator hosts the NAT and has made the IMS entities in the network aware of the NAT. When an ALG is used, the UE stays unaware of ongoing network address translation.

The ALG keeps the binding between the UEs private and public address as long as it is necessary for proper and seamless IMS operations.

If we assume that a SIP INVITE request is sent by the UEs private address, the ALG assigns a public address and binds it to the session (it is in fact more likely that the

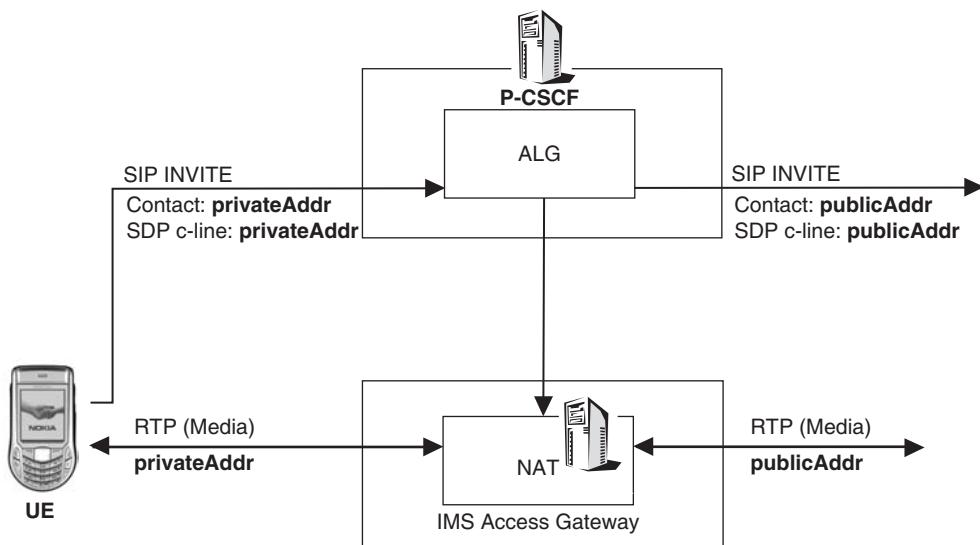


Figure 3.46 Application layer gateway in IMS

binding stays for the time of the UEs registration, but this is left out of scope here). Within the INVITE request sent from the P-CSCF towards the terminating side, the UEs public address will be included on all protocol layers, e.g.:

- within the IP packet that transports the SIP INVITE request;
- within the SIP routing related information, such as e.g. the Contact header and Via headers;
- within the SDP message, that is conveyed in the body of the SIP INVITE request, e.g. in the c-line which indicates the connection address.

The ALG informs the NAT in the IMS access gateway (GGSN) about the newly created binding. Once media is flowing between the two UEs, the NAT in the access gateway now will translate the IP addresses indicated in the RTP packets to and from the users' public and private addresses.

As may be expected, ALGs have several drawbacks:

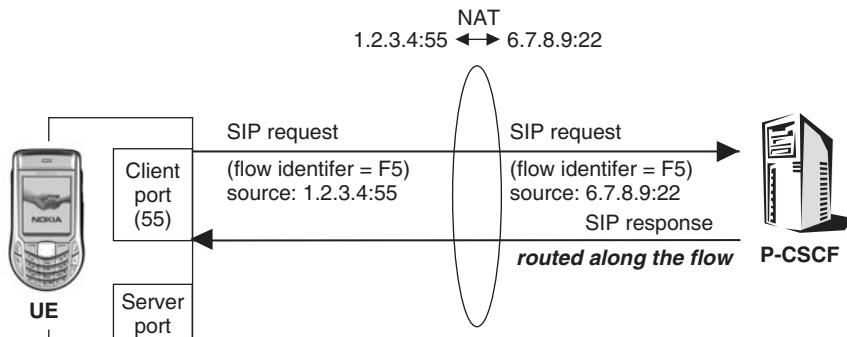
- They are a single point of failure; if the translation binding is lost, the entire session becomes unusable.
- ALGs either brake security, or do not operate correctly in the presence of an end-to-end security mechanism. Modification of messages in transit is, in effect, an active Man-in-the-Middle (MITM) attack.
- ALGs are not feature transparent; a new version of SDP, or another SIP header field containing an IP address will not be recognized or translated correctly.
- Scalability problems; ALG functionality is resource intensive, and NATs require both resources for storing bindings as well as bandwidth.

### 3.22.2.5 UE taking Care of NAT Traversal

#### 3.22.2.5.1 Overview

If the UE needs to take care of NAT traversal using the correct IP addresses within the SIP, SDP and RTP messages, we look at a far more complex scenario compared to using a ALG. The UE needs to make use of the following extensions:

- SIP Outbound mechanism, as defined in draft-ietf-sip-outbound, to make sure that SIP messages sent towards the UE traverse a NAT;
- Session Traversal Utilities for NAT (STUN), as defined in [draft-ietf-behave-rfc3489bis], which allows a UE to get aware of its public address when it is located behind a NAT;
- Traversal Using Relays around NAT (TURN), as defined in [draft-ietf-behave-turn], which provides relay functionality within the network, so that e.g. media can traverse via a TURN relay;
- Interactive Connectivity Establishment (ICE) Framework, as defined in [draft-ietf-mmusic-ice], which makes use of STUN and TURN and allows the exchange of available public and private addresses between UEs in order to make SIP and media connections work in NAT scenarios;
- in addition needs to take care, to keep the NAT bindings alive on the media and on the signalling level.



**Figure 3.47** Routing based on SIP Outbound flows

### 3.22.2.5.2 SIP Outbound

SIP Outbound enables the routing of SIP requests and responses towards a UE, when a NAT is located between the UE and the P-CSCF.

If a NAT is located between a UE and the P-CSCF, the NAT will assign a public IP address to the UE when it is sending the first REGISTER request towards the P-CSCF (see Figure 3.47). This binding assigned by the NAT will be valid for the IP address and port number (the client port – see Section 11.7.5.1) indicated from the UE in the private address.

Nevertheless, responses to SIP requests as well as SIP requests to the UE are expected to be received at the UE on the protected server port, which is different from the client port indicated in the REGISTER request. The NAT, on the other hand, did not create a binding for the protected server port and therefore cannot deliver the message to the UE.

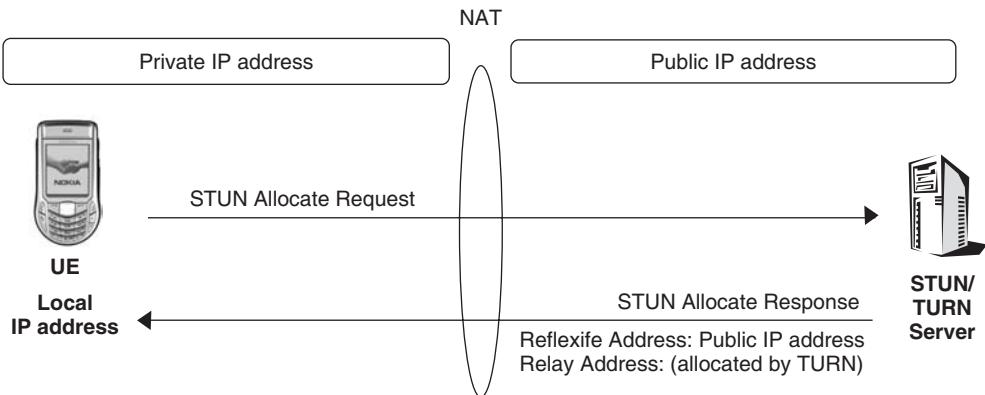
SIP Outbound allows such a transport, by creating a logical flow between the UE and the P-CSCF, which is identified by a flow identifier. Once Outbound is used, messages towards the UE will not be routed according to the normal SIP routing procedures (i.e. to the server port) but along the flow, which means it will be sent to the port from which the request originated, i.e. the client port.

### 3.22.2.5.3 Interactive Connectivity Establishment

In order to get aware whether the UE is located behind a NAT, the UE starts sending STUN messages to a STUN server. STUN servers need to be deployed within the network, in order to make NAT traversal with IMS work. The UE discovers the STUN server via specific DNS SRV procedures (for more information on DNS SRV see Section 11.4.4).

The STUN server responds to the UE, indicating in the response the public addresses that the UE got assigned by the NAT whilst the request was routed to the STUN Server. In this way the UE gets aware of the public IP address that it got assigned to it by the NAT, which is called the reflexive address.

As the NAT between the UE and the STUN server is not aware of the type of communication that the UE using, it usually assigns the binding between public and private address only for a limited amount of time. This means that after this time expires, the NAT might assign a different public address to a request originated from the UE and might also not send incoming requests to the public IP address towards the UE. Therefore the



**Figure 3.48** UE discovers reflexive and relayed addresses via STUN/TURN

UE needs to make sure that it keeps the assigned binding in place by sending keep-alive messages towards the STUN server (see Figure 3.48).

Note that for the communication of the UE with the outside world (e.g. other UEs) the STUN server does not need to be contacted further, once the UE has learned its public IP address and keeps it alive. This means that SIP, SDP and RTP messages will not traverse the STUN server.

To make the scenario even more complex, the UE might reside behind different NATs (even cascaded NATs) and might have a choice between different reflexive addresses.

In addition to this, a STUN server can act as a TURN server. TURN servers are relays for mainly media but also signalling traffic to and from a UE. They assign so-called relayed addresses, so that the UE can make use of the relay. This means that at least RTP messages to and from the UE can traverse through the TURN server, if the UE is making use of the relay address. The TURN server is a reliable connection point that is needed in some connection scenarios, based on the network topology.

In order to use the TURN relay, the UE must send STUN Allocate Requests to the STUN server. The STUN server then replies with the reflexive address and the relayed address that belong to the user.

The UE therefore can have three types of addresses:

- its local address – also called private address;
- one or more reflexive addresses, assigned by NATs;
- one or more relayed addresses, assigned by STUN/TURN servers.

These addresses are all called candidate addresses, as they are possible candidates for end-to-end connections with other UEs.

With ICE, both UEs can exchange their candidate addresses within one SDP offer/answer exchange in a prioritized order. Once they have exchanged their candidate addresses, both sides will perform connectivity checks in order to find out how to reach the other side. Once a connection has been established, they exchange another SDP offer/answer and start using the new connections (see Figure 3.49).

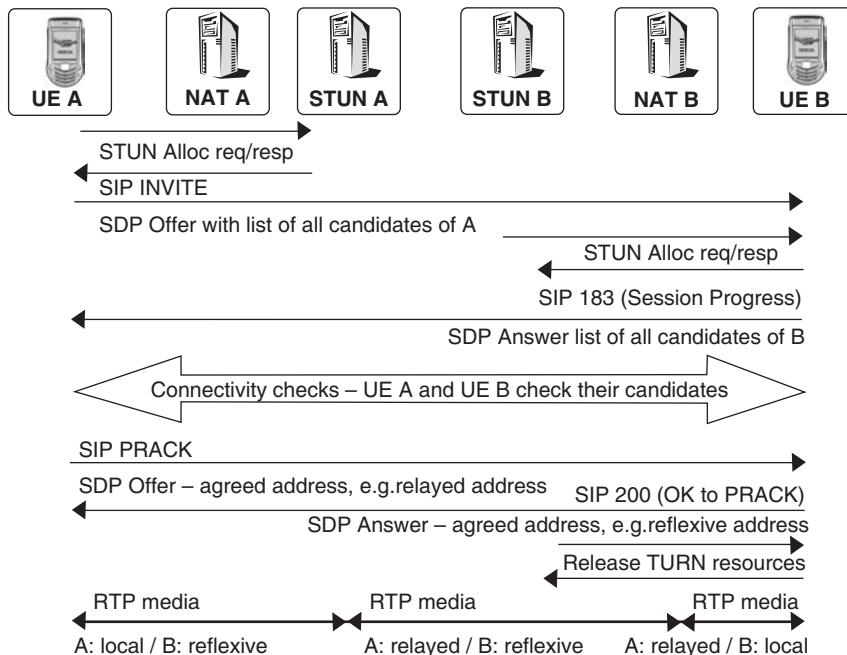


Figure 3.49 Simplified STUN/TURN/ICE flow

### 3.22.3 IPv6-Only Versus Dual Stack

One key method for successful transition to IPv6 is for hosts (and routers) to support a dual IP layer approach, also known as a ‘dual stack operation’. Dual stack is a technique for providing support for both versions of IP – i.e., IPv6 as well as IPv4. A dual stack host can send and receive both versions of IP and can therefore communicate with both kinds of nodes. A dual stack node is also able to be configured with both address versions, although the configuration mechanisms may be different. For example, the IPv4 address could be configured using DHCP and the IPv6 address using stateless address autoconfiguration.

There are also other functions that a dual stack node needs to support. For example, it needs to support both IPv4 ‘A’ and IPv6 ‘AAAA’ DNS records, irrespective of over which version of IP datagrams such results are received. For example, a DNS query made over IPv4 for a domain might return both A and AAAA records, or even AAAA records only. Then, the order in which the IP addresses are used depends on the application, but may also be influenced by the order in which the DNS server returns the respective records.

A node may also disable either protocol version from its stack. However, in practice, many IMS nodes will need to keep IPv4 functioning, as there are other applications – such as the WWW – that require IPv4 to be fully usable.

In terms of migration, dual stack UEs and IMS core offer more options than IPv4-only implementations. Even with a specific IP version disabled in the stack, later on when IPv6 becomes more prevalent, it can be enabled. The main benefit in dual stack hosts is that they minimize the need for NAT devices in the network. IPv6-only hosts don’t appear to

be feasible well into the future, since the majority of services will still remain IPv4-only for a long time.

#### 3.22.4 Interworking Scenarios

The main problem with IPv4–IPv6 interworking is, of course, the different IP version. The existence of nodes that only understand one version make it necessary for some form of translation. This applies to both signalling traffic as well as user-plane or media traffic. There are also many ways that the translation can happen, depending on whether the IMS core supports IPv6.

We split the discussion in this section into intra-domain and inter-domain scenarios. Intra-domain scenarios deal with interworking between a UE and the IMS core of a single IMS provider, whereas the inter-domain scenarios deal with the interconnection of different IMS providers' networks, and the end-to-end considerations between UEs belonging to different IMS operators' networks.

#### 3.22.5 Intra-Domain Scenarios

The intra-domain scenarios are very straightforward; the assumption is that between the UE and the P-CSCF, the IP version stays constant:

- if the IMS core supports IPv4 only – only dual stack and IPv4 clients are supported;
- if the IMS core supports dual stack – all variants of clients are supported;
- if the IMS core is IPv6 only – only dual stack and IPv6 clients are supported.

#### 3.22.6 Inter-Domain Scenarios

The inter-domain scenarios are much more complicated. The complexity is brought on by the fact that there are many more variables to consider. In addition to the sender and receiver UE and IMS core, there is also the transit network between the two domains, which can also exhibit support for one of either IPv4, dual stack or IPv6 behaviour. In addition, the IMS core may also be using IPv4 addresses from a private address space, requiring a NAT at the edge of the network in order to interconnect with other domains.

A dual-stack IMS core may need an additional NAT device that is associated with the S-CSCF. The reason for this is that the interconnection may use IPv4, which the I-CSCF and the S-CSCF support, but the UE doesn't. In that case, a NAT-PT device is needed to translate the IP versions. This is illustrated in Figure 3.50, showing the possible placement of the various NAT devices.

#### 3.22.7 Configuration and Bootstrapping

Usually, after establishing L2 connectivity the UE receives an IP address using a standard configuration, such as DHCP, or some other mechanism.

The next step for the UE is to find its contact point to the IMS – namely, the address of the P-CSCF. In IMS, the existing discovery mechanisms for the P-CSCF address are either IPv6 specific, or use a Release 5 (or later) based GPRS. For deployments of IMS based on IPv4 other mechanisms may be needed. There are a few alternative options to discovering the P-CSCF address:

- Discovery using the GPRS could be altered so that it returns both IPv4 and IPv6 addresses upon PDP context establishment.
- Using the DHCP option for configuring the P-CSCF address.
- Out-of-band provisioning mechanisms, such as one using the Short Messaging Service (SMS), Over-the-Air (OTA) or Open Mobile Alliance (OMA) device management framework.
- Pre-configuration.

Using an out-of-band mechanism requires no changes to the existing network infrastructure and is, therefore, probably preferred by most early IMS implementers. Pre-configuring the address is also an option, but it is inflexible regarding later changes in naming or topology.

In addition to configuring the outbound proxy (i.e., the P-CSCF) address, there is additional address information carried in the SIP header, which needs interworking considerations as well. For example, any route entries or addresses of charging entities need to be represented in both IPv4 and IPv6.

### 3.22.8 IPv4-Only Access Networks

Even though not strictly an IMS implementation issue, due to functionality in GPRS, the access network is a consideration in IPv4–IPv6 interworking. The reason for this is that the bearers are dependent on the IP version; an SGSN/GGSN incapable of supporting IPv6 would not allow an IPv6 PDP context to be established. Older, existing access networks might simply not support IPv6. Therefore, the first interworking issue is with the GPRS access network, both in roaming and nonroaming scenarios. If the GPRS SGSN or the

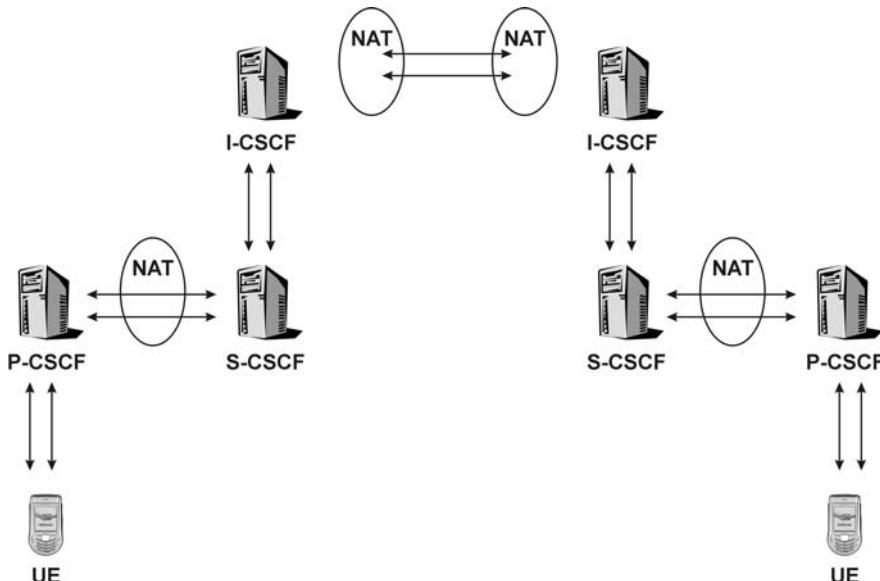
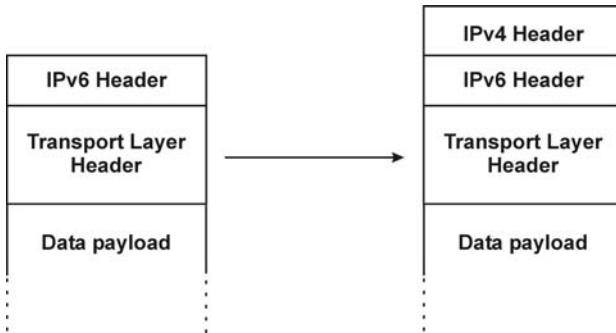


Figure 3.50 End-to-end and interconnection scenarios



**Figure 3.51** IPv6 to IPv4 tunnelling mechanism

GGSN do not support IPv6, the UE is unable to use native IPv6, regardless of whether any of the other IMS elements are supporting it. The only available option for using IPv6 in these scenarios is for the UE and IMS core to use a tunnelling mechanism, where each IPv6 packet is encapsulated in an IPv4 packet, and de-capsulated at an exit point or router – in effect, treating the IPv4 connectivity as L2 transport. This is illustrated in Figure 3.51.

Encapsulation needs to be applied between the host and a network-based node in order to be able to transparently communicate with a IPv6-only host. So, in effect, the tunnel end point will route the packets to the correct host in the network, after de-capsulation of the packet. Tunnels are addressed and set up using a tunnelling mechanism, such as the Intra-Site Automatic Tunnel Addressing Protocol (ISATAP) [Draft.ietf-ngtrans-isatap].

In ISATAP, interface addresses are created using a static mapping from the IPv4 link address (i.e., the last 32 bits of the interface address). They are accompanied by a routing table, so that packets can then be routed to the proper interface. However, there are several downsides to the tunnelling approach:

- the home network needs to be installed with a tunnelling server that serves as a tunnel end point;
- such a server also needs to be configured or discoverable by the UE;
- the tunnel end point is a single point of failure;
- using a tunnel in roaming scenarios is likely to be inefficient, as all traffic is routed via the home network – this causes triangular routing which will increase latency;
- it requires a dual stack host in any case, so using native IPv4 may be a better approach.



# Part II

# IMS Services



# 4

# Presence

Presence and instant messaging are changing the personal and corporate communications paradigm. Presence will enhance messaging as well as introduce a new service (the presence service itself) that can be used in many other applications and services. Presence will be the heart of all communications and the new way for telephony to function. Presence will also be a lucrative business opportunity for both operators and service providers.

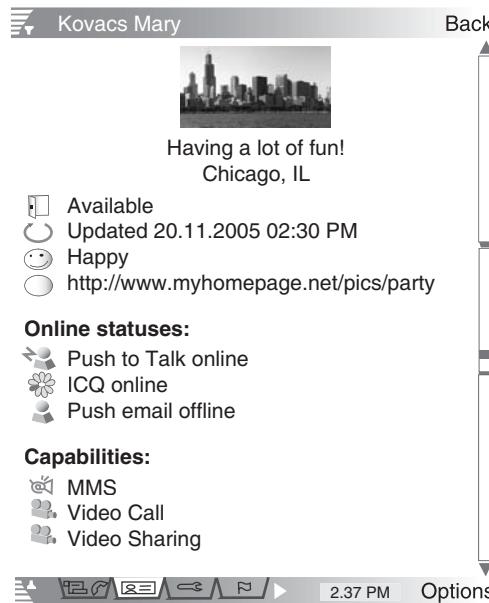
Presence is a dynamic profile of the user, which is visible to others and used to represent oneself, share information and control services. Presence can be seen as a user's status as perceived by others and others' status as perceived by the user. Status may contain information such as personal and device status, location or context, terminal capabilities, preferred contact method as well as services the user is willing to use to communicate with others, including voice, video, instant messaging as well as gaming. Example is depicted in Figure 4.1.

Presence information is also personal: it is always linked to a particular person. It shows the person initiating the communication whether the other person is available and willing to communicate. On the other hand, presence information can be used to communicate to others when a person is able and willing to communicate as well as with whom and by what means. This will allow users to control their own communication more effectively.

Presence information sharing raises security and privacy concerns. Using Session Initiation Protocol (SIP) for presence, users can control their own specific presence information and have the final say on how it is used including who can and cannot see certain parts, if not all, of the presence information.

## 4.1 Who will use the Presence Service?

There will be many different user groups, which will use presence information for different purposes. These groups will range from corporate business users to teenagers and children. While presence is likely to be used more for rational availability management in corporate usage, young consumers are constantly seeking new ways of expressing themselves and building an identity in a fun and visually rich way. Successful presence services will be adaptable to the different needs of varied user groups and segments.



**Figure 4.1** Dynamic presence

## 4.2 Presence-Enhanced Services

The basic presence service as we know it today works with the basic idea of ‘publish–receive’ of presence information about humans. Operators have the ability to fetch various infrastructure-related presence attributes, such as location and terminal availability from their communication networks. This enhances the suite of presence information presented to subscribers who subscribe to the operator’s presence service and gives the user a greater experience.

New presence-enabled services use presence information in dedicated application areas. This is a big opportunity for innovative companies that want to expand their business. Examples include presence-based call routing as well as network gaming services. In addition, presence creates an alternative advertising and information-sharing channel that the user is comfortable with (see Figure 4.2).

With presence, team working is more efficient, bringing the ability to share information amongst the team members beyond their availability. They can share information such as a meeting location, future plans, etc.

## 4.3 Presence Contributing to Business

Presence can contribute to existing businesses and create a business of its own. There will be basic presence user services as well as new presence-enabled services. Operators and other service providers have a major role in mass adoption of presence services. The basic mobile presence service can be part of the operator’s service portfolio, since other services – i.e., the new presence services – can use the basic service. The mobile



**Figure 4.2** Examples of enhanced presence service

domain reaching more than a billion subscribers worldwide, is a profitable platform for new consumer services.

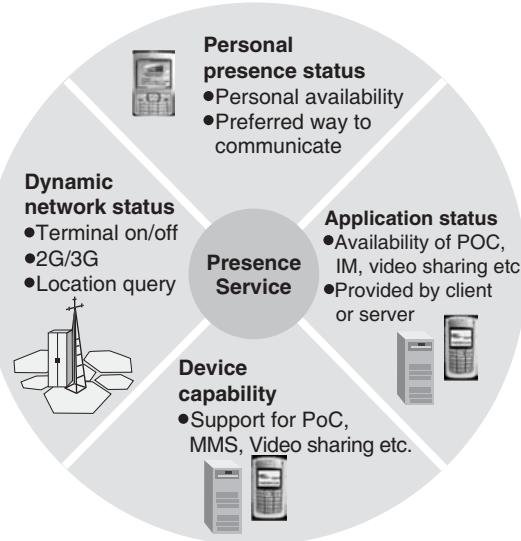
Offering a basic presence service can give a competitive advantage for an operator over other operators who are not offering it: by binding their presence information to a particular operator's services, customers have high-value services that other operators may not be able to offer without that information. Presence generates new traffic for existing services like instant messaging. Presence also minimizes incomplete calls or calls being rejected due to the called party being busy.

Operators also need to carefully consider the pricing of presence allowing people to make an easy decision about adopting the presence service, without having to think for too long about the cost/benefit.

#### 4.4 What is Presence?

Presence is in essence two things: it involves making my status available to others and the statuses of others available to me. Presence information may include (see also Figure 4.3):

- person and terminal availability;
- communication preferences;



**Figure 4.3** Overview of presence

- terminal capabilities;
- current activity;
- location;
- currently available services.

It is envisioned that presence will facilitate all mobile communication, not only instant messaging, which has been the main driver for presence. Instant messaging has been the main interactive, almost real-time communication service in the Internet and presence is a great compliment in that you know if a friend is online before you begin a chat session with her. However, in the mobile environment, it is envisioned that presence information will not only support instant messaging, it will also be used as an indicator of the ability to engage in any session, including voice calls, video and gaming: all mobile communications will be presence-based.

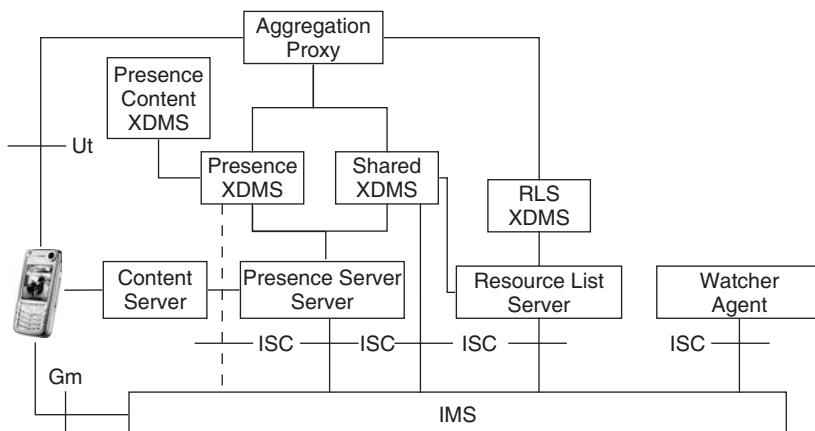
Presence-specific and presence-enhanced applications and services will be available in the near future. A typical example of a presence-specific application will be a phone book with embedded presence information, making it dynamic. Dynamic presence (Figure 4.1) will be the initial information the user sees before establishing communication. This information affects the choice of communication method and timing.

## 4.5 Presence Service in IMS

Presence service was originally introduced in 3GPP Release 6 as a standalone service capability, but later on OMA took over IMS based presence service and currently the most comprehensive Presence solution is defined in OMA. At the time of writing OMA is finalizing the second Presence service enabler release. In OMA based

presence architecture the following building blocks can be identified [OMA Presence Architecture]:

- Presence Server – an IMS application server that manages presence information uploaded by presence sources and handles presence subscription requests.
- Resource List Server – an IMS application server that accepts and manages subscriptions to presence lists, which enables a watcher application to subscribe to the presence information of multiple presentities using a single subscription transaction.
- XML Document Management Servers – application servers that store presence service related data. Four different ASes are defined: Presence XDMS (a server that contains rules for presence information subscriptions and rules for presence information publication), RLS XDMS (a server that contains user's presence buddylist), Presence Content XDMS (a server that manages media files for the Presence Service) and Shared XDMS (a server which can be reused by multiple different application servers) see Section 5.7 for additional details.
- Content server – functional entity that is capable of managing MIME objects for Presence, allowing the presence sources or the Presence Server to store MIME objects.
- Presence source – an entity that provides presence information to a presence service. The presence source can be located in a user's UE or within a network entity.
- Presence watcher – the entity that requests presence information about resources (presentities).
- Watcher agent – is an entity that controls the Watcher's Presence Service use in the Watcher domain.
- Watcher Information Subscriber – is an entity that requests Watcher Information about a presentity from the Presence Service.



**Figure 4.4** Presence architecture

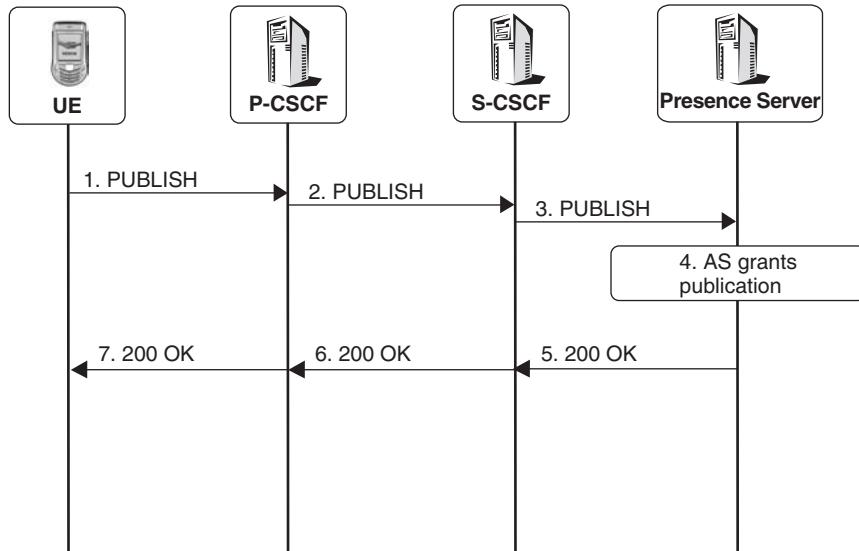
Figure 4.4 shows a reference architecture where only device based presence source and watcher is shown.

## 4.6 Publishing Presence

To publish or update presence information the Presence Source uploads presence information using the SIP PUBLISH method to the Presence Server (see Figure 4.5). The request could like this

```
PUBLISH sip:presence@example.com SIP/2.0
To: <sip:presence@example.com>
From: <sip:presence@example.com>;tag=1234wxyz
Expires: 3600
Event: presence
Content-Type: application/pidf+xml
...
<?xml version="1.0" encoding="UTF-8"?>
<presence xmlns="urn:ietf:params:xml:ns:pidf"
    xmlns:op="urn:oma:xml:prs:pidf:oma-pres"
    entity="sip:presence@example.com@example.com">
    <tuple id="a1232">
        <status>
            <basic>open</basic>
        </status>
        <op:willingness>
            <op:basic>open</op:basic>
        </op:willingness>
        <op:registration-state>active</op:registration-state>
        <op:service-description>
            <op:service-id>org.openmobilealliance:
                PoC-session</op:service-id>
            <op:version>1.0</op:version>
        </op:service-description>
        <contact>sip:presence@example.com @example.com</contact>
        <timestramp>2008-05-26T12:00:00Z</timestramp>
    </tuple>
</presence>
```

The Request-URI is used by the publisher to identify the person whose presence state is being published. The Event header is used by the publisher to identity that this request is related presence and it is used by the S-CSCF to route this request to the Presence Server. The Expires header indicates how long this presence state is valid. The body of a PUBLISH request contains the actual presence information in XML encoded. The XML-body here reveals that the user is registered, available, willing to communicate, available service is Push to talk Over Cellular and information when the information was published.



**Figure 4.5** Presence publication

The number of different presence state values (also known as presence attributes) have been specified in IETF and OMA that can be used to describe presence state such as application-specific willingness, application-specific availability, communication address, location type, geographical location, time-zone, mood, icon, timestamp, note, session-participation, registration-state, barring-state.

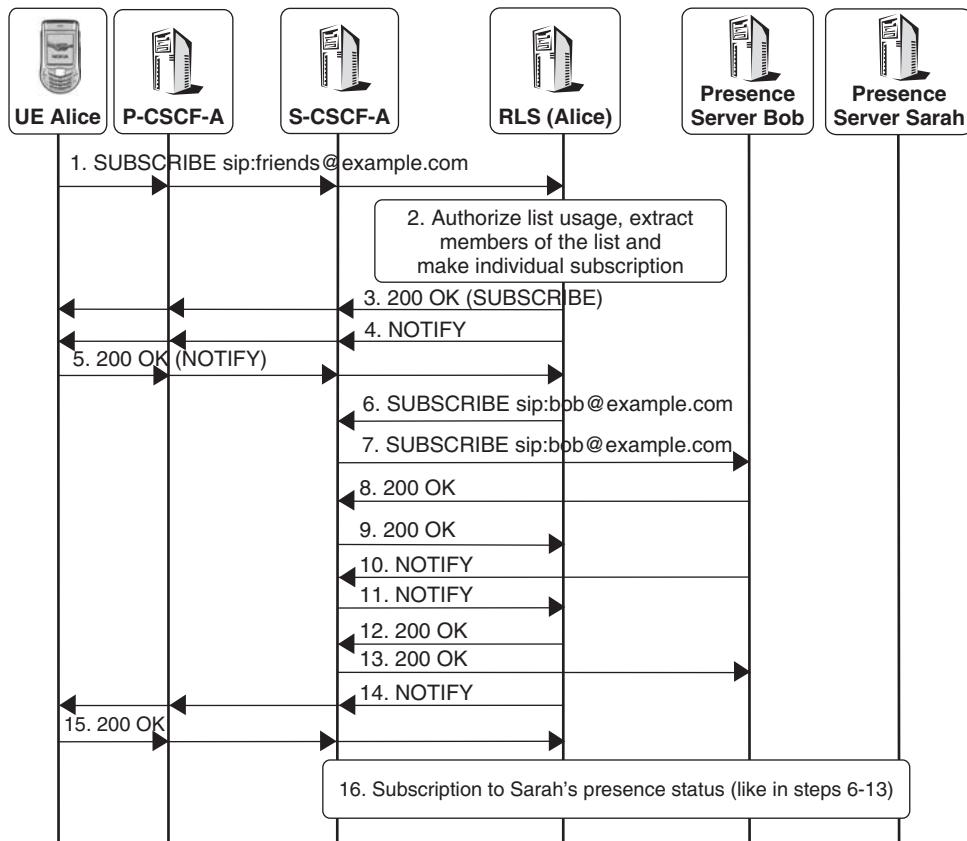
## 4.7 Subscribing Presence

To obtain presence information of other users or presence enhanced service the watcher (Alice) makes subscription to presentity's presence state by sending a SIP SUBSCRIBE request targeted to individual presentity or watcher's own presence list (e.g. `sip:friends@example.com` like created in Section 5.6) containing list of users (Bob and Sarah) which presence information the watcher wants to learn. The request could look like this

```

SUBSCRIBE sip:friends@example.com SIP/2.0
To: <sip:friends@example.com>;tag=30
From: <sip:alice@example.com>;tag=12
Expires: 3600
Event: presence
Content-Length: 0
  
```

As this request is targeted to the presence list it will be routed to the RLS which authorizes Alice's subscription and extracts members of the presence list and makes individual subscriptions to each presentity. The RLS accepts the subscription with 200 OK and according to protocol behaviour it will send immediate NOTIFY request and as the RLS does not hold any presence information about the resources in the list it then sends



**Figure 4.6** Subscription to presence information

NOTIFY with empty body. Once the RLS gets presence information from presence servers it will deliver meaningful NOTIFY request containing presentity's presence state in XML body. Step 14 in Figure 4.6 could look like this (Bob is available and he has added text saying that he is currently in London):

```

NOTIFY sip:alice@example.com SIP/2.0
To:<sip:alice@example.com>;tag=12
From: <sip:friends@example.com>;tag=30
Event: presence
Subscription-State: active;expires=3595
Content-Type: application/pidf+xml
Content-Length: ...
<?xml version="1.0" encoding="UTF-8"?>
<presence xmlns="urn:ietf:params:xml:ns:pidf">
  entity="pres:bob@example.com">
    <tuple id="sg89ae">
      <status>
```

```

<basic>open</basic>
</status>
<note>I'm in London at the moment</note>
</tuple>
</presence>

```

## 4.8 Watcher Information

As described earlier presence information sharing raises security and privacy concerns therefore mechanism to control which users can see users' (presentity) presence information have been defined. User is able to set his/her presence authorization rules at presence server in such a way that for certain users presence subscriptions are allowed, for certain users presence subscriptions are blocked or presence server is expected to ask user's interaction to accept or block new subscription.

To gain knowledge about watchers and the state of their subscriptions a user can subscribe to a watcher information template package [RFC3857]. The information carried to the watcher information subscriber contains two important items: the status of each subscription made by the watchers of the main package and the event that caused the transition from the previous status to the current one. This information is carried in XML body as defined in RFC3858.

The states of the watcher information package are:

- Init – no state is allocated for a subscription.
- Terminated – a policy exists that forbids a watcher from subscribing to the main event package.
- Active – a policy exists that authorizes a watcher to subscribe to the main package.
- Pending – no policy exists for that watcher.
- Waiting – similar to pending, but tells the template package subscriber that a user has attempted to subscribe to the main package and that the subscription expired before a policy was created.

Figure 4.7 depicts an example of signalling flow where user Alice wants to know who is interested in her presence data. Moreover, in this example, user Joe subscribes to Alice's presence data and Alice's presence authorization rules dictate that Alice must be contacted to make explicit decisions.

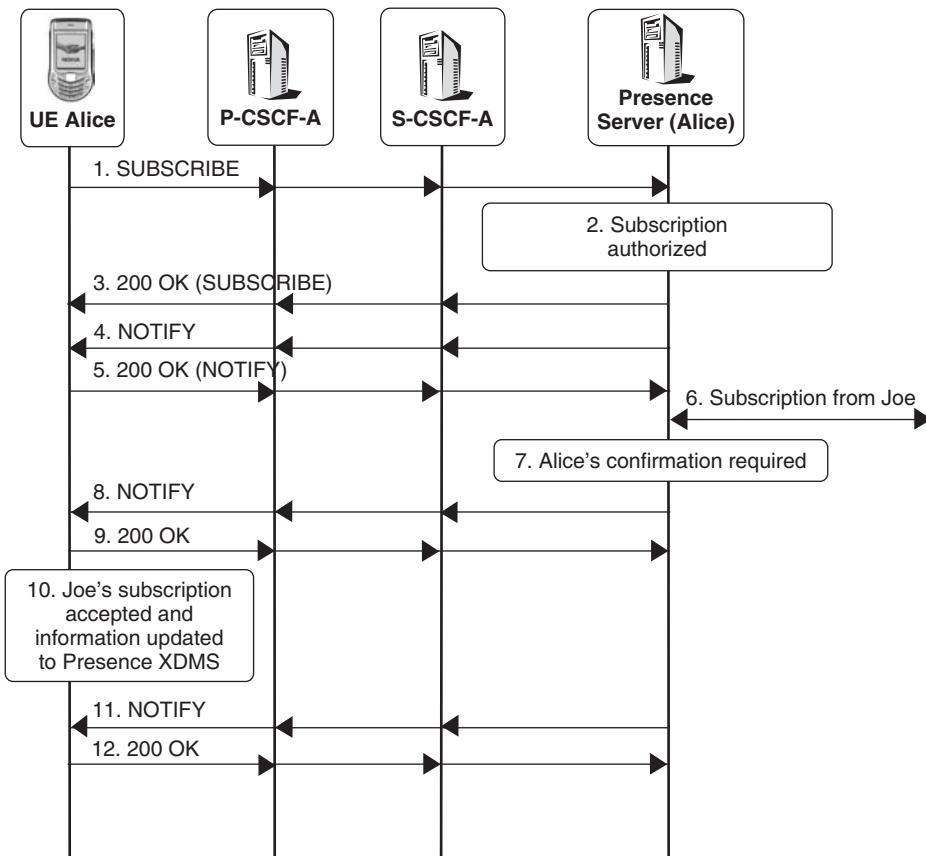
The SUBSCRIBE requests in Step 1 in Figure 4.7 could like like this:

```

SUBSCRIBE sip:alice@example.com SIP/2.0
From: sip:alice@example.com;tag=123s8a
To: sip:alice@example.com
Event: presence.winfo

```

This request gets routed to S-CSCF like any other request and the S-CSCF executes initial filter criteria (see Section 3.12.4) and the trigger point here is Alice's identity and the content of Event header which reveals that this is related to watcher information subscription (winfo token included). The Presence Server accepts the subscription and delivers an initial state of subscriptions. Later on Joe subscribes Alice's presence data in



**Figure 4.7** Subscription to watcher information

Step 6 and the Presence Server performs authorization and learns that Alice's confirmation is required and due to Alice's subscription to watcher information the Presence Server sends a notification to the Alice in Step 8. It could look like this:

```

NOTIFY sip:alice@example.com SIP/2.0
From: sip:alice@example.com;tag=xyz887
To: sip:alice@example.com;tag=123s8a
Event: presence.winfo
Content-Type: application/watcherinfo+xml
Content-Length: ...

<?xml version="1.0"?>
<watcherinfo xmlns="urn:ietf:params:xml:ns:watcherinfo"
    version="0" state="full">
    <watcher-list resource="sip:joe@example.com"
        package="presence">
        <watcher id="77ajsyy76" event="subscribe">

```

```
        status="pending">sip:alice@example.com</watcher>
    </watcher-list>
</watcherinfo>
```

Once Alice's UE gets this notification it can inform that Alice should make a presence authorization decision. Here it is assumed that Alice accepts Joe's subscription and therefore her device updates her presence authorization rules in Presence XDMS (see Section~5.7.1.1.1). Presence Server learns the updated rule and delivers presence data to Joe and sends a new notification to Alice's UE in Step 11. It could look like this:

```
NOTIFY sip:alice@example.com SIP/2.0
From: sip:alice@example.com;tag=xyz887
To: sip:alice@example.com;tag=123s8a
Event: presence.winfo
Content-Type: application/watcherinfo+xml
Content-Length: ...

<?xml version="1.0"?>
<watcherinfo xmlns="urn:ietf:params:xml:ns:watcherinfo"
              version="1" state="partial">
    <watcher-list resource="sip:joe@example.com"
                  package="presence">
        <watcher id="77ajsyy76" event="approved"
                  status="active">sip:alice@example.com</watcher>
    </watcher-list>
</watcherinfo>
```

## 4.9 Setting Presence Authorization

Presence information can be available at different levels and different scopes to different watchers. This means that different watchers may be authorized to view different parts of the presence information of a presentity. The choice of who sees what belongs to the presentity. The presentity can set such authorization levels using an XCAP-defined solution in the form of permission statements. Details on how to do this are described in Section 5.7.1.1.1.



# 5

## Group Management

With the growing need of users to employ a multiplicity of terminals – i.e., mobile phone, PDA, PC – and to have their services available on each of these devices, the situation has arisen in which users expect their service data to be available on all devices, thus obviating the need to build such data more than once.

One solution for such a problem is to use web pages. The problem with web pages is that users have yet to master web browsing on the small screen. Another problem is that solving such a need using the web does not allow such data to be integrated with existing applications running on a mobile phone or any other device.

Let's take a closer look at the issues by taking a real-life example: a user wants to create what is called a 'buddy list' on their PC and mobile phone. Without a group management solution, this user has to create their buddy list twice; once on the PC and once on the mobile. Now, the user walks into an Internet café and wants to use their web messenger. She cannot do so because her buddy list is stored locally on her PC and mobile phone. Had she a web interface that allowed her to build a buddy list that could be stored in the network, her problem would be solved. But, what if her buddy list on her mobile phone used the address book embedded in the device? Different means are needed to enable the device to store such data in the network. The same means can then be used to build a single buddy list. So, instead of the user building multiple duplicated buddy lists, she can build one on her mobile phone address book, for example, and a protocol can be used to upload such a list to the network. Now, when the user logs in from any device, like the PC, that device can automatically contact the network and retrieve the buddy list that was built using the mobile phone, without the need for any user interaction.

Another advantage with such architecture is that the user can create, modify and delete such a list and synchronization will automatically take place since the means for uploading and modifying such a buddy list would also have a built-in functionality to notify other devices of changes to the buddy list.

Users have in their possession service data that can be reused by other services. An example of this is the buddy list just mentioned. A user can use the same list – i.e., their buddy list – in a presence application. They can also use the same list to create a conference call where their buddy list represents the list of participants.

Another creative use of group management is the creation of an Access Control List (ACL). This is a list of users that the user creates as an authorization check by network entities before a communication attempt is relayed. For example, a user, Alice, can create an ACL that allows Bob and John to call her or initiate a Push to talk over Cellular (PoC) session to her while barring Sarah from taking part. The network will then automatically reject any communication attempts made by Sarah towards Alice.

## 5.1 Group Management's Contribution to Business

Group management can contribute to existing businesses and create a business of its own. Operators and other service providers have a major role to play in mass adoption of group management services in combination with other services like presence and instant messaging. A group management service can be part of the operator's service portfolio. The mobile domain, now reaching three billion subscribers worldwide, is a profitable platform for new consumer services. A group management service allowing users to store buddy lists and any other user data related to the presence service or any other service encourages customer loyalty, as shown by instant messaging in the fixed Internet. Users do not want to move to a new subscriber and waste time building their group management data all over again, just to save a couple of dollars.

Offering a basic group management service can give a competitive advantage for an operator over other operators who are not offering it: by binding their group management information to a particular operator's services, customers have available to them high-value services that other operators may not be able to offer without that information. Group management generates new traffic for existing services, like conferencing.

## 5.2 What is Group Management?

Group management (also known as ‘data manipulation’) is a service that allows users to store service-specific data in the service provider network. These data can be created, modified and deleted at will by the user. Data could be anything that a user needs to complete a service.

Various services – such as presence, PoC, IM, Multimedia Telephony etc. – need support for access to and manipulation of certain data that are needed by these services. Some examples of such data include:

- *Resource List:* a list of users who are potential notifiers, so that this list can be used to collectively subscribe to the status of each resource in that list. An example of such a list is the presence list.
- *Access Policy:* a document that contains rules for handling communication attempts to specific user or to specific resource. An example of access policy is barring list for IM communication or setting PoC auto-answer mode for particular users.
- *Configuration data for IMS Supplementary services:* a document that contains values for each active supplementary service. An example of such is call forwarding to a specific number when the user is not reachable.

The services specify the items that make up the documents representing the information in the examples above, including their semantics and usage.

Data are stored in the network where they can be located, accessed and manipulated (created, changed, deleted) by authorized users. This enables data to be shared and accessed by several devices and the services that need them.

Open Mobile Alliance (OMA) has adopted the term XML Document Management (XDM) to be synonymous with the term ‘group management’. The XDM service specifies documents that can be shared by multiple services (called ‘enablers’ by OMA). One such case is a particular type of list, the Universal Resource Identifier (URI) list (also known as a ‘resource list’), which is a convenient way for a user to group together a number of end users (e.g., ‘friends’ or ‘family’) or other resources, where such a list is expected to be reused for a number of different services.

The XML Configuration Access Protocol (XCAP), as defined by Internet Engineering Task Force (IETF) has been selected by OMA and Third Generation Partnership Project (3GPP) as the protocol for transporting, accessing, reading and manipulating the XML documents that contain the data.

### 5.3 What is XML Configuration Access Protocol?

In Extensible Markup Language (XML) Configuration Access Protocol (XCAP) a user is able to upload information to an XCAP server, which provides this uploaded information to application servers that use this information to satisfy a request demanded by the user. With XCAP, the user is also allowed to manipulate, add and delete such data. An example of the data that a user can upload is the user’s resource list for presence. XCAP uses the Hypertext Transfer Protocol (HTTP) to upload and read the information set by users and the Information is represented using XML as shown in Figure 5.1.

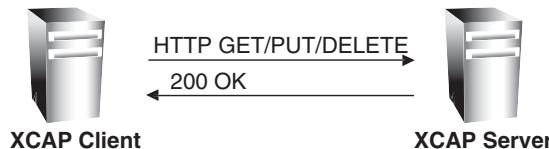
Applications, like presence and IM, need to define an XCAP application usage (AUID), which defines the way that a unique application can make use of XCAP. It defines the XML document for the application.

There are four operations inherited from HTTP namely create (HTTP PUT), fetch (HTTP GET), modify (HTTP PUT) and delete (HTTP DELETE). Each of these operations can be targeted to the whole document, an element in the document or an attribute in the document.

### 5.4 What is Common Policy?

Many applications enable access to information about a user. This can be in the form of presence information or location information, which reveals personal details about the user’s status and whereabouts. The richness of such detailed information is both an extraordinary opportunity for enabling communications as well as a considerable threat to privacy. So, as a result of such applications, there needs to exist a robust and similarly rich system to control the privacy settings of information.

Common Policy defines an authorization policy markup language that can be used to describe the detailed access rights pertaining to an application [RFC4745]. Its origins



**Figure 5.1** XCAP operations

lie in describing privacy settings that relate to geolocation information but, in the spirit of reuse, Common Policy is also the basis for describing presence authorization policies.

In fact, any application that deals with access to a resource – be it in the form of a subscription, a fetch or an invitation of sorts – can take the basic tools of Common Policy and extend its permission statements to suit the application’s special needs.

#### 5.4.1 Model and Rule Structure

The most important aspect of the Common Policy model is that of additive permissions. This means that the privacy settings for a resource always start out with no permissions at all; the settings are then added with grants of permission by the user based on some form of stimulus – e.g., a user might find that another user wants to subscribe to their presence information via the watcher information [RFC3857] mechanism.

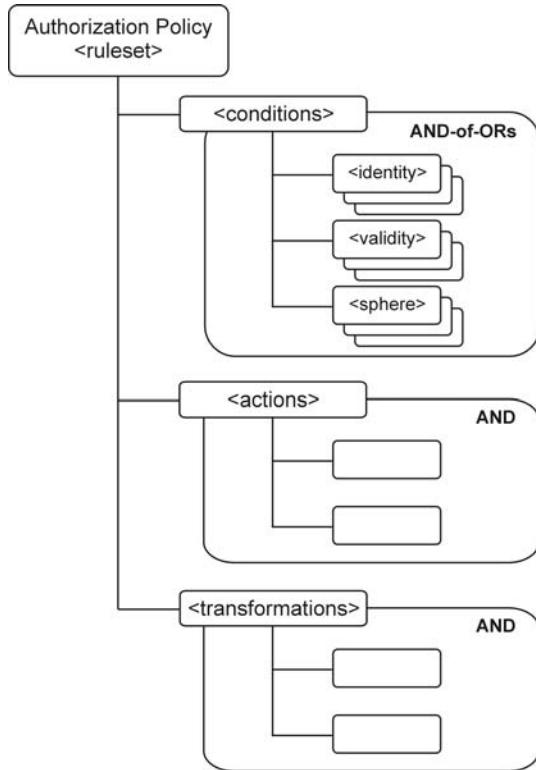
This is in contrast to a traditional blacklist model where everyone basically has access, except the blacklisted individuals. Blacklists have a major drawback in that simply changing one’s identity – called ‘identity minting’ – is enough to circumvent access controls.

Similarly to a whitelist model, additive permissions make the system much more privacy-safe and resistant to identity minting. Simply minting an identity is no longer enough to gain access; instead, a malicious user would actually need to get on another user’s buddy list. Assuming identities are authentic and cannot be faked,<sup>1</sup> getting on someone else’s buddy list is hard to do; it would normally require an exchange of business cards or an earlier communication of one sort or another.

Of course, applications benefit immensely from the ability to receive or grant some level of information to others. This is also the case with Common Policy. Everyone (with an authentic identity) can be granted low-level permission – e.g., to receive the most basic level of presence information or civic location information revealing only the country in which the user currently is – or permission to make requests. More trusted parties, on the other hand, can be given higher level access rights.

Common Policy defines an authorization policy as a set of rules – called a ‘ruleset’ – that govern the access rights to information. Each rule grants permission based on certain matching criteria, like the identity of the user accessing the information, or the date and time of day. These matching criteria are called ‘conditions’. Permission is further divided into an action and a transformation. The action part dictates in which way the system should act, while the transformation part dictates the exact way in which the action should be carried out. This structure is illustrated in Figure 5.2.

<sup>1</sup> This is not an assumption that is easy to make – as the abundance of email spam has taught us.



**Figure 5.2** Common policy data model

#### 5.4.2 Data Types and Permission Processing

As already mentioned, an authorization policy consists of a set of rules, each granting a permission. These rules are independent of each other and are processed in one pass – i.e., the ordering of the rules is not important. After all rules are processed, something called a ‘resultant’ of all granted permissions is generated. This is another important distinction between Common Policy and other access right systems. For instance, the \*nix file system makes use of Access Control Lists (ACLs) and three lists – namely, user, group and others – are visited in order; the first one to match determines the access rights.

The Common Policy processing model means that a user may get granted several permissions, with different levels of access rights. To resolve this ambiguity, an algorithm is provided that combines the different permissions allowed by the various rules. It depends on the data types of permissions – boolean, integer and set.

The boolean type can have only two values: TRUE or FALSE. When combining boolean-type permissions, the OR operation is applied across them. In other words, if one or more of the permissions are TRUE, then the resultant is TRUE; if none of the permissions is TRUE, the resultant for that permission is FALSE. The default is always FALSE; so, in the absence of any matching rule, an automatic FALSE is granted.

The integer type assigns each permission an integer value. The higher the value, the higher the rights assigned to that permission – 0 being the default granting the fewest rights. When combining integer-type permissions, the highest value of granted permissions is selected. In the absence of any rules, the default is taken, which constitutes a setting with the most privacy – i.e., releases little or no information, or blocks sharing of information.

The set type assigns each permission an attribute in a set. Each attribute names a certain permission – the empty set is the default. When combining set type permissions, a union of all permissions is undertaken. For example, if two permissions grant ‘yellow’ and ‘green’, the resultant is a union of the two: {‘yellow’, ‘green’}.

After permission processing, the system is in a position to determine which permissions to apply to the requested information.

## 5.5 Resource List

The Session Initiation Protocol (SIP) specific event notification mechanism allows a user (the subscriber) to request notification of changes to a resource state.

In many cases, a subscriber’s list of resources for a particular event about which she needs state change information is a lengthy one. Without some aggregating mechanism, this will require the subscriber to generate a SIP SUBSCRIBE request for each resource. This will also result in SIP NOTIFY requests arriving at the subscriber’s terminal more rapidly. For congestion control and bandwidth limitations, it is uneconomical to have the user’s terminal send a multiplicity of SUBSCRIBE requests, one for each resource. Figure 5.3 demonstrates the problem.

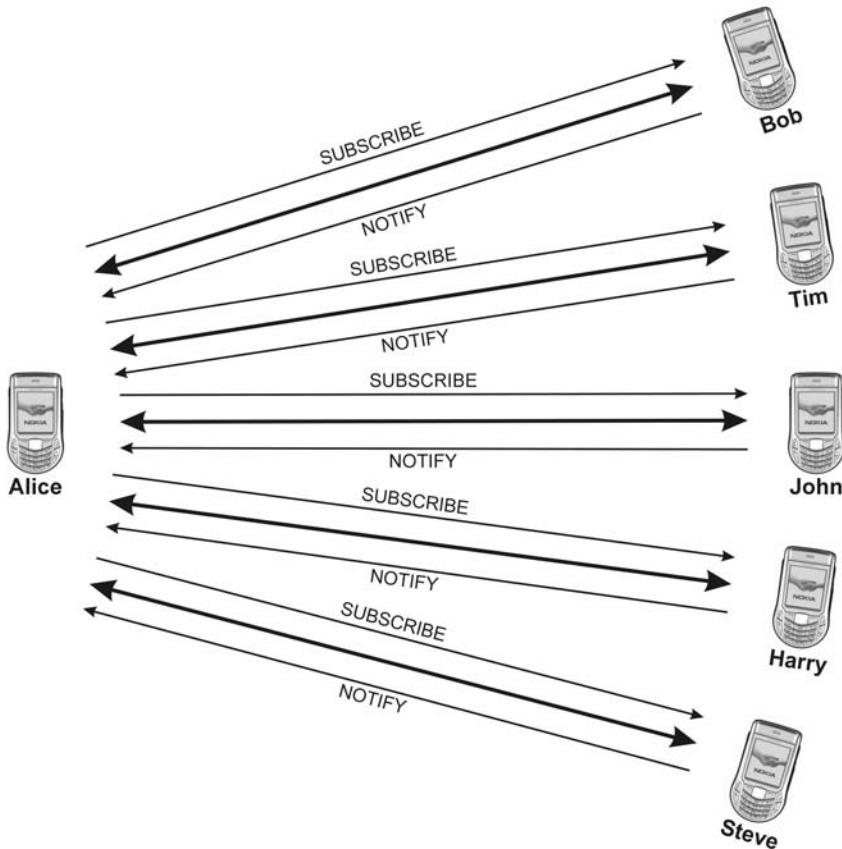
To resolve this problem, [RFC4662] describes an event notification extension that allows users to subscribe to a list of resources with a single SUBSCRIBE request. The list is identified by a URI and contains zero or more URIs pointing to atomic resources or to other lists. In a presence system these resources are presentities. The entity that processes the list’s SUBSCRIBE request is referred to as the Resource List Server (RLS). The RLS can generate individual subscriptions for each resource in the list. These subscriptions may or may not be SIP SUBSCRIBE requests. Figure 5.4 demonstrates the solution.

A client sending SUBSCRIBE requests to a list includes a Supported header with an option tag of value ‘eventlist’. If this option tag is not included and the URI in the request-URI represents a list, the RLS returns a ‘421 Extension Required’ error response.

If the subscription is accepted, then the RLS generates NOTIFY requests carrying state information about the list. The NOTIFY request contains a Require header with an option tag of value ‘eventlist’. It also contains a body of MIME type ‘multipart/related’, which internally carries a MIME type of ‘application/rlmi+xml’ that holds resource list meta-information.

## 5.6 XCAP Usage for Resource Lists

Section 5.5 (above) discusses resource lists and how to gain state information about them. This section discusses how these lists are created and maintained. The list creation solution



**Figure 5.3** Presence subscription example flow, no RLS

takes advantage of XCAP (see Section 5.3) as an application usage. [RFC4826] defines the XML schema along with its semantics.

As an example, Alice is using two terminals, one at home and one in the office. She creates her resource list from her home terminal and adds Bob and Sarah as her buddies. Figure 5.5 illustrates the example. An XCAP request is created carrying the resource (buddy) list to be stored on the server.

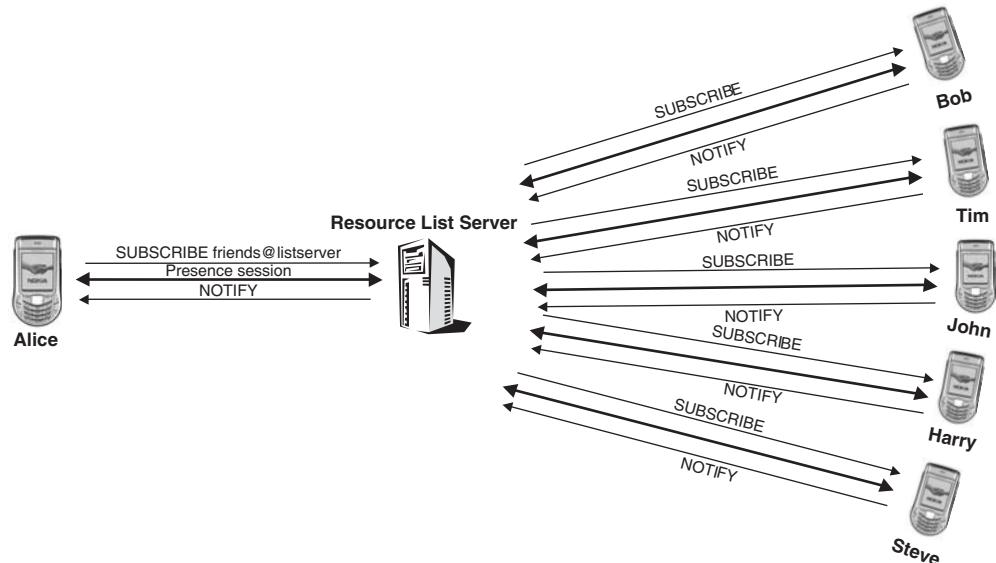
The XCAP request may look like the following:

```
PUT /resource-lists/users/sip:alice@example.com/friends HTTP/1.1
Content-Type:application/resource-lists+xml
Host: xcap.example.com
<?xml version="1.0" encoding="UTF-8"?>
<resource-lists xmlns="urn:ietf:params:xml:ns:resource-lists">
  <list name="friends">
    <entry uri="sip:bob@example.com">
      <display-name>Bob</display-name>
    </entry>
```

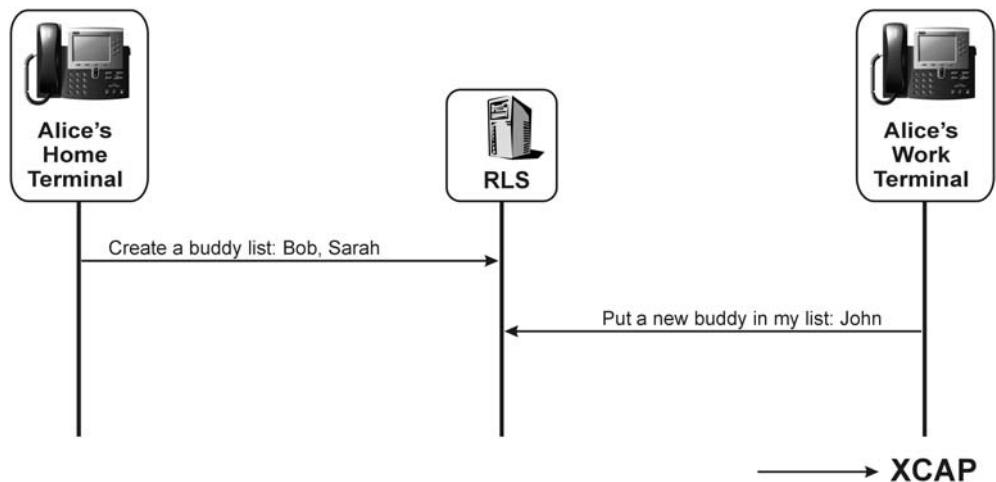
```

<entry uri="sip:sarah@example.com">
<display-name>Sarah</display-name>
</entry>
</list>
</resource-lists>

```



**Figure 5.4** Presence subscription example flow, with RLS



**Figure 5.5** Example resource list flow

Alice then goes to the office and decides that John should also be added as her buddy. She does so by using her work terminal and modifying the list created earlier using her home terminal. An XCAP request is created modifying the existing list on the server.

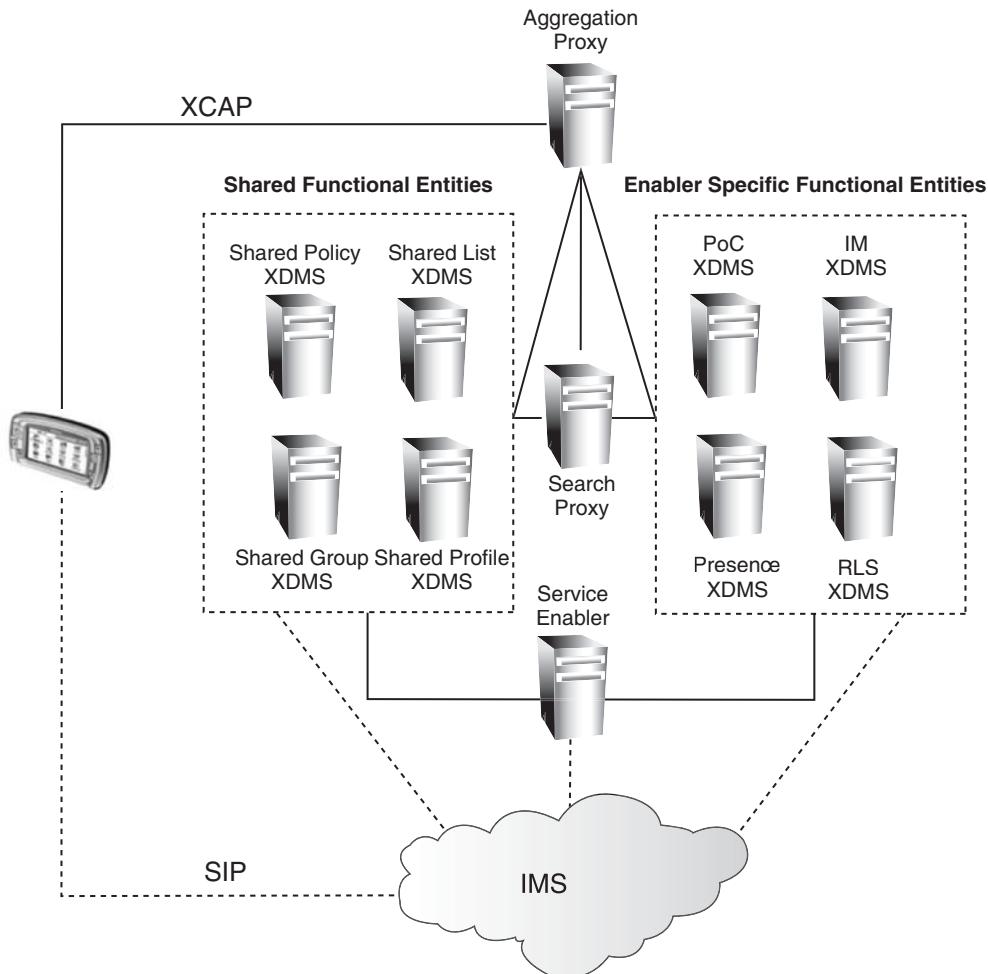
## 5.7 Open Mobile Alliance Solution for Group Management

Late 2005 OMA completed IMS services for Push to talk over Cellular (PoC) and Presence. At the same time OMA completed its first Group Management service aka XML Document Management enabler release 1.0. This package defined common mechanisms that made user-specific service-related information accessible to IMS clients, PoC servers and Presence servers. More precisely architecture, well-structured XML documents, as well as the common protocol for access and manipulation of such XML documents were standardized. Summer 2007 OMA released next version of its XDM solution, XDM release 2.0. The second release added support for OMA IM, OMA PoC 2.0 and it generalized architecture to support a wide range of future applications. Description here is mainly based on OMA XDM 2.0 release. Figure 5.6. shows XDM architecture. From the figure main building blocks and features of XDM service can be listed.

- XML Configuration Access Protocol (XCAP) [RFC4825], the common protocol, by which service data owner(s) can store and manipulate their service-related data, stored in a network as XML documents.
- SIP, the subscription/notification mechanism by which data owners can be notified of changes to such documents.
- XQuery, the mechanism by which users can search service-related data stored in a network as XML documents.
- XDM Client (XDMC), an application in IMS UE that provides access to the various XDMS features using XCAP.
- Shared XDM Server (XDMS), a server which can be reused by multiple different application servers. There are four types of shared XDMS: Shared List XDMS, Shared Group XDMS, Shared Policy XDMS and Shared Profile XDMS.
- Enabler specific XDMS, a server dedicated to support particular application only (e.g. IM XDMS, Presence XDMS).
- Search proxy, a server that receives search request (e.g. find a group that is chatting on results of American Idol contest) from XDM clients and makes queries to individual XDMSs and combines results back to the XDM clients.
- Aggregation proxy, an entity that authenticates XCAP and search requests from XDM clients, finds right target XDMS for XCAP requests and routes search requests to Search proxy.

### 5.7.1 Service Specific XML Document Management Servers

At the time of writing OMA has defined four service specific XDMS as can be seen from Figure 5.6.



**Figure 5.6** OMA XDM architecture

- IM XDMS: a server that stores conversation history metadata and deferred messaging metadata.
- PoC XDMS: a server that stores PoC group properties and information about PoC user access policies (This entity was introduced in XDM 1.0 release to support OMA PoC 1.0. OMA PoC 2.0 uses general Shared XDMS instead of PoC XDMS).
- Presence XDMS: a server that contains rules for presence information subscriptions and rules for presence information publication.
- RLS XDMS: a server that contains user's presence buddylist.

### 5.7.1.1 Presence Specific XML Document Management Server

Presence is a dynamic profile of the user, which is visible to others and used to represent oneself, share information and control services. Presence information is also personal: it is always linked to a particular person. It shows the person initiating the communication whether the other person is available and willing to communicate. Presence information sharing raises security and privacy concerns. For protecting user's presence data and for applying a different level of privacy of presence information Presence XML Document Management server exists. Another presence specific XDMS is Resource List Server XML Document Management Server. It is used to mitigate the issue outlined in Section 5.5.

#### 5.7.1.1.1 Presence XML Document Management Server

In presence service it is equally important to ensure that only authorized persons can publish presence data for a particular user and only persons that presence owner has authorized can get their presence data. To govern the former Presence XDMS stores information regarding publication authorization rules. Subscription authorization rules are stored to enforce presence subscriptions. In addition to these two policies the Presence XDMS may store permanent presence data.

A rule for presence contains three parts: conditions, actions and transformations. The condition part defines e.g. the set of watchers to whom an authorization rule applies. Identities can be defined by 'uri' or 'domain'. The action part defines e.g. how the presence subscription is handled (allow, polite-block, confirm, block). The transformation part defines which presence information are provided (visibility to tuple, person and device data elements, permissions to particular set of presence attributes) or allowed to be published.

OMA has defined AUID 'org.openmobilealliance.pres-rules' for Subscription Authorisation Rules and 'org.openmobilealliance.pub-rules' for Publication Authorisation Rules. The former conforms with RFC5025 and the latter with RFC4745.

The conditions for both policies include the identity of the subscriber, the identity of a subscriber derived from an external list, other subscriber identities that are not present in any rule and the anonymous subscriber.

The actions for subscription policy includes allowing the subscription, blocking the subscription, asking further confirmation from the presence data owner and allowing the subscription without providing any useful presence data i.e. executing so called polite-blocking. The actions for publication policy include allowing publishing and blocking the publishing.

The transformation defines rules regarding presence content. In other words, which parts of user's presence data the Presence server gives to a particular subscriber or which part of user's presence data the publisher can publish.

As an example a presence owner may define two rules for subscription policy. The first rule is to give all available presence information for family members and the second rule is to show only available services and willingness for communication to all others.

```

<cr:rule id="family">
    <cr:conditions>
        <cr:identity>
            <cr:id>husband@example.com</cr:id>
            <cr:id>son@example.com</cr:id>
            <cr:id>daughter@example.com</cr:id>
        </cr:identity>
    </cr:conditions>
    <cr:actions>
        <pr:sub-handling>allow</pr:sub-handling>
    </cr:actions>
    <cr:transformations>
        <provide-all-attributes>
    </cr:transformations>
</cr:rule>

<cr:rule id="rest">
    <cr:conditions>
        <other-identity/>
    </cr:conditions>
    <cr:actions>
        <pr:sub-handling>allow</pr:sub-handling>
    </cr:actions>
    <cr:transformations>
        <pr:provide-services>
            <op:service-id>org.openmobilealliance:PoC-session</op:
                service-id>
        </pr:provide-services>
        <op:provide-willingness>true</op:provide-willingness>
    </cr:transformations>
</cr:rule>

```

#### **5.7.1.1.2 Resource List Server XML Document Management Server**

Most likely users will have many persons whose presence information is of interest to them. For congestion control and bandwidth limitations, it is uneconomical to have the user' device send a multiplicity of SUBSCRIBE requests, one for each person. To solve this issue, RLS is used. RLS XDMS is in fact storage for resource lists (see Sections 5.5 and 5.6). Typically user creates single list such as 'sip:friends@example' to the RLS XDMS and this list could contain a number of individual entries or preferably it could contain reference to a list which is stored in Shared List XDMS (see Section 5.7.2.2). How this list is used in presence service is shown in Section 4.7 (Presence Subscription).

For this functionality OMA uses IETF defined AUID 'rls-services' as described in [RFC4826].

#### **5.7.1.2 PoC Specific XML Document Management Server**

One of key features of PoC is that users are able to create personalized and longlasting communication groups whenever they like and as many as the operator allows. Moreover,

the user should be able to define whether the group is open to everybody or does it have a well defined member list and whether users should dial-in themselves or should PoC server invite them based on a trigger from specific user(s). During the design phase of PoC it was realized that storing this type of information at the PoC server or PoC clients is not a sound long term architectural solution. Therefore it was agreed to store this data elsewhere. This turned out to be a PoC XDMS.<sup>2</sup> At the time of OMA PoC release 1.0 standardization (2003-2005) commercial market pressure was very high and the outcome was that a PoC specific XDMS was created instead of creating media independent group management solution. It is fair to note that parallel to OMA work IETF was standardizing media independent solution for conference control which still does not exist today. PoC XDMS consists of two XCAP application usages:

- PoC group application usage: is a list of PoC participants who can take part in a PoC session as well as additional PoC-specific properties.
- PoC user access policy application usage: is a set of rules that a user creates to control who can and who cannot initiate a PoC session to him/her.

#### *PoC group*

A PoC group is the list of PoC participants who can take part in a PoC session as well as additional PoC-specific properties. [OMA-TS-PoC\_XDM] defines the XML schema along with its semantics. It defines the AUID as ‘org.openmobilealliance.poc-groups’.

PoC group document contains the following:

- a URI representing the PoC group identity;
- a display name for the group;
- lists of group members (URIs);
- an indication as to whether the group members will be invited to the PoC session by the PoC server or not;
- a maximum number of participants for a particular PoC session from that group;
- an authorization policy associated with the group.

Each list of group members needs to contain entries identifying a single user by a SIP or tel URI, or a reference to an external list URI.

The structure of the authorization policy (also referred to as a ‘ruleset’) must conform to the Common Policy [RFC4745]. The conditions for such an authorization policy include the identity of the participant, the identity of a participant derived from an external list, other identities that are not present in any rule and a condition that checks whether the user is a list member.

The actions that can be taken if a condition returns as ‘true’ are: allow the user to subscribe to the PoC session state, allow the user to dynamically invite participants to join the PoC session, allow the user to join, bar the user from joining, allow the user to start the session, allow the user to be anonymous and treat the user as a key participant.

As an example, Alice wants to create a PoC session with Bob, Sarah and John. This group is her friends group. She would like her friends to join on their own – i.e., they will

---

<sup>2</sup> This can be considered as the birth of OMA XDM work.

not be invited by the server. The maximum number of people joining the session cannot exceed four. This allows Alice to add more friends to the list but still limit the number of participants as she does not want too many people to be involved in discussions.

Alice wants to authorize those people who constitute only a small part of the member's list (currently Bob, Sarah and John) to be able to join the conference and subscribe to the session state. No anonymity is allowed. Below is what the XCAP request would look like. The XML document is also shown as the payload:

```
PUT org.openmobilealliance.pocgroups/users/
    sip:alice@example.com/friendsgroup.xml HTTP/1.1 ...
Host: xcap.example.com
Content-Type: application/list-service+xml

<?xml version="1.0" encoding="UTF-8"?>
<group xmlns="urn:oma:params:xml:ns:list-service"
xmlns:r1="urn:ietf:params:xml:ns:resource-lists"
xmlns:cr="urn:oma:params:xml:ns:common-policy">
<list-service uri="sip:myconference@example.com">
    <list>
        <r1:entry uri="sip:bob@example.com"/>
        <r1:entry uri="sip:sarah@example.com"/>
        <r1:entry uri="sip:john@example.com"/>
    </list>
    <display-name xml:lang="en-us">Friends</display-name>
    <invite-members>false</invite-members>
    <max-participant-count>4</max-participant-count>
    <cr:ruleset>
        <cr:rule id="1a">
            <cr:conditions>
                <cr:is-list-member/>
            </cr:conditions>
            <cr:actions>
                <allow-conference-state>true</allow-conference-state>
                <join-handling>allow</join-handling>
                <allow-anonymity>false</allow-anonymity>
            </cr:actions>
        </cr:rule>
    </cr:ruleset>
</list-service>
</group>
```

#### *PoC user access policy*

The PoC user access policy is a set of rules that a user creates to control who can and who cannot initiate a PoC session to him/her.

[OMA-TS-PoC\_XDM-V1\_0] defines the XML schema along with its semantics. It defines the AUID as ‘org.openmobilealliance.poc-rules’.

The structure of the access policy (also referred to as a ‘ruleset’) conforms to the Common Policy [RFC4745]. The conditions for such an authorization policy include the identity of the participant from whom the user is accepting or rejecting a PoC session,

an identity of a participant derived from an external list and other identities that are not present in any rule.

The only action defined for a condition that has been satisfied is what to do with the invitation. One of three values may appear: pass, allow and accept. ‘Pass’ instructs the PoC server to process the PoC session invitation using manual answer procedures (see Section 6.3.4). ‘Accept’ instructs the PoC server to accept the invitation according to the user’s answer mode setting, while ‘reject’ instructs the PoC server to reject the invitation.

To give an example, Alice is creating a PoC user access policy that only allows her friends to invite her to a PoC session. Below is what the XCAP request would look like. The XML document is also shown as the payload:

```
PUT org.openmobilealliance.poc-
rules/ users/sip:alice@example.com/pocrules HTTP/1.1
Host: xcap.example.com
Content-Type: application/auth-policy+xml

<?xml version="1.0" encoding="UTF-8"?>
<ruleset xmlns="urn:ietf:params:xml:ns:common-policy"
          xmlns:poc="urn:oma:params:xml:ns:poc-rules"
          xmlns:ocp="urn:oma:xml:xdm:common-policy">
    <rule id="f3g44r1">
        <conditions>
            <identity>
                <one id="tel:+358501111111"/>
                <one id="tel:+358402222222"/>
                <one id="sip:myfriend1@example.com"/>
                <one id="sip:myfriend2@example.com"/>
            </identity>
        </conditions>
        <actions>
            <poc:allow-invite>accept</poc:allow-invite>
        </actions>
    </rule>
    <rule id="f3g44r1">
        <conditions>
            <ocp:other-identity/>
        </conditions>
        <actions>
            <poc:allow-invite>reject</poc:allow-invite>
        </actions>
    </rule>
</ruleset>
```

### 5.7.1.3 IM Specific XML Document Management Server

OMA Instant Messaging will differentiate from current and traditional operator messaging solutions in multiple ways. One example is capability to review and select which deferred messages the user wants to download to their client. This type of capability does

not exist in SMS and MMS services but exists in email systems where for example users can download most important email message headers (e.g. to, from, date, time, subject) instead of full email. This is a nice feature when the user is behind slow link and emails contain big attachments. OMA IM allows this type of capability by storing most important and informative part of messages received offline to an IM specific XDMS server while the actual message is still stored in the IM server. Another capability is to store user's messages to the network and make this archive searchable and browseable. This previously mentioned information is stored with two XCAP application usages in IM XDMS.

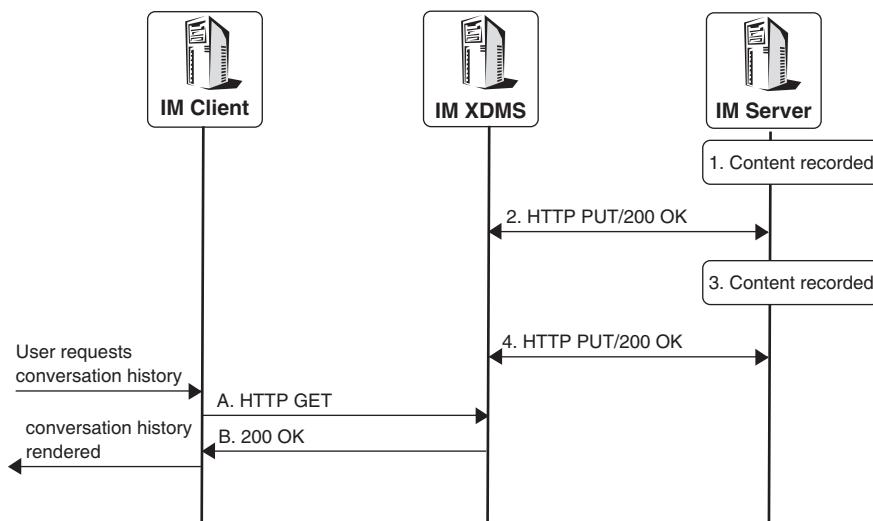
#### *Conversation history metadata*

When a user has activated the conversation history feature the IM server taking care of storing the conversation history also takes care of storing conversation history metadata to the IM XDMS (see Section 7.5.3 for details of conversation history function). The metadata gets stored once content is recorded at the IM server. After that content can be retrieved by the client as shown in Figure 5.7.

[OMA-TS-IM\_XDM] defines the XML schema along with its semantics. It defines the AUID as ‘org.openmobilealliance.conv-history’.

IM Conversation history metadata application usage contains the following:

- *date* representing the date at which the conversation history recording began;
- *history-reference* uniquely identifying a particular message from other messages;
- *size* representing the size of the saved content;
- *expire* representing the date at which the history expires;
- *subject* representing the subject header in SIP;



**Figure 5.7** Storing conversation history metadata and retrieving it

- *pager* containing information of pager mode message or large message mode as follows:
  - *time-stamp* representing the time when the message was recorded;
  - *from* ‘From’ header field of the SIP request;
  - *to or recipient-list* ‘To’ header field of the SIP request or content of URI-list when request is sent to multiple users;
  - *auth-id* representing network asserted identity of the sender.
- *conference* containing information of IM session mode messaging as follows:
  - *time-start* representing the start time of the history recording;
  - *time-end* representing the end time of the history recording;
  - *recording-name* element representing the user given name for the recorded conference;
  - *conf-list* containing all the participants to the conference.

As an example, Bob worked at the office with his PC and he processed all incoming messages with it. While at the home Bob suddenly realizes that he does not remember any more car pool members for tonight’s play-off game. Luckily, he is using conversation history function and he does not need to bother others as he can find previous discussion with his mobile device as well. Bob opens the IM client and the IM client fetch the messaging history file from the IM XDMS and shows one instant message from David and one conversation regarding car pool with David and John. Based on this information Bob recalls it was indeed John who bought tickets from George. The document which has been stored in IM XDMS and the document delivered to Bob’s device could look like this.

```
<?xml version="1.0" encoding="UTF-8"?>
<history-list xmlns="urn:oma:xml:im:history-list">

<history date="2008-03-07" history-reference="sip:12@history.example.com">
  <size>10</size>
  <expiry>2008-03-09T19:13:00.0Z</expiry>
  <subject>are you coming?</subject>
  <pager>
    <time-stamp>2006-03-07T09:13:00.0Z</time-stamp>
    <from>sip:david@example.com</from>
    <to>sip:Bob@example.com</to>
    <auth-id>sip:david@example.com</auth-id>
  </pager>
</history>

<history date="2008-03-07" history-reference="sip:13@history.example.com">
  <size>25</size>
  <expiry>2008-03-09T15:05:00.0Z</expiry>
  <subject>"Car pool"</subject>
  <conference>
```

```

<time-start>2008-03-07T15:05:00.0Z </time-start>
<time-end>2008-03-07T15:15:00.0Z </time-end>
<recording-name>friends</recording-name>
<conf-list>
<entry uri="sip:david@example.com">
    <display-name>David</display-name>

<entry uri="sip:john@example.com">
    <display-name>John</display-name>
</entry>
    <entry uri="sip:bob@example.com" />
</conf-list>
</conference>
</history>

</history-list>

```

#### *Deferred messaging metadata*

When a user is offline the deferring messaging function of the IM server stores received immediate messages and in addition it stores messaging metadata to the IM XDMS. The purpose of this metadata is to enable selective and user initiated downloading of deferred messages (see Section 7.5.4 deferred messaging for details of deferred messaging).

[OMA-TS-IM\_XDM] defines the XML schema along with its semantics. It defines the AUID as ‘org.openmobilealliance.deferred-list’. IM Conversation history metadata application usage contains the same elements and attributes as conversation history metadata except conference element.

#### 5.7.2 Shared XML Document Management Servers

In OMA XDM architecture four entities are designed to offer maximum re-use for all kind of services. So far four different shared XDMSs are specified:

- Shared List XDMS: a server that stores all kind of user lists which can be re-used in other XDMSs.
- Shared Group XDMS: a server that stores information and policies regarding user/service provider created groups.
- Shared Policy XDMS: a server that stores policies on how incoming communication attempts should be treated.
- Shared Profile XDMS: a server that stores a profile which the user wishes to make available to other users.

##### 5.7.2.1 Shared Group XML Document Management Server

Section 5.7.1.1.2 described that PoC XDMS supported two PoC specific XCAP application usages; PoC group application and PoC user access policy. Shared Group XDMS can be seen as a next generation of PoC group application usage as it is backward compatible but it will also support other services than PoC.

As an example, when a user wishes to create a group let's say for wedding planning they want to form a temporal group that would be valid for the next three months and it could be used for all kinds of communication (e.g. IM, PoC, voice conference, video conference), moreover, she wishes that all members can invite additional members to communicate on a need basis. Without Shared Group XDMS users would need to create a group definition to multiple places. Now, users can define a group once and use it with multiple applications.

To enable generic group definition OMA has defined the XML schema along with its semantics. It defines the AUID as 'org.openmobilealliance.groups'. This application usage contains the following information:

- 
- A URI representing the group identity
  - subject describing the group
  - an indication whether the group members will be invited to the group session by the server (e.g. PoC server etc.) or not
  - a age limitation defining allowed age or age-range(s) of a group member
  - an indication as to whether the Shared Group XDMS will advertise the created group automatically to the group members or not
  - list of services that could use this group
  - quality of experience associated to the group
  - a display name for the group
  - list of group members (URIs)
  - a maximum number of participants for a particular PoC session from that group
  - a session activation policy associated with the group
  - an indication whether the Group Identity can be retrieved using a search sequest
  - an authorization policy associated with the group
- 

Each list of group members needs to contain entries identifying a single user by a SIP or tel URI, or a reference to an external list URI.

The structure of the authorization policy (also referred to as a 'ruleset') conforms to the Common Policy [RFC4745] and additional clarifications by OMA. The conditions for such an authorization policy include the identity of the participant, the identity of a participant derived from an external list, other identities that are not present in any rule, a condition that checks whether the user is a list member and a condition that defines allowed medias for the user in the group.

The actions that can be taken if a condition returns as 'true' are: allow the user to subscribe to the session state, allow the user to dynamically invite participants to join the group session, allow the user to join, bar the user from joining, allow the user to start the session, allow the user to be anonymous, treat the user as a key participant, allow the user to create subconferences, allow the user to send private messages in the group, allow

the user to initiate a session with particular media or to add new media, allow the user to remove other users from the session, bar the user from sending group advertisement. In addition, there is a condition for what media the user is allowed to remove (none, all, media they have added themselves).

### 5.7.2.2 Shared List XDMS

In OMA XDM architecture Shared List XDMS is the entity to store all kind of users' lists. The shared list can contain lists that the user has created themselves (e.g. mygolfbuddies and myfamily). For this purpose IETF resource-list AUID (see Section 5.6 for additional details) is used. In addition, to the user created lists the user can add other users' created lists<sup>3</sup> (e.g. company\_distributionlist and volunteer\_fire-brigade\_unit3) inside the shared list using OMA specific AUID org.openmobilealliance.group-usage-list.

Once the list is created to the Shared List XDMS the user is able to utilize newly create lists in other applications. As an example the user could create an IM distribution list to the Shared Group XDMS and she could add e.g. mygolfbuddies and myfamily as members of the list among selected individual users. Similarly the user could re-use content of mygolfbuddies and myfamily when setting presence subscription to the RLS XDMS.

To improve re-use of lists even further and minimize migration problems between devices OMA has defined four well known names for shared lists as follows:

- oma\_allcontacts: This name is used to store all users' URIs that it knows about, in one list independent of how the URIs are used.
- oma\_buddylist: This name is used to store all users' URIs that it wants to use for all types of communication, in one list.
- oma\_pocbuddylist: This name is used to store users' URIs that it wants to use for PoC communication, in one list.
- oma\_blockedcontacts. This name is used to store users' URIs that it wants to block/reject in a number of Application Usages, in one list.

As an example the user could have one user Alice and a reference to the oma\_buddylist in her oma\_allcontacts list. The list oma\_buddylist contains a reference to the oma\_pocbuddylist and reference to list-a. The list oma\_pocbuddylist contains one user 'Carl' and a reference to list-b. The owner of the list wants 'Carl' and all members of list-b to be visible as Push to talk buddies. List-a and list-b then contains individual members of particular list.

### 5.7.2.3 Shared Policy XDMS

Section 5.7.1.2 described that PoC XDMS supported two PoC specific XCAP application usages; PoC group application usage and PoC user access policy application usage. Shared Policy XDMS can be seen as a second generation of PoC user access policy application usage as it is backwards compatible but it will also support other services than PoC.

The shared policy is a set of rules that a user creates to control who can and who cannot initiate a communication to him/her so it can be seen as a capability which is only executed

---

<sup>3</sup> Usually, the user is not allowed to modify the content of these lists (e.g. delete or add members).

in the terminating network. For this purpose dedicated, AUID‘org.openmobilealliance.access-rules’ is defined.

The structure of the shared policy (also referred to as a ‘ruleset’) conforms to the Common Policy [RFC4745] (see Section 5.4). The conditions for such an authorization policy include the identity of the participant from whom the user is accepting or rejecting a communication request, an identity of a participant derived from an external list, other identities that are not present in any rule, anonymous request, offered media and offered service. Example conditions are: if the request is received from Bob then . . . , if the request is received from mygolfbuddies then . . . , if the request is not coming from Robert then . . . , if the request is for message session then . . .

All existing actions are related to how to treat an incoming communication request. The following actions are possible; accept incoming communication attempt, bar incoming communication attempt, apply automatic answer mode, apply manual answer mode, forward attempt to offline communication store.

#### 5.7.2.4 Shared Profile XDMS

Shared Profile XDMS contains one or more information elements of the user that they wish to make available to other users. It includes the following information:

- Communication identity (SIP URI, TEL URI, E.164 number, email address) and display Name (e.g. Joe Smith).
- Name of the user (given-name, family name, middle-name, name-suffix, name-prefix), gender of the user and birth date.
- Postal address of the user.
- Freetext inserted by the user.
- Communication types and user’s favourite links.

For this purpose OMA has defined the XML schema along with its semantics. It defines the AUID as ‘org.openmobilealliance.user-profile’.

This type of online telephone directory is searchable so a user could for example use her XCAP client to find information on specific user (alice@example.com) or users living in Finland and interested in bird watching.

## 5.8 Multimedia Telephony and Service Management

IMS multimedia telephony is a blended multimedia service suite. It allows users to establish communications between them and enrich that by enabling supplementary services as described in Chapter 9. Four supplementary services can be configured with XCAP, namely: communication barring, communication diversion, originating identification and terminating identification. This section describes how users can manage these supplementary service settings with XCAP.

XCAP usage for supplementary services was invented in ETSI TISPAN standardization forum around 2005 and it was included in ETSI TISPAN NGN Release 1 as a new AUID simservs.ngn.etsi.org. During 2007 this work was moved to 3GPP to be part of 3GPP Release 7. Description here is based on 3GPP Release 7.

### 5.8.1 *Communication Barring*

Three different types of communication barring services are defined for multimedia telephony: incoming communication barring, anonymous communication barring and outgoing communication barring (see Section 9.3.1). For these an authorization policy (also referred to as a ‘ruleset’) has been defined and it conforms to the Common Policy [RFC4745] (see Section 5.4). The following conditions have been defined: the identity of the calling user, the identity of the called user, an identity of the calling user derived from an external list, other identities that are not present in any rule, anonymous calling user, offered media, the called user’s current presence status, time, roaming status and diverted session. For each condition or set of conditions action is defined. Action takes either value true or false. When one of the actions is evaluated true then the session is allowed to continue otherwise the session is blocked.

An example is authorization policies: if the session is received from Bob then block, if the session is not coming from Robert then allow, if the request is for video then bar, if my presence status is offline and time is 8am–4pm then bar, if the session is targeted to domain ‘expensive.com’ then bar, if I am roaming then bar.

### 5.8.2 *Communication Diversion*

Communication diversion service allows user to re-direct an incoming request that fulfils certain provisioned or configured conditions to another destination (see Section 9.3.2). To make this possible an authorization policy (also referred to as a ‘ruleset’) has been defined and it conforms to the Common Policy [RFC4745] (see Section 5.4). The following conditions have been defined: the called user is busy, the called user is not registered, the called user’s current presence status, the identity of the calling user, anonymous calling user, time, offered media, the called user does not answer, an identity of the calling user derived from an external list, other identities that are not present in any rule, the called user is not reachable. When the condition or set of conditions is evaluated true then the only defined action is to forward the session to a given destination.

An example of authorization policies: if I’m not responding then forward the session to multimedia mailbox, if I’m busy and my presence status is meeting and the caller is my wife then forward to my secretary, if I’m not registered to the network then forward to multimedia mailbox.

### 5.8.3 *Originating Identification Services*

Originating Identification Restriction (OIR) is a service that an originating user can use when he/she does not want to expose his/her real identity to the other party/parties (see Section 9.3.7). There are two ways for end users to use this service: either to use SIP protocol itself for activating OIR service, a session basis or set more static rule (OIR active/deactive) to the network with XCAP protocol.

In addition, AUID simservs.ngn.etsi.org contains a separate element to activate Originating Identification Presentation (OIP) service (see Section 9.3.6).

### 5.8.4 Terminating Identification Services

Termination Identification Restriction (TIR) is a service that a terminating user can use when he/she does not want to expose his/her real identity to the other party/parties (see Section 9.3.9). There are two ways for end users to use this service: either to use SIP protocol itself for activating TIR service, a session basis or set more static rule (TIR active/deactive) to the network with XCAP protocol.

In addition, AUID simservs.ngn.etsi.org contains separate element to activate Terminating Identification Presentation (TIP) service (see Section 9.3.8).

### 5.8.5 Multimedia Telephony Service Management Example

In this example a user expresses that they allow to deliver their identity to called parties and they want to block communication coming from sip:baduser@example.com and tel:+1-666-666-6666 and in addition they desire to forward communication attempts to the mailbox when they are busy with another session.

```
<?xml version="1.0" encoding="UTF-8"?>
<simservs
  xmlns="http://uri.etsi.org/ngn/params/xml/simservs/xcap"
  xmlns:cp="urn:ietf:params:xml:ns:common-policy">

  <originating-identity-presentation-restriction active="true">
    <default-behaviour>presentation-not-restricted
    </default-behaviour>
  </originating-identity-presentation-restriction>

  <incoming-communication-barring active="true">
    <cp:ruleset>
      <cp:rule id="rule66">
        <cp:conditions>
          <identity>
            <one id="sip:baduser@example.com"/>
            <one id="tel:+1-666-666-6666"/>
          </identity>
        </cp:conditions>
        <cp:actions>
          <allow>false</allow>
        </cp:actions>
      </cp:rule>
    </cp:ruleset>
  </incoming-communication-barring>

  <communication-diversion active="true">
    <cp:ruleset>
      <cp:rule id="CFB">
```

```
<cp:conditions>
    <busy/>
</cp:conditions>
<cp:actions>
    <forward-to>
        <target>
            Sip:mymailbox@example.com
        </target>
    </forward-to>
</cp:actions>
</cp:rule>
</cp:ruleset>
</communication-diversion>
</simsservs>
```

# 6

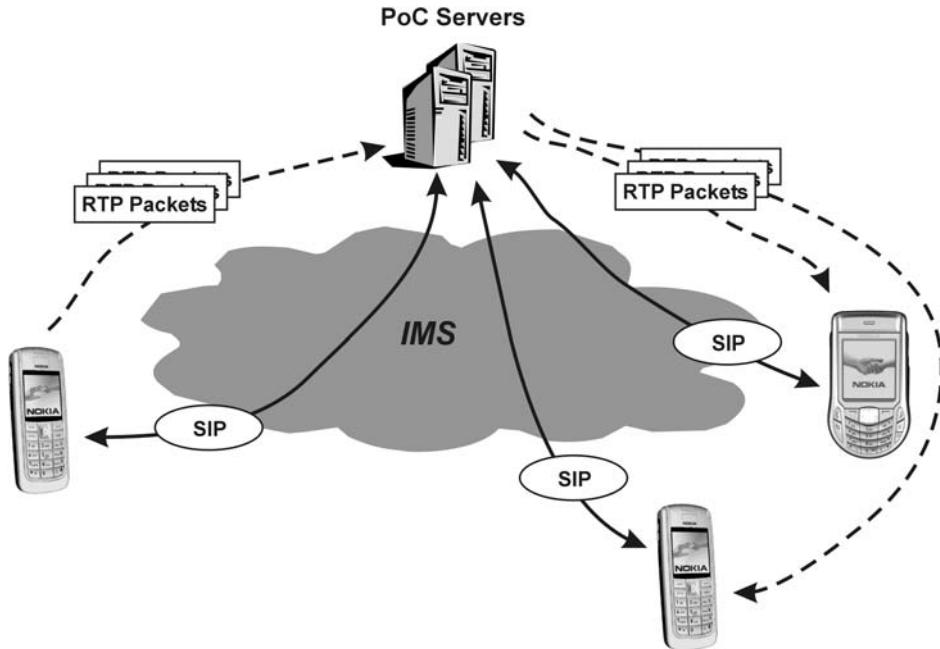
## Push to Talk Over Cellular

Push to talk over Cellular (PoC) provides a direct one-to-one and one-to-many voice communication service. The idea is simple. Users select the individuals or groups they wish to talk to, and then press the push to talk key to start talking. The session is connected in real time. Push to talk sessions are one-way communication: while one person speaks, the other(s) only listens. The turns to speak are requested by pressing the push to talk key and granted on a first-come-first-served basis. Push to talk speech is usually connected without the recipients answering and heard through the phone's built-in loudspeaker. Alternatively, a user can choose to receive push to talk sessions only after accepting an invitation. If more privacy is needed, they can also listen to sessions through an earphone or headset.

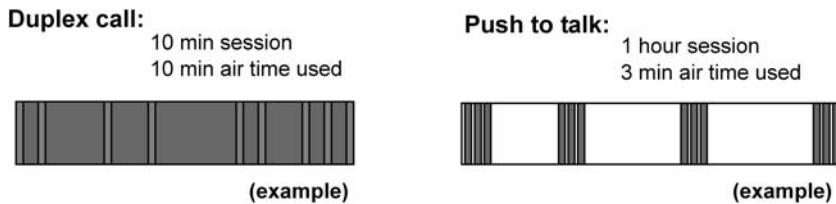
The push to talk service is based on multi-unicasting. Each sending client sends packet data traffic to a dedicated push to talk application server and, in the case of a group session, the server then duplicates the traffic to all the recipients (see Figure 6.1). No multicasting is performed either in the access or core network and mobility management is carried out by the radio network. This is why the push to talk solution works transparently over cellular and fixed networks. PoC session control and other signalling is based on Session Initiation Protocol (SIP) and voice traffic is carried through a Real-time Transport Protocol/Real-time Transport Control Protocol (RTP/RTCP) based streaming bearer.

PoC uses cellular access and radio resources more efficiently than circuit-switched services. Network resources are reserved one-way for the duration of talk bursts, rather than two-way for an entire session. Figure 6.2 shows an example. Compared with conventional two-way radio solutions, such as Land Mobile Radio (LMR) and Professional Mobile Radio (PMR), as well as the Family Radio Service (FRS), PoC provides better coverage as it utilizes coverage provided by GSM/WCDMA/CDMA networks. It allows users to make push to talk sessions between two people or within a group of people over nationwide networks and across regional borders (GPRS EDGE/WCDMA/CDMA2000).

In this chapter we describe PoC as defined in Open Mobile Alliance PoC standard Release 1.



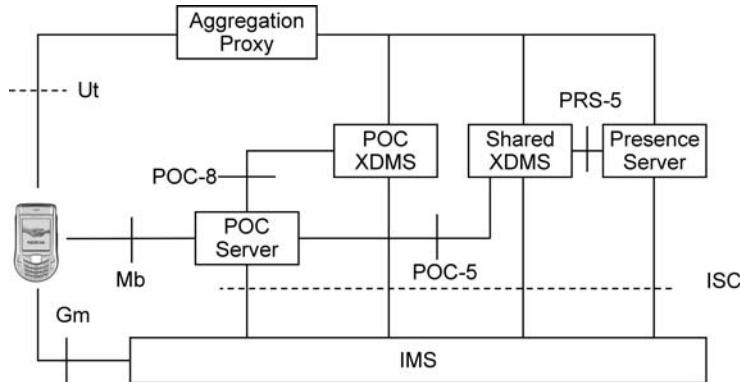
**Figure 6.1** Push to talk over cellular



**Figure 6.2** Voice call versus push to talk over cellular

## 6.1 PoC Architecture

Open Mobile Alliance (OMA) PoC standard Release 1 architecture is based on PoC clients, PoC application server and PoC XML Document Management Server (XDMS). XDMS can be considered as a means of application configuration setting management as it stores application-specific configuration settings. An XDMS server that stores PoC-specific data is called a ‘PoC XDMS’. Using XCAP based reference point POC-8 (see Figure 6.3), the PoC server can fetch PoC-related documents (e.g., access lists) and using XCAP based reference point POC-5 (see Figure 6.3), the PoC server can fetch generic lists (e.g., members of myfriends@example.com list) from a shared XDMS. The PoC servers handle application-specific tasks such as talk burst control (the reservation of talk burst for one user at a time) and PoC session control. They also provide interfaces to the



**Figure 6.3** Push to talk Over Cellular architecture

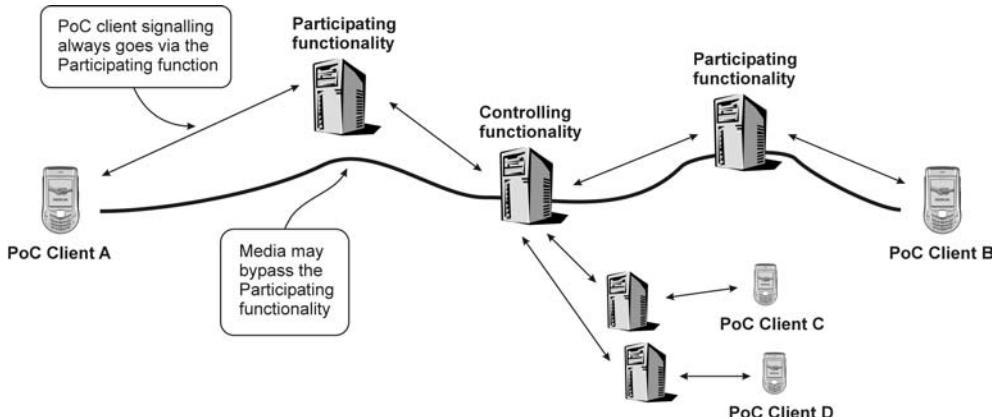
operator's provisioning and network management systems and create application-specific Charging Detail Records (CDRs). The PoC server is connected to the IMS via the IMS Service Control reference point. The IMS takes care of common functions, such as user authentication for PoC, session routing and generic charging based on SIP. PoC client is usually software in the User Equipment (UE) but it could be an application (also in the PC).

Usually, a presence service is associated with PoC, as presence adds value to PoC (e.g., users are able to learn another user's willingness and availability for PoC communication). Even though the PoC service works without presence, it is still shown here in the architecture (Figure 6.3).

#### 6.1.1 PoC Server

A PoC server is an application server in the IMS architecture that provides the PoC service for users. It controls the PoC session setup procedure, enforces policy defined for PoC group sessions (e.g., who is allowed to join, who is allowed to invite more members, decides whether a session should be released when a particular user leaves, decides whether other users should be invited when a particular user joins), provides information about group users (e.g., informs when somebody joins in or leaves a group). Furthermore, the PoC server takes care of media distribution and adaptation, if needed. Moreover, it acts as a control point for talk bursts – i.e., the PoC server decides who has the right to send media, informs other users that permission to send media has been granted to somebody, etc. This mechanism is called 'Talk Burst Control' (further described in Section 6.3.1). So, in short, a PoC server handles both control-plane and user-plane traffic associated with the PoC service, and for this purpose it uses IMS reference points ISC and Mb.

In OMA two different PoC server roles have been defined: participating PoC function and controlling PoC function. The assignment of the PoC server role takes place during a PoC session setup in such a way that there will be only one PoC server performing the controlling PoC function and two or more PoC servers performing the participating PoC function depending on the number of PoC session participants. In the case of a one-to-one



**Figure 6.4** PoC server architecture

PoC session and an ad hoc PoC group session the PoC server of the inviting user performs the controlling PoC function. In the case of a chat PoC group and pre-arranged group session the PoC server owning/hosting the group identity performs the controlling PoC function [OMA PoC AD].

SIP signalling from PoC clients always goes first of all to a participating PoC function, which sends SIP signalling traffic further on towards a controlling PoC function. In contrast, PoC clients may have a direct media and media signalling connection to the controlling PoC function. Figure 6.4 shows the architecture. The distribution of the different functions of a PoC server is summarized in Table 6.1.

### 6.1.2 PoC Client

The PoC client according to the OMA standard is a functional entity on the UE that is able to register itself to IMS using a PoC feature tag and, indicating a PoC release version in SIP REGISTER request, initiates/modifies/releases PoC sessions, supports user-plane procedures (e.g., sending and receiving talk bursts to/from PoC server, supports talk burst control mechanisms, user-plane adaptation), supports the capability to set PoC service settings and receives instant personal alerts.

## 6.2 PoC Features

### 6.2.1 PoC Communication

PoC supports various types of communication models to meet the differing needs of group communication. The main differences between these models relate to group policy and session setup. In other words, how do users create a group and add/delete group members? How do they activate a group session and how is access control arranged? In dial-out group communication, a user invites a group of users to participate in a group session. The invited users receive an indication of the incoming session and they may join in using either an automatic or manual answer. The invited group may be a pre-arranged

**Table 6.1** PoC server functional distribution

Function	Role
SIP session handling	Both
Provides for privacy of the PoC Addresses of Participants	Both
Supports User Plane adaptation procedures	Both
Support Talk Burst Control Protocol negotiation	Both
Provide transcoding between different codecs (optional)	Both
Provides policy enforcement for participation in Group Sessions (e.g. who is allowed to join)	Controlling
Provides the Participants information	Controlling
Provides the centralized media distribution	Controlling
Provides the centralized Talk Burst Control functionality including Talker Identification	Controlling
Provides centralized charging reports	Controlling
Collects and provides centralized media quality information	Controlling
Provides policy enforcement for incoming PoC Session (e.g. Access Control, Incoming PoC Session Barring, availability status, etc)	Participating
Stores the current Answer Mode, Incoming PoC Session Barring and Incoming Instant Personal Barring preferences of the PoC Client	Participating
Provides the Participant charging reports	Participating
Provide filtering of the media streams in the case of Simultaneous PoC Sessions (optional)	Participating
Provide the Talk Burst Control message relay function between PoC Client and PoC Server performing Controlling PoC Function (optional). Note. If the participating PoC server stays on media path then this is mandatory function	Participating

PoC group or a list of users selected from the calling user's phone book as a temporary arrangement (so-called ad hoc PoC group). In the latter case, in particular, the ability to see other users' availability, or presence statuses, before making the dial-out session brings clear additional value for the user. A dial-out session suits unplanned situations or cases where the participants must be handpicked.

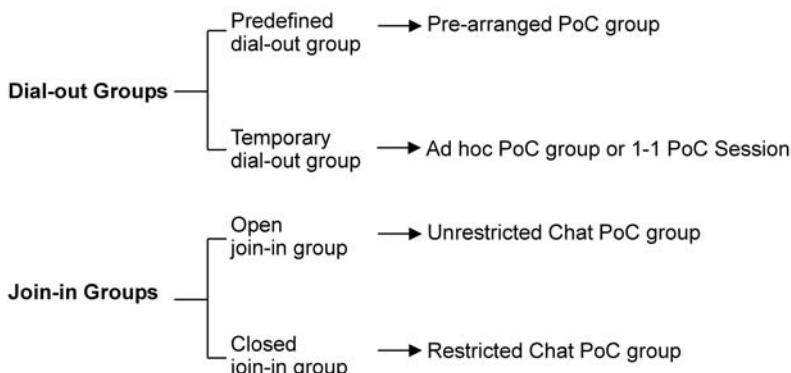
There are some special rules for pre-arranged PoC groups (defined in OMA). First, a PoC session between group members is established when any individual member of the same pre-arranged PoC group invites other members of the group to join in. Second, the communication starts after the first pre-arranged group member accepts the invitation and the controlling PoC server grants media permission to the initiator of the pre-arranged group. Third, participation in a pre-arranged PoC group session is only allowed for pre-defined members (i.e., members of the pre-arranged group) [OMA POC RD]. Similarly, there are some rules for ad hoc PoC groups as well. An ad hoc PoC group is created when a PoC user invites one or more users to a PoC session (note that a one-to-one PoC session is an ad hoc session having two participants). Only those users that have been invited to the ad hoc PoC session are allowed – i.e., a user must have received a request to join (such as a SIP INVITE or SIP REFER from the controlling PoC server) the ad hoc group from one of the current members of the ad hoc PoC group. A local policy at the controlling PoC server may only allow the ad hoc PoC group initiator to add more users (for charging reasons) [OMA POC RD].

In join-in group communication, chat group, the participants themselves explicitly join a PoC group session for communication. In this way users have full control over which groups they participate in. They will never receive any traffic unless they have joined the group. Join-in operation is well suited for communication during any routine or pre-planned activities. Participation in a chat PoC group is analogous to real-life activities such as watching TV, going to a movie or participating in a meeting. Chat PoC sessions can go on for hours, with actual communication comprising only a small portion of the total session time. So, being in a chat PoC session should not prevent a user receiving other sessions in the meantime. The user should also be able to participate in several chat sessions simultaneously (e.g., ‘my family’, ‘basketball friends’, ‘beer buddies’). This requires simultaneous session support.

A chat group can be an unrestricted group without any access control or a restricted group with a list of members. Unrestricted groups are open to anyone who knows the group identification (SIP URI of the group). The group identification can be found, for instance, on an operator portal or chat room. Unrestricted PoC chat groups are suitable as open discussion forums on general and specific topics (e.g., fishing, cars, football). Restricted groups are groups where access is limited to pre-defined users. For restricted groups where access control methods are used, see Section 6.2.4. To join a restricted group, users need to know the group identification (SIP URI of the group) and they need to have the right to join the group session. Restricted PoC chat groups are best suited to the needs of business users where continuous communication within secure groups is needed to support daily work. Figure 6.5 summarizes the different PoC communication models.

### 6.2.2 Simultaneous PoC Sessions

Compared with the traditional telephony service, PoC offers the capability to participate in more than one PoC session at the same time without placing any of the sessions on hold. This capability is called a ‘simultaneous PoC session functionality’. For instance, a user, Alice, could have the following functionality on her PoC device: it automatically joins in with those groups that Alice has pre-configured when the device is turned on. Let’s assume



**Figure 6.5** Different PoC communication models

that Alice has three groups that she actively follows: ‘my family’, ‘work colleagues’ and ‘basketball team’. After joining the groups Alice is able to send and receive media from any of them. This functionality has a clear benefit over the single-session model, as Alice does not need to guess which group is active in a certain time period. Moreover, this model also allows users to hang on in chat groups and still be able to receive one-to-one PoC sessions with other users.

When Alice wants to speak she just chooses the correct group and presses the PoC button. Receiving media from the network is a bit more challenging in that it requires support from Alice’s PoC server. The PoC server needs to filter incoming PoC traffic if there is incoming media in more than one PoC session in which Alice is participating at the same time. The participating PoC function filters the traffic so that Alice hears a single conversation. The following rules govern traffic filtering (in order of preference):

- The user can lock themselves into a single group. Only traffic from that group will be delivered to the user (analogous to what happens in a telephony service).
- The user can set one of the sessions as a primary PoC session. Traffic in the primary session is delivered to the user (if the user is not currently speaking in the secondary session), although there is ongoing traffic in a secondary PoC session.
- Among secondary PoC sessions, the traffic of ongoing conversation is delivered as long as the conversation remains active. After a silent period (length is defined by the operator), the PoC server selects active media from another session.

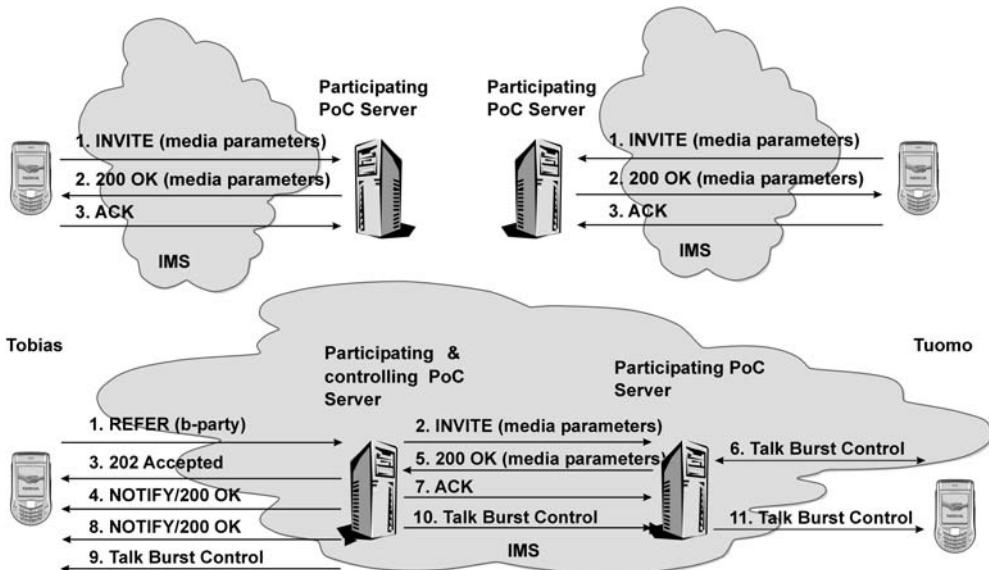
Our example user, Alice, would select ‘my family’ as a primary PoC group and the rest of the groups would then be automatically classified as secondary groups. Now she will hear media from her family members whenever they have something to say. To set up the ‘my family’ group as a primary session, her device needs to include value ‘1’ in a specific attribute,  $a = poc\_sess\_priority = '1/0'$  in the SDP payload of the session request (SIP INVITE, UPDATE or RE-INVITE). To switch the priority value, ‘0’ can be used or the device can indicate priority ‘1’ to other groups and, then, the PoC Server is assumed to swap the priorities.

When Alice wants to concentrate on single-group communication, her device needs to include value ‘1’ in a specific attribute,  $a = poc\_lock = '1/0'$  in the SDP payload of the session request (SIP INVITE, UPDATE or RE-INVITE). To unlock the session, her device needs to assign locking to other sessions or send another session request and indicate value ‘0’ [OMA POC Control Plane].

### 6.2.3 PoC Session Establishment Models

In this section different session establishment models are explained. The reader is recommended to read the section on PoC communication (see Section 6.2.1) before exploring this section.

Two different session models exist: on-demand and pre-established session. The main difference between session models is media parameter negotiation. In a pre-established session model, a user establishes a session towards her participating PoC function and negotiates all media parameters prior to making requests for PoC sessions to other PoC users. In an on-demand model, the ‘normal’ SIP method is used – i.e., media parameters

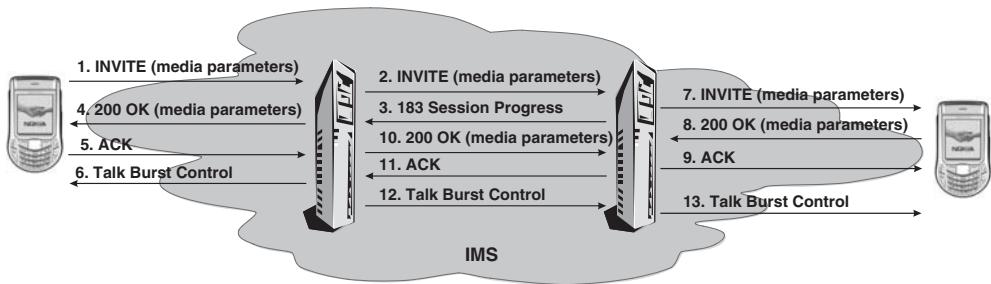


**Figure 6.6** Pre-established PoC session setup

are negotiated when a user makes a request for a PoC session. The pre-established session model allows a PoC client to invite other PoC clients or receive PoC sessions without negotiating the media parameters again. This will bring additional savings in session setup time. Figure 6.6 shows a simplified example of pre-established session usage.

In the upper part of Figure 6.6, PoC clients establish pre-established sessions. In the lower part of the figure, Tobias wants to contact a user named Tuomo. Tobias's PoC client issues a SIP REFER request that contains Tuomo's identity as a target user. Tobias's PoC server will play the role of controlling PoC function, in addition to participating as the PoC function, and sends a SIP INVITE request via IMS towards the terminating network. Tuomo's PoC server (participating) gets the SIP INVITE request and accepts the session immediately, as Tuomo is using a pre-established session and has set his answer mode to auto-answer. Tuomo's PoC client receives a talk burst control message from his PoC server (participating). The talk burst message indicates that a call is coming from Tobias. Tuomo's PoC client acknowledges this message. When the controlling PoC function gets a 200 OK, it will send a talk burst control message to Tobias indicating that permission to speak has been granted. Similarly, the controlling PoC function sends a talk burst control message to Tuomo's PoC client indicating that permission to send media has been granted to Tobias.

Figure 6.7 shows a PoC session setup in which an on-demand PoC session setup model is used. When Tobias wants to have a PoC session with Tuomo his device generates a SIP INVITE request including the device's media capabilities and media transport information. Tobias's PoC server will play the role of controlling PoC function, in addition to participating as the PoC function, and sends a SIP INVITE request via IMS towards the terminating network. Tuomo's PoC Server (participating) gets the SIP INVITE request and becomes aware of Tuomo's answer mode setting. In this case Tuomo had earlier



**Figure 6.7** On-demand PoC session setup using an unconfirmed mode in the terminating network

indicated auto-answer mode and, therefore, his PoC server returns a 183 Session Progress SIP response back to the controlling PoC server. This response contains the information that Tuomo is using auto-answer mode. At the same time, Tuomo's PoC server sends a SIP INVITE request to Tuomo's device. After receiving the INVITE request Tuomo's device automatically responds with a 200 OK SIP message. When Tuomo's PoC server gets the 200 OK response it sends a 200 OK response towards the originating network. At the time of receiving the 183 Session Progress response the controlling PoC function sends a 200 OK response to the initiator (Tobias). The controlling PoC function may grant permission to speak (talk burst control message in Figure 6.7) to the originating user (Tobias) after receiving the 183 Session Progress response if it is willing to buffer media streams. Buffering is needed as the controlling PoC function is not allowed to forward media to recipient participating PoC function(s) before they have responded with a 200 OK SIP final response. If the controlling PoC function is unwilling to buffer media it grants permission to speak to the originating user when the first 200 OK SIP response is received.

In addition to the two examples described above, there are a number of different combinations as the originating user may use either the on-demand or pre-established model and similarly the terminating user may use either on-demand or pre-established models. In addition, the calling user is able to request the called user's PoC device to answer the incoming session automatically – i.e., the inviting PoC user's speech is immediately audible at the called PoC user's device without any action by the called PoC user. This feature is called as 'the manual answer override feature'. Authorization of the manual answer override takes place in the terminating user's network – i.e., users must set proper authorization rules if they want to allow somebody to use the manual answer override. Details on how users can request a manual answer override were still under discussion at the time of writing.

Moreover, the terminating user may use either a manual or automatic answer mode. Table 6.2 shows possible PoC session combinations.

#### 6.2.4 Incoming PoC Session Treatment

At the beginning of the chapter it was stated that push to talk speech is usually connected without the need for the recipient to answer and is heard through the PoC device's built-in loudspeaker. This occurs when a user has set her terminal to an auto-answer mode, the caller is included in the user's access control list and the caller has a specific value in

**Table 6.2** Summary of different PoC session setup combinations

Originating side	Terminating side
Pre-established session	Pre-established session (auto-answer)
Pre-established session	Pre-established session (manual-answer)
Pre-established session	On-demand session (auto-answer)
Pre-established session	On-demand session (manual-answer)
On-demand session	On-demand session (auto-answer)
On-demand session	On-demand session (manual-answer)
On-demand session	Pre-established session (auto-answer)
On-demand session	Pre-established session (manual-answer)

the access control list. In this section we explain how this can be done and what other mechanisms are in place to control incoming PoC sessions.

Let's start with answering modes. Two different answer modes have been defined: auto-answer and manual answer mode. When auto-answer mode is turned on, the PoC device will accept the incoming PoC sessions without waiting for any specific actions from the user – i.e., incoming media streams can be played immediately. Manual answer mode is an ordinary mode in which a user needs to accept an incoming PoC session request before the PoC device confirms the acceptance of an incoming PoC session to the PoC server. The answer mode used is also signalled back to the PoC server performing the controlling PoC and, further, to the originating user function using Unconfirmed<sup>1</sup> or Confirmed<sup>2</sup> indications in the SIP response. A Confirmed indication is given when the terminating PoC device is using manual answer mode and has accepted a new PoC session by sending a SIP 200 OK response. Otherwise, an Unconfirmed indication is generated by the terminating participating PoC server function. Section 6.4 describes how a PoC client informs the PoC server about the desired answer mode.

Using auto-answer mode when you are available would be a nice and useful feature when you can be sure that the calling user is going to behave correctly – e.g., the caller would not say something improper. However, users cannot be sure who might call them and, therefore, they may not feel comfortable to use auto-answer mode for all possible users. On the other hand, using manual answer mode all the time is not convenient either. Moreover, a user may want to automatically decline PoC sessions from certain users or PoC groups.

In order to overcome these types of issues, the access control mechanism was developed. Access control means that a user can:

- allow or block, selectively, incoming PoC sessions from other PoC users and PoC groups; and
- define, selectively, those users whose PoC sessions are to be accepted automatically.

<sup>1</sup> An Unconfirmed indication is an indication returned by the PoC server to confirm that it is able to receive media and believes the PoC client is able to accept media; the PoC server sends the Unconfirmed indication prior to determining whether all egress elements are ready or even able to receive media.

<sup>2</sup> A Confirmed indication is a signalling message returned by the PoC server to confirm that the PoC server, all the other network elements intermediary to the PoC server and a terminating PoC client are able and willing to receive media [OMA PoC Control Plane].

Access control is executed at the PoC server performing the participating role for the called PoC user. In order to execute access control a PoC user needs to create an access control list document and send it to the PoC XDMS. The way to do this is further described in Section 5.7.1.2. In addition, a PoC user is able to bar all incoming PoC sessions by activating Incoming PoC Session Barring (this procedure is covered in Section 6.2.7).

The result of answer mode and access control list usage at the PoC server and PoC client is perhaps best illustrated by a simple example. Tobias has just bought a new PoC-enabled device, as most of his friends are using them. Tobias's device is turned on and it is registered to the IMS. First, Tobias opens a window and sets up an access control list for his buddies. He decides to include Theresa, Tuomo and Matias as his closest friends and, therefore, sets their identities in the 'accept' category in his access control list. At the same time, he inserts John's identity in the 'reject' category as he does not want to receive any sessions from him. After completing the access control list, Tobias accepts the list and his device uploads the document to the PoC XDMS. Second, Tobias chooses to set his answer mode to auto-answer mode. This information is then sent to Tobias's PoC server by his device. Now that all the necessary settings are done, Tobias is ready to make use of his settings.

After a short while (maybe after sending an instant personal alert) Tuomo tries to reach Tobias. A one-to-one PoC session request is sent from Tuomo's device towards Tobias's device. When it reaches Tobias's PoC server, the PoC server will fetch the access control list from the PoC XDMS and discover that Tuomo is included in the 'accept' category in the access control list and that Tobias has indicated a willingness to accept incoming PoC sessions automatically; therefore, the PoC server decides to let this PoC session go through. When the PoC server sends the incoming PoC session request to Tobias's PoC device it includes an auto-answer indication in the request. Based on the auto-answer indication, Tobias's phone accepts the PoC session, activates the loudspeaker and waits for the incoming PoC speech stream.

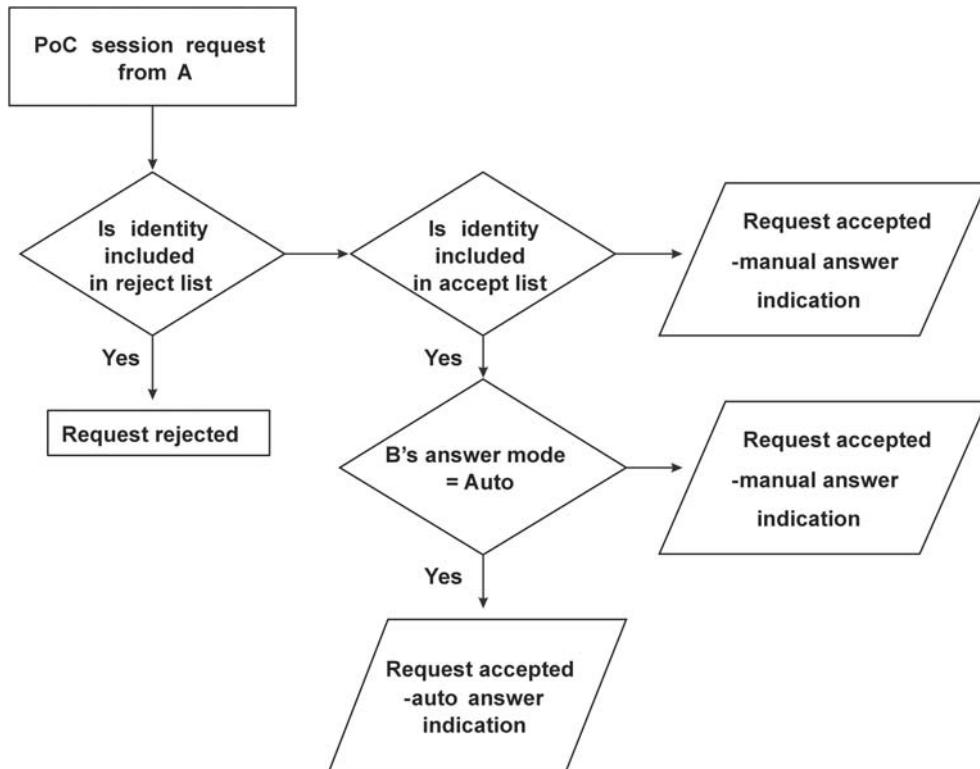
Later on the same day, one of Tobias's friends, Marja, who has subscribed to Tobias's presence information, realizes that Tobias's presence information shows PoC availability. After realizing this, Marja attempts to reach Tobias. When the PoC session request reaches Tobias's PoC server, this time the PoC server is unable to find Marja in the access control list. Nevertheless, Tuomo has indicated a willingness to accept incoming PoC sessions automatically, so the PoC server decides to accept the incoming PoC session but, compared with the previous case, it will add a manual answer indication in the request. Based on the manual answer indication Tobias's phone will not accept the PoC session immediately; instead it alerts Tobias and displays that the caller is Marja.

When John tries to reach Tobias, the PoC server will realize that John is included in the reject category and it will automatically reject the incoming PoC session from John. This time the PoC session request does not reach Tobias's device at all.

Figure 6.8 shows incoming session treatment using access control lists and answer modes.<sup>3</sup>

---

<sup>3</sup> If a manual answer override is requested and the calling user is allowed to use this feature, then an automatic answer mode is applied. If the request is not allowed the answer mode selection follows Figure 6.8.



**Figure 6.8** Incoming session treatment decision tree showing impact of access control list and user's answer mode

#### 6.2.5 Instant Personal Alerts

Sometimes a caller user is not able to reach a recipient (e.g., the user has Incoming Session Barring activated), then a caller may wish to leave an explicit message to request the called party to call back. This message is called an ‘instant personal alert’. The instant personal alert request can also be used instead of a push to talk voice transmission when a less noticeable method of alerting is desired.

Whenever users want to send an instant personal alert request to another user, users select the target (e.g., from the phonebook) and send the instant personal alert request. For this purpose a SIP MESSAGE method is used. From a recipient point of view there is a need to differentiate between an ordinary SIP instant message and PoC instant personal alert, as both use the SIP message method. Differentiating PoC instant personal alerts makes it possible to create automatic operations at the PoC client. For this purpose the originating PoC client is mandated to include an Accept-Contact header with the PoC feature tag, ‘+g.poc.talkburst’ along with ‘require’ and ‘explicit’ parameters:

```

MESSAGE sip:bob@example.com SIP/2.0
From: <sip:tobias@home1.fr>;tag=31415
To: <sip:bob@example.com>
  
```

```
Accept-Contact: *;+g.poc.talkburst;require;explicit
User-Agent: PoC-client/OMA1.0
```

If a user does not want to receive instant personal alerts, then they can activate Incoming Instant Personal Alert Barring. When this service is activated, the participating PoC server serving the user is mandated to block the delivery and to return a SIP 480 Temporarily Unavailable error response to the sender [OMA POC Control Plane].

### 6.2.6 Group Advertisement

When a PoC user, Tobias, wants to plan his summer vacation to Finland he decides to create a chat PoC group with his sister Theresa and his Finnish friends Tuomo and Matias. After creating a group, summervacation@poc.example.com, he decides to advertise the created group to Theresa, Tuomo and Matias. To do this, Tobias sends a group advertisement that contains all the necessary information about the created group: name of the group (URI and display name), type of group, advertisement text. When the other group members receive a group advertisement, they may decide to store the contact information for further communication. Using the received information they could immediately decide to dial into the group.

A group advertisement could be sent to one or more users or it could be sent to all group members at once using a SIP MESSAGE that has a PoC-specific content in the form of a Multipurpose Internet Mail Extension (MIME) vnd.poc.advertisement+xml body. Additionally, a sender must include an Accept-Contact header with the PoC feature-tag '+g.poc.groupad' along with the 'require' and 'explicit' parameters. Using a feature tag and the above-mentioned parameter guarantees that the advertisement is only delivered to PoC-enabled UEs that are able to understand the group advertisement message.

Our example group advertisement, as sent by Tobias's PoC device, looks like:

```
MESSAGE sip:summervacation@poc.home1.fr SIP/2.0
From: <sip:tobias@home1.fr>;tag=31415
To: <sip: summervacation@poc.home1.fr >
Accept-Contact: *;+g.poc.groupad;require;explicit
User-Agent: PoC-client/OMA1.0
Content-Type: application/vnd.poc.group-advertisement+xml
Content-Length: (482)

<?xml version="1.0" encoding="UTF-8"?>
<group-advertisement
  xmlns="urn:oma:params:xml:ns:poc:group-advertisement"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:oma:params:xml:ns:poc:
    group-advertisement">
  <note>Tobias summer vacation planning</note>
  <group type="dialed-in">
    <display-name>Great trip to Finland</display-name>
    <uri>sip:summervacation@poc.example.com</uri>
  </group>
</group-advertisement>
```

Group advertisement XML documents are based on XML 1.0 and use UTF-8 encoding. The namespace URI for elements defined by an OMA PoC control-plane document is a URN, using the namespace identifier ‘oma’. This URN is: urn:oma: params+xml:ns:poc: groupadvertisement. The key PoC elements in the group advertisement XML document are:

- <note> element – advertisement text added by a sender;
- <group> element – contains information about type of group (dialled-in, dialled-out, other), display name of the PoC group, and URI of the PoC group.

When ‘dialled-in’ is used in the type attribute it indicates a chat PoC group. Similarly, ‘dialled-out’ means a pre-arranged PoC group. ‘Other’ is added for future expandability [OMA POC Control Plane].

Support of this feature is optional for both PoC clients and PoC servers in OMA PoC specifications. Controlling PoC servers may support delivery of Group Advertisement messages to all PoC group members and may apply different authorization rules that stipulate who is allowed to send group advertisement information to all group members. The most obvious authorization rules are:

- only the group owner is allowed to send Group Advertisement information to all group members;
- all group members are allowed to send Group Advertisement information to all group members.

If the PoC client does not want to receive group advertisement messages then it shall not include a PoC feature-tag ‘+g.poc.groupad’ in the SIP REGISTER request.

### *6.2.7 Barring Features*

A PoC user can selectively block PoC incoming sessions using an access control list, as explained in Section 6.2.4. In addition, a PoC user is able to instruct a PoC server to reject all new incoming PoC sessions. This feature is called Incoming Session Barring (ISB). When barring is activated, the participating PoC server will reject all incoming session requests with a SIP 480 Temporarily Unavailable response and, at the same time, the participating PoC server will keep ongoing PoC session(s) intact and deliver incoming Instant Personal Alerts to the user. If the user wants to block Instant Personal Alerts as well, then they need to activate Instant Personal Alert Barring. Section 6.4 shows how barring can be activated and de-activated.

### *6.2.8 Participant Information*

A PoC user can request information about PoC session participants and their status in the PoC session. The PoC client uses the conference-state event package to learn about changes in PoC participants: in other words, a user can learn, through notifications, who has joined or left a conference. This event package also allows participants to learn the status of a user’s participation in a conference (on-hold, alerting).

Users can subscribe to a conference state by sending a SIP SUBSCRIBE request to the conference URI that identifies the controlling PoC server. The controlling PoC server acts as a notifier for this event package.

The name of this event package is ‘conference’. This token appears in the Event header of the SUBSCRIBE request. The body of a notification carries the conference-state information document in the MIME-type ‘application/conference-info+xml’ as defined in RFC4575. The PoC Release 1.0 specification uses only a limited subset of offered functions as follows:

- PoC group identity;
- PoC user identities are now part of the PoC session;
- PoC user status (connected, disconnected, on-hold, alerting).

## 6.3 User Plane

The user plane of PoC includes three components, as illustrated in Figure 6.9: the media flow (consisting of talk bursts), talk burst control and quality feedback.

The participating PoC server relays talk bursts, talk burst control messages and quality feedback measurements back and forth between the controlling PoC server and the PoC client, except in the following scenarios:

- the PoC client has a pre-established session with the PoC server;
- the PoC client and PoC server support simultaneous sessions;
- the PoC server needs media transport logging to support charging functionality;
- the PoC server is used for transcoding services or other media adaptations;
- the PoC server needs to support lawful interception;
- the PoC server acts as a gateway between two different talk burst control protocols.

The PoC server – in performing both the participating and controlling function – acts as an RTP translator. A translator, as opposed to an RTP mixer, only passes media packets from one IP flow to another, without re-packetizing or alteration to the media format.

### 6.3.1 Talk Bursts

A talk burst is simply a single burst of media flowing from a participant to the controlling PoC server. The controlling PoC server distributes the talk burst to all of the participants of the session. The participants receive the media flow, and render it via their sound systems and play it out from their loudspeakers. The PoC session is a shared floor,

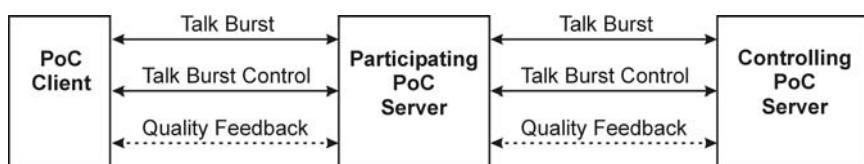


Figure 6.9 User plane Protocol entities

which means that only one single talk burst at a time can successfully be distributed and played out at the receiving end. In addition, since in a group PoC session there may be a number of participants to a single floor, there needs to be an arbitrator function – a floor moderator that controls who is allowed to send media at any given moment. Requests to hold the floor may also be queued and granted in succession. Alternatively, a request is either granted or denied, and the user will usually discover this via some visual or sound indication – just like when using a regular walkie-talkie device.

### 6.3.2 Talk Burst Control

For floor control, the PoC specifications define the Talk Burst Control Protocol (TBCP) as part of the user-plane specifications. TBCP is used to request, grant, deny and release the PoC session floor. The floor moderator, or the TBCP server, is always located in the controlling PoC function. The participating PoC server usually only relays the messages back and forth to the PoC client, but may also perform additional functions, as we describe later in this chapter. TBCP includes the following messages:

1. TBCP Talk Burst Request (TB\_Request) – this message is sent by the clients to request permission to send a talk burst. Such a message may be generated for instance when a user presses the send key in order to speak to the PoC session.
2. TBCP Talk Burst Granted (TB\_Granted) – this message is sent by the PoC server to the PoC client to notify that the client has been granted the floor. This means that the client is allowed to send a talk burst that will be heard by the other participants in the PoC session.
3. TBCP Talk Burst Deny (TB\_Deny) – this message is sent by the PoC server to the PoC client to announce that the client has been denied the floor. This means that the client is not allowed to send a talk burst and one would not be heard by the other participants at this time.
4. TBCP Talk Burst Release (TB\_Release) – this message is sent by the PoC client to the PoC server to announce that the client has completed sending its talk burst.
5. TBCP Talk Burst Taken (TB\_Taken) – this message is sent by the PoC server to all of the other PoC session participants in order to inform them that another PoC client has gained possession of the floor and is in the process of sending a talk burst. This message can also be sent reliably by the PoC server, requesting that an immediate acknowledgement be sent by the recipient of the message.
6. TBCP Talk Burst Revoke (TB\_Revoke) – this message is used by the PoC server to revoke a grant to speak. For example, the server may interrupt a talk burst that is overly long.
7. TBCP Talk Burst Idle (TB\_Idle) – this message is sent by the PoC server to all of the PoC session participants in order to inform them that the floor is open and that the server is willing to accept TBCP Talk Burst Request messages.
8. TBCP Talk Burst Acknowledgement (TB\_Ack) – this message is sent by the PoC client to the PoC server upon request of an acknowledgement in the corresponding request.
9. TBCP Talk Burst Queue Status Response (TB\_Queue) – this message is sent by the PoC server to the PoC client informing it that the talk burst request has been queued. The message may also indicate the current position of the PoC user in the queue.

10. TBCP Talk Burst Queue Status Request (TB\_Position) – this message is sent by the PoC client to the PoC server to request its current position in the talk burst or floor queue.
11. TBCP Connect (Connect) – in a pre-established session, this message is sent by the PoC server to the PoC client to indicate that a new session has been received and that a corresponding floor has thus been created.
12. TBCP Disconnect (Disconnect) – in a pre-established session, this message is sent by the PoC server to the PoC client to indicate that a particular session has ended and the corresponding floor destroyed.

The TBCP shares protocol formatting with the RTCP – in fact, the protocol utilizes the RTCP APP packet, as illustrated in Figure 6.10, for transmitting control messages. Since RTCP operates over an unreliable transport – the User Datagram Protocol (UDP) – the protocol uses a request–reply scheme, with application level retransmissions based on specific timers. In addition, the message can be sent reliably – i.e., immediate acknowledgement can be requested by the sender.

The Version (V) and Padding (P) fields are identical to other RTCP packets. Specifically, the version field is set to 2 (the current RTP version), and the padding field is a boolean, indicating whether or not any padding was added at the end of the RTCP packet.

The Name field identifies the application; in TBCP, the Name field is set to ‘POC1’, indicating that the APP packet is part of PoC Release 1. The Subtype field identifies the TBCP message type. In Table 6.3 the Subtype bit pattern is mapped to the corresponding TBCP message. The SSRC field identifies the source of the TBCP request. In case the sender is the PoC server, it contains the controlling PoC server’s identity, except for TB\_Connect and TB\_Disconnect messages, which are sent by the participating PoC server.

### 6.3.3 Quality Feedback

The PoC server and client can optionally create, send and process RTCP quality feedback reports. The receivers of RTP can generate reception quality feedback that can take one of two forms: Sender Reports (SRs) and Receiver Reports (RRs). The difference between an RR and an SR is that the SR also includes information about the data sent; both reports include statistics about received media packets, including total number of RTP packets, total number of octets of data received and interarrival jitter, approximating the relative transit time of the media packets.

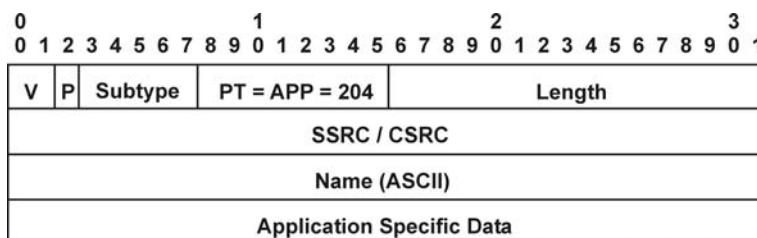


Figure 6.10 RTP control Protocol APP packet format

**Table 6.3** Mapping of subtype bit patterns to TBCP Protocol messages

Bit Pattern	TBCP Message
00000	TB_Request
00001	TB_Granted
00010	TB_Taken (TB_Ack requested)
00011	TB_Deny
00100	TB_Release
00101	TB_Idle
00110	TB_Revoke
00111	TB_Ack
01000	TB_Position
01001	TB_Queue
01010	<i>Reserved for future use</i>
01011	TB_Disconnect
01100	<i>Reserved for future use</i>
01101	<i>Reserved for future use</i>
01110	<i>Reserved for future use</i>
01111	TB_Connect
10010	TB_Taken (No TB_Ack requested)

In the PoC user-plane, a PoC client is instructed to generate an SR every time it has completely sent a talk burst – i.e., after having sent the TB\_Release message to the PoC server. The PoC client is instructed to send an RR when:

- The PoC client receives an SR.
- The PoC client receives an indication that a talk burst has ended (i.e., the TB\_Idle message).
- The PoC client's RTP media timer fires.

As PoC servers are acting as RTP translators in PoC sessions, they generally only relay RTCP reports between PoC session participants. This means that SRs are fanned out from a single sender to all others, and similarly, RRs are suppressed from participants that didn't actually receive any media. This can happen if the PoC client is participating in multiple simultaneous sessions, in which case only the primary session is heard if there happens to be conflicting incoming talk bursts.

## 6.4 PoC Service Settings

In Section 6.2 (PoC features) different features were described but it was not shown how answer mode, barring settings or support for simultaneous sessions are activated or deactivated. In OMA PoC Release 1 it was decided that SIP PUBLISH should be used for these service settings. At the time of writing, the required Internet Engineering Task Force (IETF) solution is not ready, therefore this is subject to further change.

The main principle is that after each successful initial registration the PoC client sends SIP PUBLISH that has a PoC-specific content (XML document) in the form of MIME ‘application/poc-settings+xml’ body. Additionally, a sender must include an Accept-Contact header with the PoC feature tag ‘+g.poc.talkburst’ along with the ‘require’ and ‘explicit’ parameters. Using a feature tag and the above-mentioned parameters guarantees that the request is correctly delivered to the user’s PoC server. The XML document itself contains four PoC-specific elements as follows:

```
<isb-settings>; active=true/false  
<am-settings>; manual/auto  
<ipab-settings>; active=true/false  
<sss-settings> active=true/false
```

The example below describes a case where the PoC Client does not activate barrings, answer mode is set to automatic answer, and the device is capable for simultaneous PoC Sessions.

```
PUBLISH sip:tobias@home1.fr SIP/2.0  
From: <sip: tobias@home1.fr>;tag=31415  
To: <sip: tobias@home1.fr>  
Accept-Contact: *;+g.poc.talkburst;require;explicit  
User-Agent: PoC-client/OMA1.0  
Event: poc-settings  
Expires: 7200  
Content-Type: application/poc-settings+xml  
Content-Length: (...)  
  
<?xml version="1.0" encoding="UTF-8"?>  
<poc-settings xmlns="urn:oma:params:xml:ns:poc:poc-settings">  
  <entity id="do39s8zksn2d98x">  
    <isb-settings>  
      <incoming-session-barring active="false">  
    </isb-settings>  
    <am-settings>  
      <answer-mode>automatic</answer-mode>  
    </am-settings>  
    <ipab-settings>  
      <incoming-personal-alert-barring active="false"/>  
    </ipab-settings>  
    <sss-settings>  
      <simultaneous-sessions-support active="true"/>  
    </sss-settings>  
  </entity>  
</poc-settings>
```



# 7

# Messaging

There are currently many forms of messaging services available. In general, messaging entails sending a message from one entity to another. Messages can take many forms, include many types of data and be delivered in various ways. It is usual to have messages carry multimedia as well as text and be delivered either in near-real time as in many instant messaging systems or into a mailbox as in email today. In this chapter we give some details about messaging in the Internet Protocol Multimedia Subsystem (IMS) context.

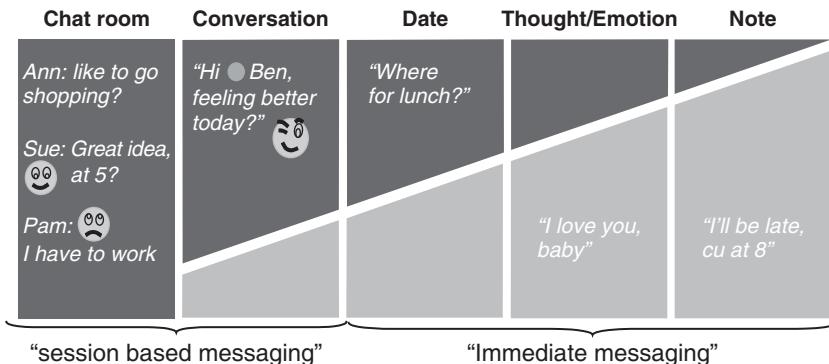
## 7.1 Overview of IMS Messaging

Figure 7.1. reveals two different types of IMS messaging forms: immediate messaging and session-based messaging. Each form of IMS messaging has its own characteristics; so, even though messaging in its simplest form can be thought of as a single service – after all, all forms of messaging are really about sending a message from A to B – the fact that these characteristics differ makes them each a service on their own. However, the way in which applications are built on top of these services may well hide the fact that these are different forms of messaging. In fact, one of the key requirements for IMS messaging is easy interworking between different messaging types.

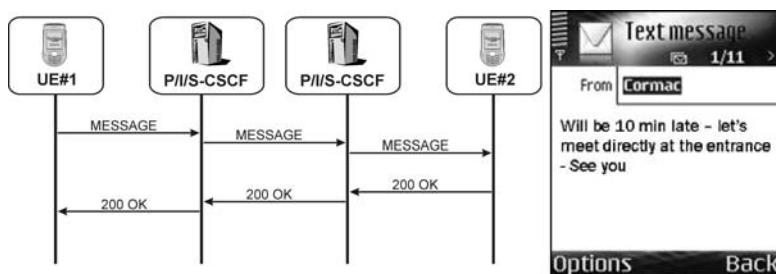
## 7.2 Immediate Messaging

Immediate messaging, or page-mode messaging, is the familiar instant messaging paradigm adopted in the IMS framework. It uses the Session Initiation Protocol (SIP) MESSAGE method to send messages between peers in near-real time. Figure 7.2 illustrates a typical message flow.

In immediate messaging, the User Equipment (UE) simply generates a MESSAGE request, fills in the desired content – which typically consists of text, but can also contain snippets of multimedia such as sounds and images – and populates the request-URI (Uniform Resource Identifier) with the address of the recipient. The request is then routed via the IMS infrastructure similar to the manner used for an INVITE, until the immediate message finds its way to the UE of the recipient user or gets stored in the network.



**Figure 7.1** Instant messaging types



**Figure 7.2** Immediate messaging flow

There might, of course, be a reply to this message; in fact, a full dialogue of immediate messages back and forth between the two users is likely. However, in contrast to session-based messaging, the context of this session only exists in the minds of the participating users. There is no protocol session involved: each immediate message is an independent transaction and is not related to any previous requests.

If an immediate message is received while the IMS subscriber is offline, or in an unregistered state, the MESSAGE will route to an Application Server (AS). The AS can hold the message in storage, and when the user registers, the AS can then deliver the message to its final destination.

Usually, immediate messages are addressed to another peer’s public user identity. However, the user can also send a single message to a number of recipients using a list server extension in the IMS. Basically, an IMS user can create a list, using a SIP address in the form of a Public Service Identifier (PSI), and populate this list with a set of SIP URIs of the intended membership. Whenever a MESSAGE method is sent to the PSI corresponding to this list, the request is routed to the list server. The list server, which is an AS itself, will intercept the message and generate a new request for each of the members of the list.

If an immediate message is targeted to a user who is not an IMS subscriber then it is possible to route the MESSAGE to an Application Server that performs messaging interworking. This AS could for example convert the MESSAGE to SMS or MMS or even email. At the time of writing 3GPP is standardizing MESSAGE to SMS interworking in Release 8 and OMA is standardizing Converged IP Messaging service enabler which is expected to support other types of messaging interworking scenarios. Interworking is explained in more detail in Section 7.4.

### 7.3 Session-Based Messaging

Session-based messaging relates to a familiar paradigm of messaging already in use in the Internet: Internet Relay Chat (IRC) [RFC2810]. In this mode of messaging the user takes part in a session in which the main media component often consists of short textual messages. As in any other session a message session has a well-defined lifetime: a message session starts when the participants begin the session and stops when the participants close the session. After the session is set up – using SIP and SDP between the participants – media then flows directly from peer to peer. Figure 7.3 illustrates the typical message flow of a message session.

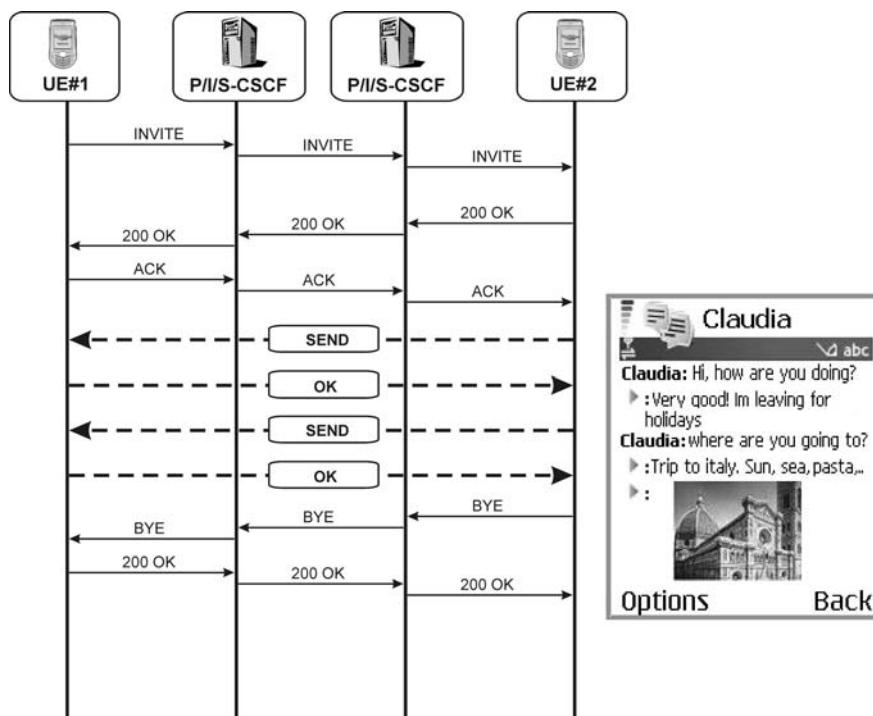


Figure 7.3 Session-based messaging flow

Session-based messaging can be peer to peer, in which case the experience closely mimics that of a normal voice call. An ordinary invitation to a session is received, the only difference being that the main media component is a session of messages. However, this is not an actual limitation to session-based messaging, since it is, of course, possible to combine other media sessions with message sessions. In fact, many useful and exciting applications are enabled by this functionality: for example, video calls with a text side channel might be a valuable application for hearing-impaired people.

The actual protocol for conveying the messages within a session is called the Message Session Relay Protocol (MSRP) [RFC4975]. MSRP layers on top of Transmission Control Protocol (TCP), and can carry any Multipurpose Internet Mail Extensions (MIME) encapsulated data. Messages can be of arbitrary size, since one of the protocol features is the ability to support sending a complete message in small chunks that are automatically reassembled at the recipient end.

Session-based messaging forms a natural unison with conferencing as well. Using the conferencing functionality, session-based messaging can turn into a multiparty chat conference. In this mode of operation, session-based messaging can enable applications similar to modern day voice conferences. A chat conference can also be comparable with a channel in the IRC.<sup>1</sup> A service provider will typically offer the possibility for users to have both private chats, where the set of participants is restricted and public chats, some of which are maintained by the service provider. In this context network typically provides additional functionality called MSRP Switch. It is used to relay messages between participants.

## 7.4 Messaging Interworking

Two messaging technologies are widely deployed in mobile networks namely SMS and MMS. It can be said that SMS meets the following requirements: simplicity, ubiquity, mass penetration, critical mass and reliability. SMS is easy to understand and fairly easy to use, almost anyone who has a mobile phone is equipped with SMS and failures and protracted delays in delivery are now uncommon. Similarly in developed countries the number of connections in mobile networks that are MMS-capable is expected to be above 30 %. Building mass penetration and getting end users to use these standardized messaging technologies has been slow but it turned out to be a huge revenue generator for mobile operators. Instant Messaging solutions (e.g. Yahoo, MSN, AOL, ICQ) have been a major success in Internet (PCs) and Enterprise domains and it is likely to emerge in the mobile domain as well. Business model in Internet messaging has been different compared to SMS/MMS solutions. Usually, service is provided free of charge to the user, interoperability outside the community is often limited and users may get advertisements. It's to be seen whether these business models can mix together. One thing is still sure, IMS messaging solution needs to interwork with most common messaging technologies in order to get wide usage space. Most obvious interworking technology is SMS. 3GPP has taken deliberately actions in this domain.

---

<sup>1</sup>In this respect a channel such as # Helsinki on an IRC server could be simply represented by a SIP URI: sip:helsinki@some.chat.net, or a chat group in a typical Internet chat service.

The first action in 3GPP was to define how to send and receive SMS over the IMS network. This capability is called as SMS over IP. Solution is very straightforward; the actual SMS is attached as a special content type to the SIP MESSAGE method. This would enable delivering SMS to devices that are attached to non-cellular IP Connectivity Access Networks (e.g. WLAN, WiMAX). It could be also seen as an alternative termination bearer to current bearer options (CS, GPRS). Utilizing this type of interworking all kinds of existing value added SMS services can be delivered to users connected to the IMS. Figure 7.4. shows SMS over IP termination flow example. At Step 1 IP-Short-Message-Gateway Application Server receives Short Message (SM) from SM Service Centre and it composes and sends SIP MESSAGE in Step 2. Key fields at the MESSAGE request are: Request-URI that points to a registered IMS Public User Identity, Accept-Contact stating that this request can only be delivered to a device that has registered capability to receive SMS over IP, Request-Disposition having value ‘no-fork’ stating that this request is to be delivered to one and only one UE and Content-Type stating that message payload is actual short message [3GPP TS 24.341]. Figure 7.5 shows an originating SMS over IP procedure. Like in the terminating flow the actual short message is included as the body of the MESSAGE request, but here the target address in SIP level is actually Service Centre (recipient’s address is included in the actual short message).

The second course of action in 3GPP is to specify a native interworking between SMS and SIP based messaging solutions. It means that SMS is fully converted to SIP based request meaning that IMS UE (e.g. PC) does not need to implement SMS stack anymore. This work can be seen as an evolution step from SMS over IP feature. Standardization work is ongoing in 3GPP Release 8 and scope at the time writing was to build interworking between SMS and Open Mobile Alliance SIMPLE IM feature which is described in more detail in the next section.

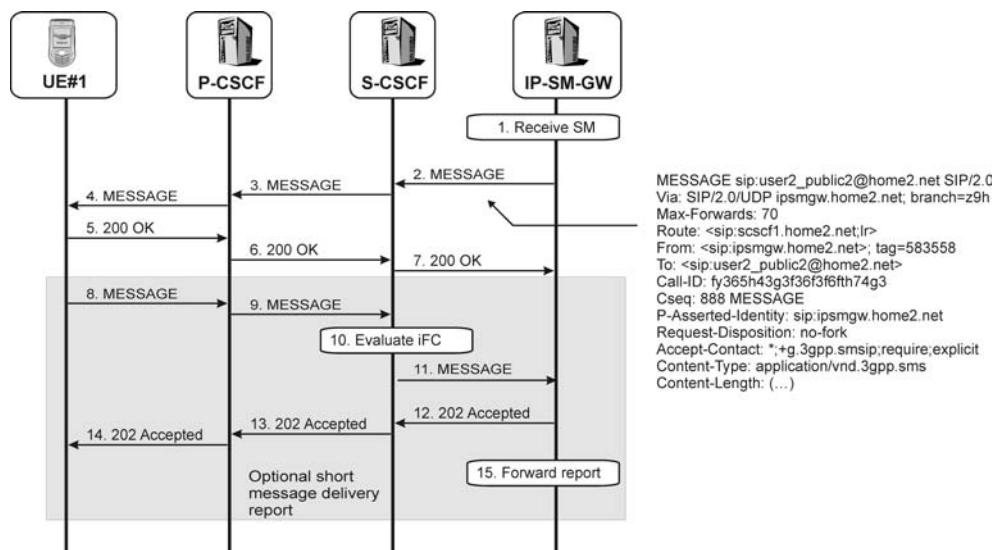


Figure 7.4 Example of terminating SMS over IP

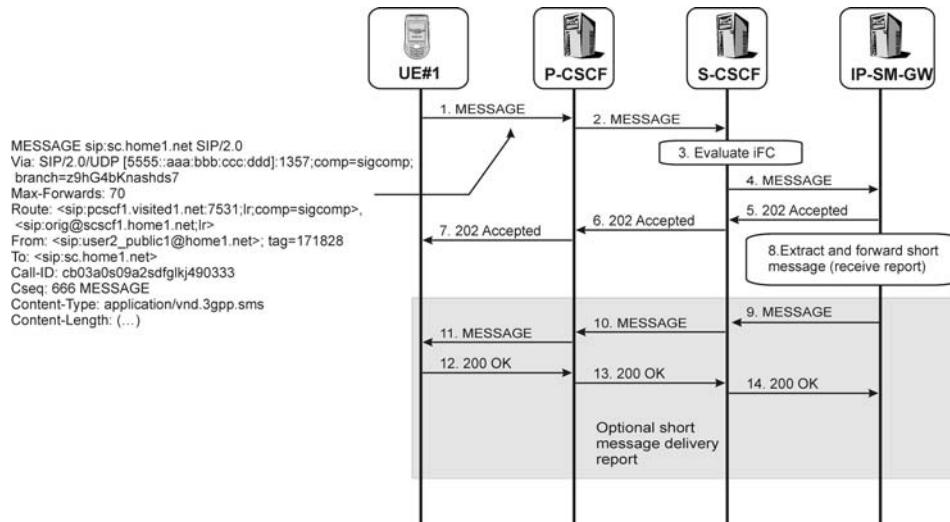


Figure 7.5 Example of originating SMS over IP

For example, a traditional GSM user (+358501112223) sends SMS ‘Will be 10min late – let’s meet directly at the entrance – See you’ to her friend (+358508888888). As the recipient is an IMS user the terminating SMS gets routed to IP-SM-GW which converts the short message to SIP MESSAGE for example as follows:

```

MESSAGE tel:+358508888888 SIP/2.0
Via: SIP/2.0/UDP ipsmgw.home2.net; branch=z9h
Max-Forwards: 70
Route: <sip:scccf1.home2.net;lr>
From: <tel:+358501112223>; tag=583558
To: <tel:+358508888888>
Call-ID: fy365h43g3f36f3f6fth74g3
Cseq: 888 MESSAGE
P-Asserted-Identity: tel:+358501112223
Content-Type: text/plain
Content-Length: (...)

Will be 10min late -let's meet directly at the entrance - See you
```

Key differences compared to MESSAGE request carrying SMS over IP (Figure 7.4) are: the actual message is included in text format, the request shows that request is coming from A-party and no need to set Accept-Contact and Request-Disposition headers. When the recipient gets this request she will not be able to see that the original request was SMS.

Let’s assume that the IMS user responds to this message ‘Ok. I’ll wait for you at the entrance. For doing so the UE simply generates a MESSAGE request, fills in the desired content and populates the request-URI with the address of the recipient. The message looks like this:

```

MESSAGE tel:+358501112223 SIP/2.0
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: <tel:+358508888888>; tag=171828
To: <tel:+358501112223>
Call-ID: cb03a0s09a2sdfglkj490333
Cseq: 666 MESSAGE
Content-Type: text/plain
Content-Length: (...)

Ok. I'll wait for you at the entrance.
```

The message gets routed to IP-SM-GW at the user's home network which realizes that the recipient is not an IMS user and it converts this request to a short message which is sent to a recipient via short message service centre. The GSM user gets this message as an ordinary SMS and does not see difference to any other SMS.

There is one additional factor which should be taken into account in IMS messaging interworking, size of message. SIP MESSAGE RFC [3428] contains a size limitation. The RFC states that it is not allowed to send SIP MESSAGE outside of session if message size is at least 200 bytes less than the lowest MTU value found en route (usually this limit is 1300 bytes). If IP-SM-GW receives concatenated SMS message (group of messages formed of several standard length short messages to be sent together as if they were one longer message) and the size limit of SIP MESSAGE would be exceeded then the IP-SM-GW should use session mode messaging to send the message.

## 7.5 Instant Messaging by Open Mobile Alliance

Let's begin this section with an example scenario to highlight some advanced capabilities of the most comprehensive standard based IMS messaging, OMA IM, can deliver. Tuomo is enjoying a well earned winter vacation in Thailand and he is currently riding on an elephant. He asks his friend to take a photo with his mobile phone. He quickly reviews the photo and decides to send the high quality photo with some sunny greetings from Thailand to his own buddylist. Tuomo's device realized that overall message size is too big to be sent via MMS<sup>2</sup> and Tuomo does not want to use email as it does not reach all his friends as fast he likes. Therefore IM client is invoked and it initiates OMA IM large message mode connection to his IM server at his home network. The IM server makes all necessary service checks and discovers members of Tuomo's buddylist and initiates individual messages to all recipients (Tobias, Marja, Kristiina, John) and stores the sent message to Tuomo's message history file at the network. Tobias and John are currently online and are willing to receive all kind incoming messages so IM server will pass the message to them. Kristiina is tired of getting messages from Tuomo, so she has added Tuomo to her IM blocklist. Therefore, IM server blocks message to Kristiina completely. Marja's battery is flat and she is not logged-in IMS so IM Server takes action and stores the incoming message for later delivery. Once Marja logs in the IM server

---

<sup>2</sup> MMS has a size limitation of 600kbytes which means that pictures taken e.g. with 5 megapixels camera cannot be sent without converting it first to lower resolution.

automatically delivers the stored messages to her. After checking Tuomo's message she decides to check if Tuomo is still online and as he is she sends a chat invitation to Tuomo using the same messaging system that delivered the one-shot message to her earlier ...

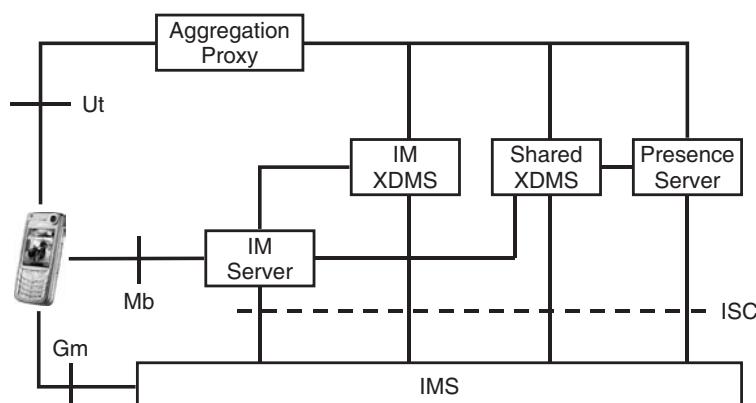
### 7.5.1 OMA IM Architecture

Open Mobile Alliance (OMA) IM standard Release 1 architecture is based on IM clients, IM application server and XML Document Management Servers (XDMS). An XDMS server that stores IM-specific data is called a 'IM XDMS'. The IM servers handle application-specific tasks such as distribution of MSRP messages to all chat members and distribution of immediate messages to all buddylist members. Servers also provide interfaces to the operator's provisioning and network management systems and create application-specific Charging Detail Records (CDRs). The IM server is connected to the IMS via the IMS Service Control (ISC) reference point. The IMS takes care of common functions, such as user authentication for IM, session routing and generic charging based on SIP. IM client is usually software in the User Equipment (UE) but it could be an application (also in the PC).

Usually, a presence service is associated with IM, as presence adds value to IM (e.g., users are able to learn another user's willingness and availability for IM communication). Even though the IM service works without presence, it is still shown here in the architecture (Figure 7.6).

#### 7.5.1.1 IM Server

An IM server is an application server in the IMS architecture that provides the IM service for users. It controls the IM session setup procedure, enforces policy defined for incoming/outgoing IM (e.g., who is allowed to join, who is allowed to invite more members, decides whether a session should be released when a particular user leaves, decides whether other users should be invited when a particular user joins, enforces operator's



**Figure 7.6** OMA IM architecture

policy for message size and content), provides information about group users (e.g., informs when somebody joins in or leaves a group), executes user's blocklist/grantlist, may publish IM related presence information on behalf of the user, informs and delivers stored messages when the user comes online again, stores messages in the network when requested and provides the possibility of retrieving stored conversations later on. Furthermore, the IM server takes care of MSRP traffic distribution. So, in short, an IM server handles both control-plane and MSRP user-plane traffic associated with the IM service, and for this purpose it uses IMS reference points ISC and Mb.

In OMA four different IM server roles have been defined: participating IM function, controlling IM function, Deferred Messaging function, and Conversation History Functions. To fulfil all the features of the IM Service, an IM Server must play all of these roles from time to time. IM Server roles must not be interpreted as physical entities. The assignment of the IM server role takes place during an IM session setup in such a way that there will be only one IM server performing the controlling IM function and two or more IM servers performing the participating IM function depending on the number of IM session participants. In the case of a one-to-one IM session and an ad hoc IM group session the IM server of the inviting user performs the controlling IM function.<sup>3</sup> In the case of group communication the IM server owning/hosting the group identity performs the controlling IM function. The Deferred Messaging Function and Conversation History Functions are bounded to participating IM function [OMA IM AD]. Figure 7.7 shows different roles in action taking into account roles in the originating and terminating side of requests.

### 7.5.1.2 IM Client

The IM client according to the OMA standard is a functional entity on the UE that is able to register itself to IMS using a IM feature tag and, indicating a IM release version in SIP REGISTER request, sends and receives immediate messages, initiates/modifies/releases IM sessions, performs file transfer operations, supports the capability to set IM service settings, gets participants of IM session, retrieves/deletes stored messages, activates/deactivates conversation history and uses MSRP for IM user-plane.

### 7.5.2 IM Communication

OMA IM supports three fundamental modes of IM communication: (1) Pager Mode (2) Large Message Mode and (3) IM Session Mode. The first is appropriate for brief message exchanges such as announcements with acknowledgements. The second is for brief message exchanges in which the size of the individual message is large (such as when carrying multimedia content). Large message mode is also considered as immediate messaging because the SIP session is only used to transmit exactly one large message after which the SIP session is torn down. The last is similar to a conference hosted by a network where individual users join and leave the group conversation over time [OMA IM AD].

---

<sup>3</sup> Controlling IM Function can be optional in One-to-One IM sessions since the need for a centralized control point would not be necessary.

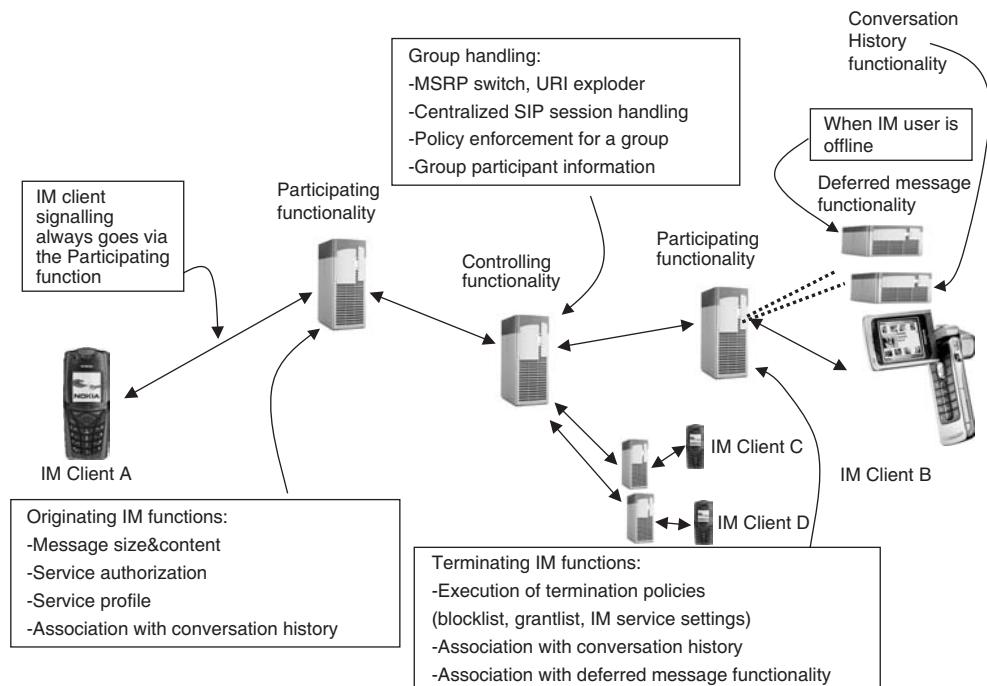
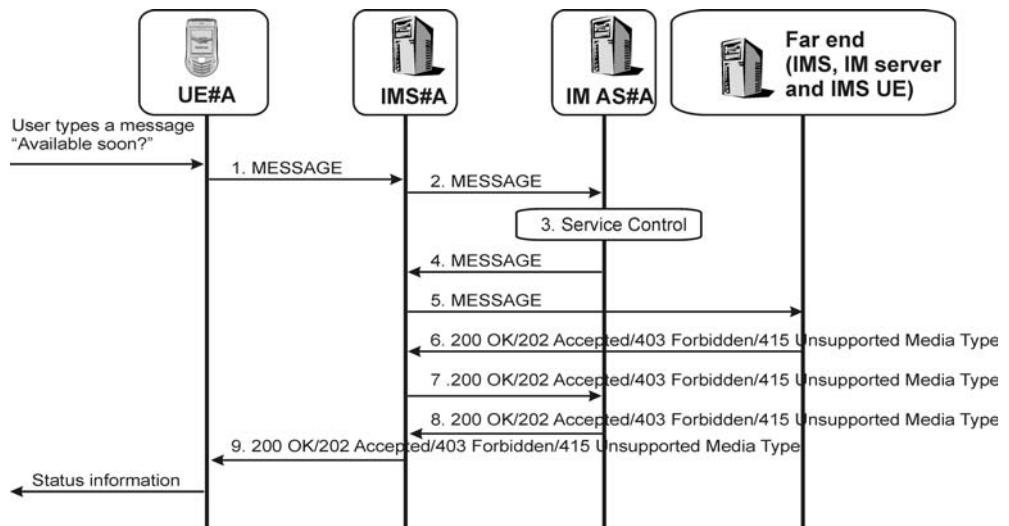


Figure 7.7 OMA IM server architecture

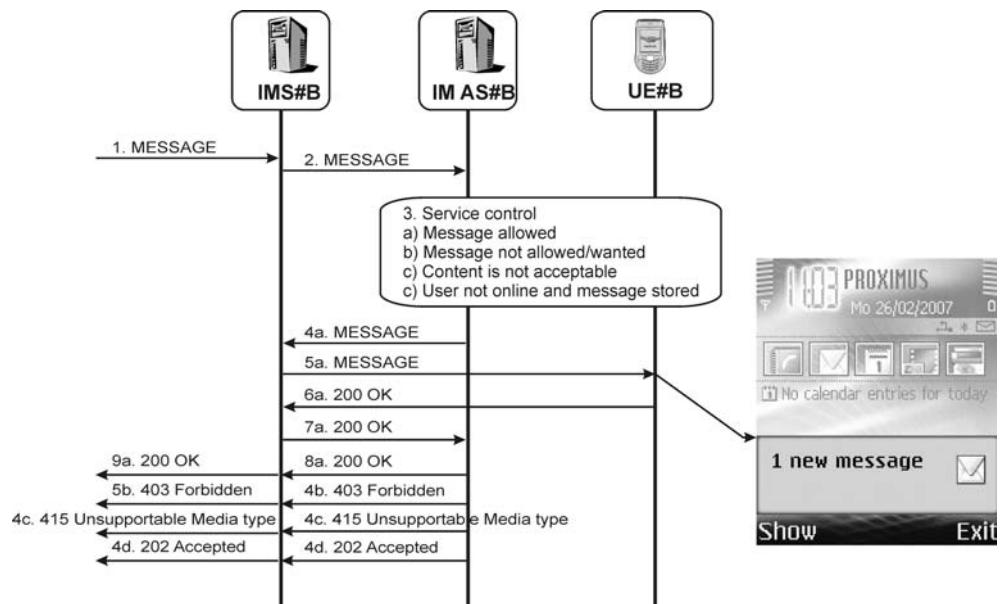
### 7.5.2.1 OMA IM Immediate Messaging

OMA IM Immediate Messaging follows principles introduced in Section 7.2 but it adds value by introducing additional features around it. Main features enabled in the originating side are: capability to use message buddylists, possibility to store sent messages in the network and enhanced operator control of messaging service. In addition to the two last mentioned features the usage of the terminating IM Server allows users to create own blocklists and grantlists based on own preferences, temporal to block all incoming immediate messages and to get messages stored when being offline.

Figures 7.8, 7.9 and 7.10 show how immediate messages get delivered according to the OMA IM standard. As described earlier the IM AS is connected via the ISC reference point to the IMS so by setting up proper IMS initial filter criteria (see Section 3.13) all originated immediate messaging requests get routed to the IM server. The originating IM server executes all necessary operator service control procedures and if the request does not conform to operator policies the IM server can reject the request with an appropriate SIP error message. Here we assume that the request meets all operator requirements and gets routed towards a recipient. Once the request arrives at the terminating network it gets again routed to the local IM Server based on IMS initial filter criteria. The terminating IM server first checks the recipient user's willingness to receive any immediate message by inspecting user's IM service setting values that the user has a published right after IMS registration for the IM service. If barring is activated the IM server immediately rejects the request with SIP 403 Forbidden response including proper warning text which



**Figure 7.8** Originating immediate message in OMA IM



**Figure 7.9** Terminating immediate message in OMA IM

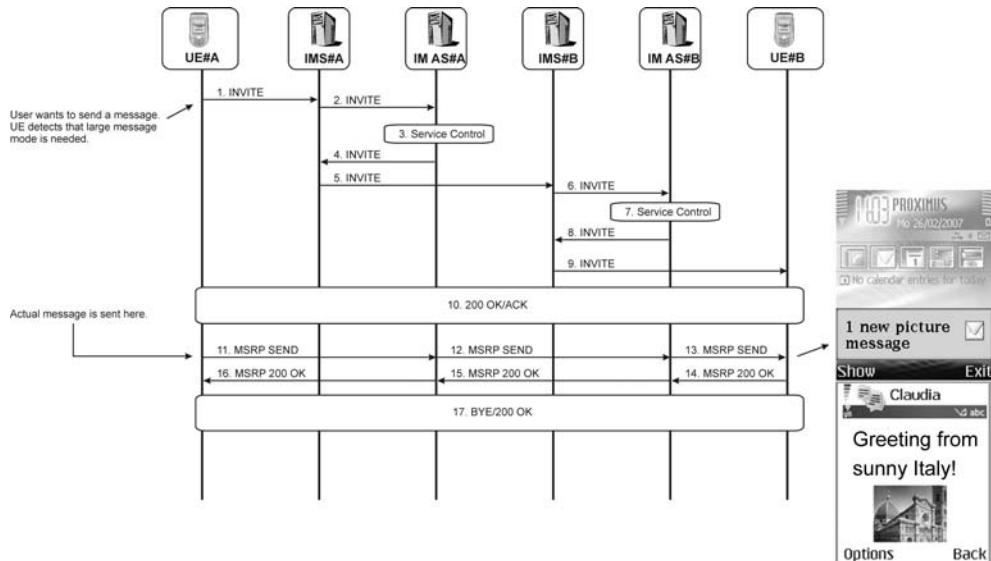


Figure 7.10 Large message mode in OMA IM

could be rendered to the originator's display. Next check is to analyze terminating user's grant-and blocklist if this check does not allow the request to bypass then it gets rejected again with SIP 403 Forbidden. If the message is still alive then the attached content, if any, (e.g. image, voice clip) is checked against operator's policy and only if it conforms the policy (if it is not 415 Unsupported Media type error response is given), final test, registration check is done. For users online the message gets delivered and for offline users the message gets stored in the IM Server for later delivery.

### 7.5.2.2 OMA IM Session Based Messaging

OMA IM session based messaging follows principles introduced in Section 7.3 but it adds value by supporting various types of communication models and flexible session policies with the help of IM servers.

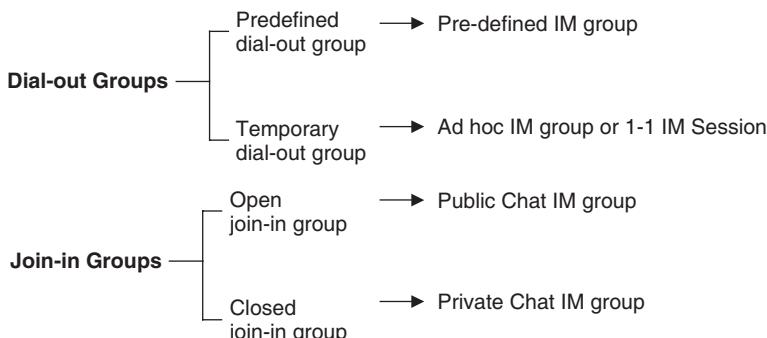
The first type of communication model is dial-out messaging session. In dial-out sessions, a user invites a person or group of users to participate in a messaging session. In the case of group communication the invited group may be a pre-defined IM group or a list of users selected from the calling user's phone book as a temporary arrangement (so-called ad hoc IM group). In the latter case, in particular, the ability to see other users' availability, or presence statuses, before making the dial-out session brings clear additional value for the user. A dial-out session suits unplanned situations or cases where the participants must be handpicked. The invited client receives an indication of the incoming session and may automatically accept messaging request or it may ask permission to accept the communication request from the end user.

The second type of communication model is join-in messaging sessions, chat group, where the participants themselves explicitly choose to join a IM messaging session. In

this way users have full control over which groups they participate in. They will never receive any traffic unless they have joined the group. Join-in operation is well suited for communication during any routine or pre-planned activities. Participation in a chat group is analogous to real-life activities such as watching TV, going to a movie or participating in a meeting. Chat IM sessions can go on for hours, with actual communication comprising only a small portion of the total session time. So, being in a chat IM session should not prevent a user receiving other sessions in the meantime. The user should also be able to participate in several chat sessions simultaneously (e.g., ‘my family’, ‘basketball friends’, ‘beer buddies’). A chat group can be a public group without any access control or a private group with a list of members. Public groups are open to anyone who knows the group identification (SIP URI of the group). The group identification can be found, for instance, on an operator portal or chat room listing. Public IM chat groups are suitable as open discussion forums on general and specific topics (e.g., fishing, cars, football). Private groups are groups where access is limited to pre-defined users. For private groups where access control methods are used, see Section 5.7.2.1. To join a private group, users need to know the group identification (SIP URI of the group) and they need to have the right to join the group session. Private IM chat groups are best suited when you don’t want to expose your messages beyond a well defined group. Figure 7.11 summarizes the different IM communication types.

Figure 7.12 shows an example of IM session initiation and Figure 7.13 shows an example of IM session termination. Once the UE gets a request from the user it forms a SIP INVITE request targeted at the recipient. The request contains some OMA specific flavours as follows: Accept-Contact and Contact headers contain OMA IM specific feature-tag, +g.oma.sip-im, and the request contains SIP User-Agent header indicating the IM release version. When the session is for 1-1 and predefined IM group then an address of target user or name of predefined IM group is included in the Request-URI header and in the case of ad-hoc IM group session the Request-URI header contains conference factory URI and intended group members are included in MIME resource-list body (see Section 8.2.1 for details).

The S-CSCF at IMS#A executes the initial filter criteria and based on presence of the OMA IM specific feature-tag the request gets routed to the originating IM server. The originating IM server performs all necessary service checks (e.g. verifies that proposed



**Figure 7.11** Different IM session types

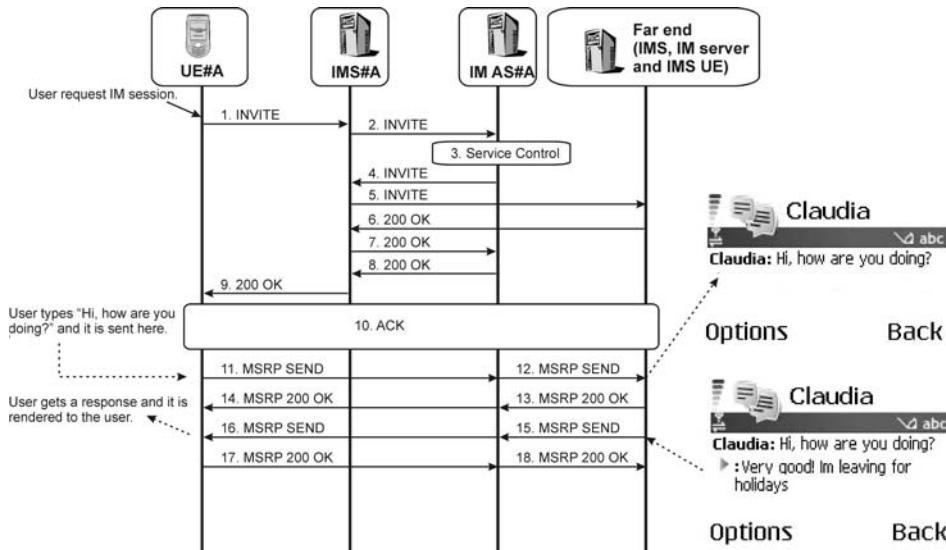
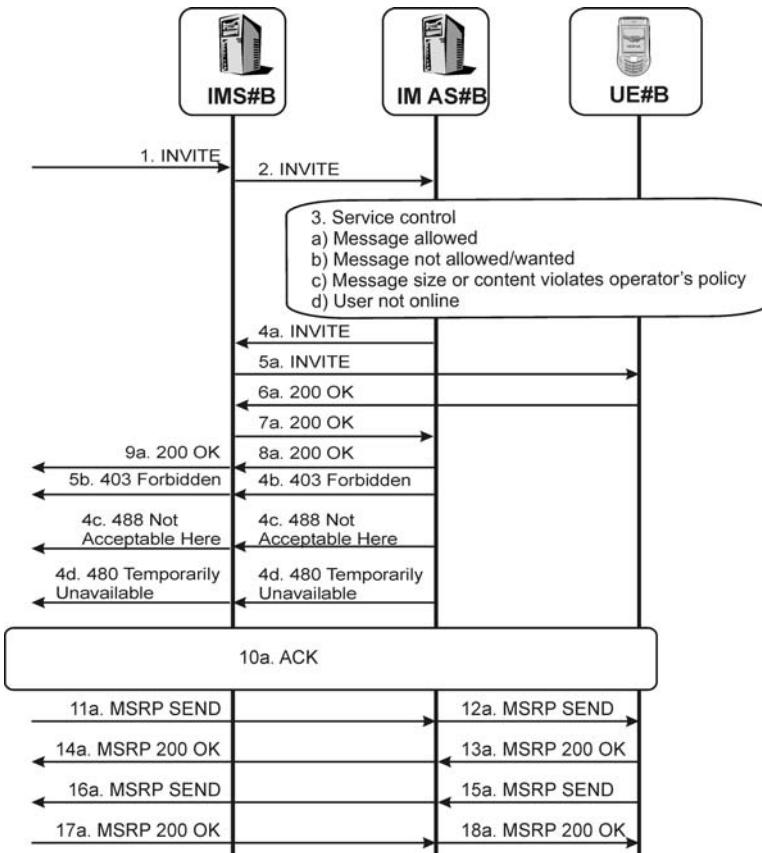


Figure 7.12 OMA IM session initiation

SDP media attributes are according to operator policy and the user has a valid messaging subscription, is the pre-defined group for the IM server and if the user is allowed to use the pre-defined IM group, or whether the number of invited members exceeds group/operator's limit). If no service control breach is found the IM Server initiates session establishment towards session participant(s) [only one recipient is shown in the example figure].

The following steps are repeated for each participant. The IM session request gets routed using normal IMS routing mechanisms to the recipients network and like in the originating network the S-CSCF detects presence of the OMA IM specific feature tag and routes the request to the terminating IM server. The terminating IM server first checks the recipient user's willingness to receive any session request by inspecting the user's IM service setting values that the user has published right after IMS registration for the IM service. Service settings contain a value to block all incoming IM session requests (incoming-session-barring). If barring is activated the IM server immediately rejects the session attempt with a SIP 403 Forbidden response including proper warning text which could be rendered to the originator's display. The next check is to analyze terminating user's grant- and blocklist. If this check does not allow the request to pass then it gets rejected again with SIP 403 Forbidden. If message is still alive then SDP media attributes are checked against operator's policy and only if attributes conforms the policy (if it is not 488 Not Acceptable Here error response is given), final test, registration check is done. For online users the session request gets delivered and for offline users the request is rejected with SIP 480 Temporarily Unavailable.

When session setup is completed all session members can start to send actual data using the MSRP protocol. Usually MSRP traffic itself goes via MSRP switch at the IM Servers (see Section 7.5.6).



**Figure 7.13** OMA IM session termination

### 7.5.2.3 Private Message During IM Session

Default behaviour in IM session is that all sent messages are delivered to the all session participants by the IM Server. Sometimes the user may want to deliver a specific message to one or more participants of the IM session. Sending a single message to a limited set of session participants is called private messaging. This capability is not included in basic IETF MSRP protocol and IETF has not showed great interest in standardizing this feature. Therefore, OMA defined its own application logic to support private messaging. An example of private message is given in Section 7.5.6.

### 7.5.2.4 Chat Alias

In public chat rooms and discussion forums users usually tends to prefer some other identity instead of their real identity. These identities can be called chat aliases or nicknames which could take, for example, the following forms: Gamemaster, Miss Joy,

Mister President, John-X. Moreover, user may want to use different aliases in different forums and they may even want to change their identity on the fly. Unfortunately, the IETF SIP and MSRP specifications do not currently support the capability to negotiate chat alias. OMA IM has defined its own mechanism for chat alias negotiation using the SIP privacy capability [RFC3323] and conference state event package [RFC4575] as follows.

For using chat alias in the IM session the IM client needs to request anonymity as part of the initial INVITE for IM session. This request may contain the preferred dynamic chat alias. The wanted chat alias is given as display-name field of From header for example as follows:

```
From: "Mister President" <sip:anonymous@anonymous.invalid>;
tag=9802748
Privacy:id
```

This request is processed by the controlling IM function. It verifies the requested chat alias and ensures that it is unique (checks that the proposed chat alias is not already in use in this IM session) and stores the acceptable chat alias for later use. The IM Server communicates the assigned chat alias to the UE when the UE subscribes to IM session participant information (conference state event package see Section 8.2.3). The chat alias is delivered as OMA specific XML attribute, ‘yourown’ to [RFC4575]. The example below does not include all elements of the XML document in [RFC 4575] for the sake of simplicity.

```
<conference-state>
<user-count>33</user-count>
</conference-state>
<users>
<user entity="sip:bob@example.com" state="full">
<display-text>Bob Hoskins</display-text>
</user>
...
<user entity="sip:anonymous@anonymous.invalid" state="full"
      yourown="true">
<display-text>Mister President </display-text>
</user>
</users>
```

After learning the assigned chat alias it will be used to send messages in the IM session (MSRP SEND).

### 7.5.2.5 Barring of IM Communication

An IM user is able to instruct an IM server to reject all new incoming IM communication attempts. An IM user is able to bar separately immediate messaging and session mode messaging. When barring is activated, the participating IM server will reject all incoming requests with a SIP 480 Temporarily Unavailable response and, at the same time, the

participating IM server will keep any ongoing IM session(s) intact. Section 7.5.5 shows how barring can be activated and de-activated.

#### 7.5.2.6 Participant Information

An IM user can request information about IM session participants and their status in the IM session. The IM client uses the conference-state event package to learn about changes in IM participants: in other words, a user can learn, through notifications, who has joined or left a conference. Users can subscribe to a conference state by sending a SIP SUBSCRIBE request to the SIP address that identifies the controlling IM function. The controlling IM function acts as a notifier for this event package. This generic capability is described in more detail as part of the conferencing service in Chapter 8. Here limitations and extension against the generic functionality are provided.

The OMA IM Release 1.0 specification uses only a limited subset of offered functions as follows:

- identifier of IM group format depends on IM session type;
- IM user identities are now part of the IM session;
- IM user status (connected, disconnected).

Before the IM client displays this information to the user it is expected to compare the received list against user's IM block list and if a match is found then the IM client is expected to highlight this additional information to the user.

In OMA, an extra attribute has been defined on top of basic conference-state event package to convey user's chat alias. See Section 7.5.2.4 for more information.

#### 7.5.3 Conversation History

Conversation history function provides the capability to store a copy of all or selected messages at network storage based on user's desire. The owner of stored messages can search and browse summary of stored messages, and they can retrieve one or more messages, and they can delete one or more messages from conversation storage.

##### 7.5.3.1 Conversation History Activation and Deactivation

When a user wants to store a copy of immediate messages or the IM sessions she can request the conversation history function of the IM server to take required action. There are two different ways to activate conversation history function. The IM client could activate storage of messaging by publishing IM service settings with a proper value for *hist-settings* (see Section 7.5.5). This model is well-suited when the user wants to get all messages stored at the network. Another model is applicable for IM sessions. The user could start storing active conversation to the network by requesting to add its network storage as an additional member to the ongoing session. When the user wants to stop the history function then the IM client needs to publish IM service settings again with the proper value or remove the conversation history function from the ongoing IM session.

### 7.5.3.2 Creating Messaging History

Once the conversation history function is activated then the participating IM function stores all incoming (for terminating request) and outgoing (for originating request) messages (see Figure 7.14). For pager mode messages (SIP MESSAGE) complete SIP MESSAGE gets stored. For large message mode and session mode messaging all relevant SIP headers of INVITE, 200 OK and BYE requests (e.g. From, To, P-Asserted-Identity, Subject, Date) are stored as well as all relevant content of actual MSRP messages. The IM server assigns a history reference for each stored message. This unique reference will be used to retrieve/delete particular stored message(s) at a later stage. In addition to storing the actual message, the participating IM server stores metadata of the message in IM-XDMS. This summary contains, for example, date, time, history reference, size of the saved message, type of message and identity of sender.

### 7.5.3.3 Retrieving Stored Messages from Conversation History Function

Retrieving stored messages contains three steps: learning history reference of the stored message, establishing SIP session between IM client and conversation history function and receiving an actual stored message using MSRP. For completing the first step the IM client needs either to connect to IM XDMS and retrieve the XML document summarizing all stored messages or to connect to Search Proxy and search individual stored messages using keywords (see Section 5.7.1.3 for details). After getting the XML document describing wanted message(s) including history references the IM client sends

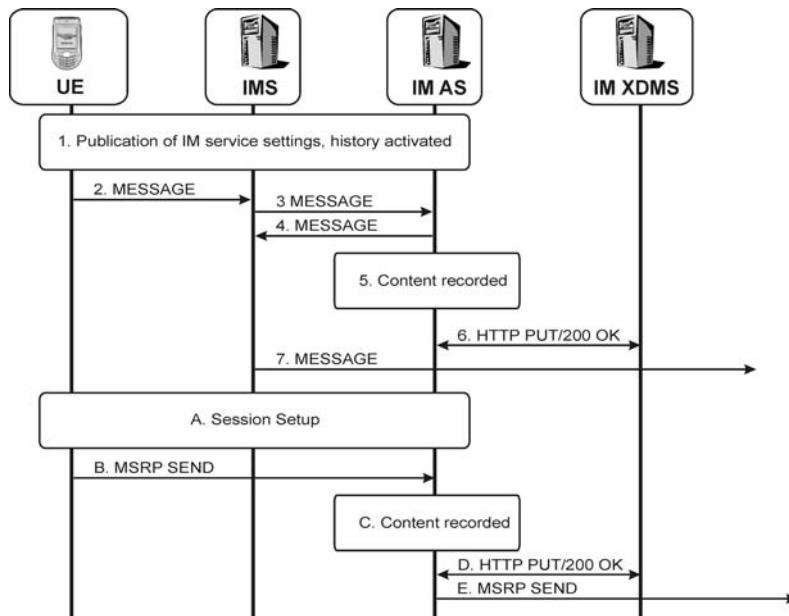


Figure 7.14 Conversation history function

SIP INVITE to the following provisioned SIP address *History@hostname* (e.g. *History@imserver1.example.com*). This request also contains URI-list body including history references of messages to be retrieved.

```
INVITE sip:History@imserver1.example.com SIP/2.0
Content-Type: multiparty/mixed; boundary='Boundary'
--Boundary
Content-Type: application/SDP
a=reconly
--Boundary
Content-Type: application/resource-list+xml
Content-Disposition: recipient-list

<?xml version="1.0" encoding="UTF-8"?>
<resource-lists xmlns="urn:ietf:params:xml:ns:resource-lists">
    <list>
        <entry uri="pagermessage1@imstorage.example.com" />
        <entry uri="session1@imstorage.example.com" />
    </list>
</resource-lists>
Boundary--
```

The IM server verifies and authorizes the incoming request and if the requesting user is allowed to retrieve the messages it accepts the session requests and downloads stored messages from storage and initiates MSRP transfer for each individual saved message. Once the IM client receives MSRP traffic it is able to render the stored messages to the end user like any other messages.

#### 7.5.3.4 Deleting Stored Messages from Conversation History Function

Similarly to retrieval operation the IM client needs to know history reference(s) of the message to be deleted before the user can delete one or more stored messages. For obtaining needed history reference the IM client needs to use XDM architecture (see Section 5.7.1.3). Once the IM client knows the reference(s) it uses SIP REFER request to remove message(s). This request is sent to provisioned SIP address *Delete@hostname* (e.g. *Delete@imserver1.example.com*). Refer-To field is used to indicate whether user wants to delete one or several or all stored messages.

- Single message to be deleted: Refer-To contains single history reference.
- Multiple messages to be deleted: Refer-To contains reference to XML body containing list of messages to be deleted.
- All messages to be deleted: Refer-To contains *sip:History@hostname*.

#### 7.5.4 Deferred Messaging

Like in SMS, MMSC and email services OMA IM provides the capability of storing immediate messages for later delivery when user is offline. Deferred messaging function of IM server is used for this purpose. When user is offline messages get stored at the

terminating network and metadata of stored message is pushed to IM XDMS. The metadata contains, for example, date, time, message reference, size of the saved message and identity of sender. For SIP MESSAGES whole message is stored as received and for large message mode all relevant SIP headers of INVITE, 200 OK and BYE requests (e.g. From, To, P-Asserted-Identity, Subject, Date) are stored as well as all relevant content of actual MSRP message. The IM server assigns a message reference for each stored message. This unique reference will be used to retrieve/delete particular stored message(s) at a later stage.

For obtaining deferred messages push and pull models exist. In the push model the IM client indicates in IM service settings that it wants to retrieve all deferred messages. Once the IM server gets this request it initiates SIP session toward the client and pushes the deferred messages with MSRP. In the pull model the IM client contacts IM XDMS and retrieves message references of deferred messages. After selecting message(s) to be retrieved the IM client sends SIP INVITE to `sip:Deferred@hostname` (e.g. `sip:Deferred@imserver1.example.com`). If only some messages are to be retrieved then those are identified with message references at URI-list body of the INVITE request. The actual messages are delivered by MSRP. This is shown in Figure 7.15.

For deleting deferred messages the IM client needs to obtain message references from IM XDMS (see Section 5.7.1.3) before the user can initiate the removal of deferred

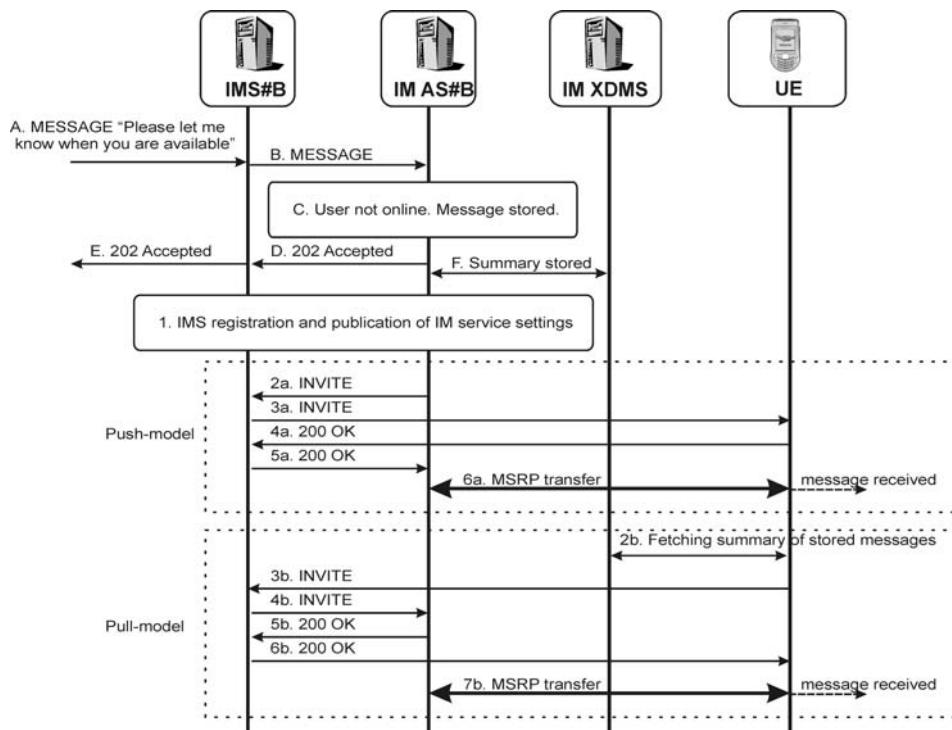


Figure 7.15 Store and forward functionality for IM users

message. Once the IM client knows the reference(s) it uses SIP REFER request to remove message(s). This request is sent to provisioned SIP address *Delete@hostname* (e.g. *Delete@imserver1.example.com*). Refer-To field is used to indicate whether the user wants to delete one or several or all deferred messages.

- Single message to be deleted: Refer-To contains single message reference.
- Multiple messages to be deleted: Refer-To contains reference to XML body containing list of message references to be deleted.
- All messages to be deleted: Refer-To contains *sip:History@hostname*.

### 7.5.5 IM Service Settings

With IM service settings an IM user can turn on/off barring of IM communication, activate/de-activate IM conversation history, request immediate delivery of stored immediate messages and indicate whether she/he want to be visible to other IM users (presence status shown as ‘available/not available for IM communication’) as described in previous sections.

IM reuses the service setting delivery mechanism from OMA PoC service settings (see Section 6.5). That is, the IM client sends service settings to the IM server with SIP PUBLISH that contains OMA specific XML payload. More precisely IM uses two already defined PoC service setting XML elements for pager mode barring (*ipab-settings*) and IM session barring (*isb-settings*) and it has defined three new XML elements on top of XML schema defined for PoC. Table 7.1 shows different XML elements, use case and possible values. To prevent problems in the future when a single IMS application server may implement both IM and PoC service additional XML attribute, ‘service-id’ was introduced to clearly distinguish between barring values for IM and PoC service.<sup>4</sup>

The example below describes a case where the IM Client does not activate barrings, sets his/her visibility to other users as invisible, does not want to store pager mode messages, but wants to store session mode conversations and wants to get stored messages pushed when gets online.

```
PUBLISH sip:tobias@home1.fr SIP/2.0
From: <sip: tobias@home1.fr>;tag=31415
To: <sip: tobias@home1.fr>
Accept-Contact: *;+g.oma.sip-im;
User-Agent: IM-client/OMA1.0
Event: poc-settings
Expires: 7200
Content-Type: application/poc-settings+xml
Content-Length: (...)

<?xml version="1.0" encoding="UTF-8"?>
<poc-settings xmlns="urn:oma:params:xml:ns:poc:poc-settings"
xmlns:ss=" urn:oma:xml:im:service-settings">
<entity id="do39s8zksn2d98x">
```

---

<sup>4</sup> Service-id XML attribute may be omitted when service setting publication is issued only to the IM server and it contains only IM service settings.

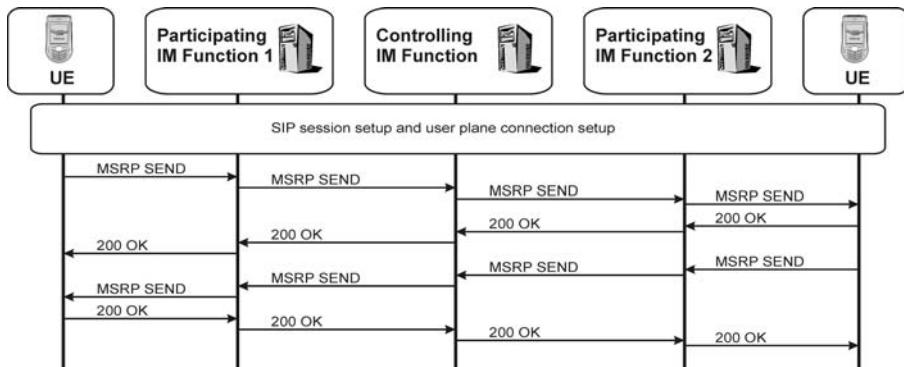
```

<isb-settings service-id="IM">
    <incoming-session-barring active="false"/>
</isb-settings>
<ipab-settings service-id="IM">
    <incoming-personal-alert-barring active="false"/>
</ipab-settings>
<ss:deferred-settings service-id="IM">
    <ss:offline-delivery active="true"/>
</ss:deferred-settings>
<ss:vis-settings service-id="IM">
    <ss:vis-status active="false"/>
</ss:vis-settings>
<ss:hist-settings service-id="IM">
    <ss:hist-activation active="false">
        <ss:pager-large-mode/></ss:hist-activation>
    <ss:hist-activation active="true"><ss:session-mode/>
        </ss:hist-activation>
    </ss:hist-settings>
</entity>
</poc-settings>

```

**Table 7.1** OMA IM service settings and possible values

Used XML items	Use case	Value
Isb-settings	Setting incoming IM Session barring active	condition true/false true = barring off false = barring on default = true
ipab-settings	Setting incoming pager mode and large mode IM barring active	condition true/false true = barring off false = barring on default = true
vis-settings	IM invisibility setting	condition true/false true = visible false = invisible default = true
deferred-settings	Pager mode and large IM offline delivery activation	condition true/false true = push-model false = pull-model default = false
hist-settings -> hist-activation -> session-mode	IM conversation storing activation for session mode communication	condition true/false true = history on false = history off default = false
hist-settings -> hist-activation -> pager-large-mode	IM conversation storing activation for pager mode and large mode communication	condition true/false true = history on false = history off default = false



**Figure 7.16** OMA IM user plane

### 7.5.6 IM User-Plane

In OMA user-plane protocol, MSRP, is used for large mode messaging, IM sessions, retrieving deferred and stored messages. Typically the IM client sends its MSRP traffic to its participating IM function that forwards the traffic to the controlling IM function. The controlling IM function takes care of relaying the received MSRP packet to all session members usually via terminating participating IM function. Figure 7.16 illustrates typical media stream path of the MSRP.<sup>5</sup> Example content of MSRP SEND packet as sent by IM client #1

```

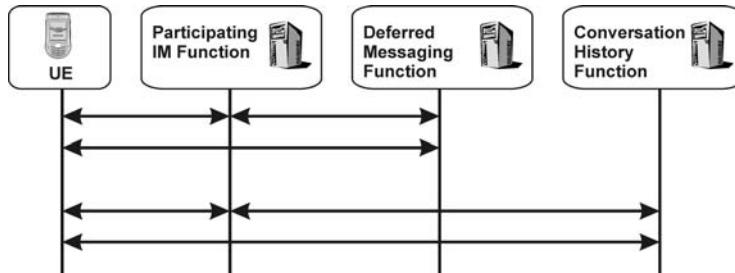
MSRP 6aef SEND
To-Path: msrps://participatingim1.example.com:9000/kjfjan;tcp \
          msrps://controllingim.example.com:9000/aeiug;tcp \
          msrps://participatingim2.terminating.com:8145/foo;tcp \
          msrps://imclient2.terminating.com:8145/xxx;tcp
From-Path: msrps://imclient1.example.com:7965/bar;tcp
Success-Report: no
Byte-Range: 1-*/*
Message-ID: 87652
Content-Type: text/plain

Hi Bob, I'm about to send you file.mpeg
-----6aef$

```

When a user wants to store content of IM session the participating IM function adds a conversation history function as an additional MSRP end point and it relays all traffic to it as well. For retrieving the stored message the IM client setups a session to the conversation history function and it receives stored messages with MSRP possible via the participating IM function (see Figure 7.17). When a user is offline and she receives a large mode message the terminating IM function inserts the deferred messaging function

<sup>5</sup> It should be noted that direct MSRP connection between IM clients is possible in one-to-one communication subject for operator's policy. Participating IM functions may not be involved on user-plane subject for operator's policy.



**Figure 7.17** OMA IM user plane for deferred messaging and conversation history

as the MSRP end point instead of terminating IM client and so the incoming message gets stored. At some later point when the message is finally to be delivered a MSRP session is created between IM client and deferred IM function, possible via the participating IM function (see Figure 7.17).

OMA IM user-plane mainly follows base MSRP RFCs RFC[4975] and RFC[4976] but it has defined additional logic in order to support chat alias and private message features.

For chat alias feature the controlling IM function needs to inspect that ‘From’ header field of the message/CPIM of the MSRP request. If the ‘From’ header field does not contain a valid value based on expected information the MSRP request will be rejected (i.e. sending user attempts to use invalid user identity).

For sending a private message inside the ongoing IM session the IM client places target user(s) URI(s) ‘To’ header of the message/CPIM payload instead of IM Session Identity or IM Group Identity. When the controlling IM function gets MSRP SEND request it needs to inspect content of ‘To’ header field of the Message/CPIM of a MSRP request and if it contains IM address of the recipient IM user(s), the request as a message is to be distributed only to those members given in ‘To’ header field. Otherwise the request should be delivered to all session participants. Similarly the received IM client needs to inspect incoming ‘To’ header field. When it contains its own address instead of IM Session Identity or IM Group Identity then it should treat the request as a private message and render it differently.

### 7.5.7 Delivery Reports

In many message exchange systems, message senders often wish to know if the recipient actually received a message. This capability is called a delivery report mechanism. There are separate solutions for pager mode messages (SIP MESSAGE) and MSRP based messages (large mode messaging and session mode messaging). Pager mode message delivery reports are fully based on IETF solution as defined in [draft-ietf-simple-imdn-06] while for MSRP based message communication OMA has defined additional functionality on top solution provided by the base MSRP protocol.

#### 7.5.7.1 Pager Mode Messaging

When a user wishes to receive delivery reports it adds appropriate message/CPIM payload to the outgoing SIP MESSAGE request based on [draft-ietf-simple-imdn-06]. The

requesting user gets delivery reports back from the controlling IM function with standalone SIP MESSAGE(s) that contains a message/CPIM payload carrying the status of sent messages. An example of a message requesting delivery report looks like this:

```
MESSAGE sip:fcunited@imservice.example.com SIP/2.0
From: <sip:tobias@home1.fr>;tag=31415
To: <sip:fcunited@imservice.example.com>
Accept-Contact: *;+g.oma.sip-im
User-Agent: IM-client/OMA1.0
Content-type: Message/CPIM
From: <sip:tobias@home1.fr>
To: <sip:fcunited@imservice.example.com>
Subject: Updated information about the game!
NS: imdn <urn:ietf:params:imdn>
imdn.Message-ID: 34jk324j
imdnDisposition-Notification: positive-delivery, negative-delivery
Content-type: text/plain;
```

Hi all, kick-off time has been changed. New kick-off time is  
45 minutes later. See you all soon!

In the above example Tobias earlier on created a pre-defined IM group, fcunited@imservice.example.com, that contains leadership team members of their football team and now he wishes to inform them that the game starts bit later than expected. As he considers this information important he wishes to see if the message was successfully delivered in order to call those members who did not get the message due to whatever reason. For fulfilling Tobias' wish the IM client sends the actual message encapsulated inside a message/CPIM wrapper and the message/CPIM contains a specific header 'imdnDisposition-Notification' to express the user's desire. In this example it indicates that the user wishes to receive positive and negative delivery notifications. In addition, the 'imdn.Message-ID' header is included with a unique value enabling the IM Sender to match any delivery reports with their corresponding IMs sent earlier.

### 7.5.7.2 Session Mode Messaging and Large Message Mode Messaging

When a user wishes to receive delivery reports for MSRP SEND request it adds message/CPIM payload containing additional OMA IM specific content. This OMA IM specific content consists of a new namespace, <urn:oma:xml:poc:final-report>, and a new header field in message/cpim body. The name of this header is 'Final-Report' and belongs to the above mentioned namespace.

```
NS: FR <urn:oma: xml:poc:final-report>
FR.Final-Report=yes
```

For delivering the actual delivery notifications a OMA specific MIME body, 'application/vnd.oma.poc.final-report', is attached to message/CPIM payload of MSRP REPORT or MSRP SEND request. Delivery reports are sent by controlling IM function which may aggregate a multiple response to single delivery report to reduce signalling overhead. An

example of final delivery report MIME body for the requesting IM client could look like this.

```
<?xml version="1.0" encoding="UTF-8"?>
<final xmlns="urn:oma:xml:poc:final-report" Message-ID="r2d2"
      last="true">
<leg uri="sip:alice@example.com" status="200"/>
<leg uri="sip:bob@example.com" status="200"/>
<leg uri="sip:cecile@example.com" status="9999" max-size="50000"/>
</final>
```

The Final Delivery Report document begins with the root `<final>` element. The `<final>` element consists of number of `<leg>` elements and two attributes, ‘last’ and ‘Message-ID’. The ‘last’ attribute is used only in the last Final Report and it indicates that this is the last Final Delivery Report document. ‘Message-ID’ contains the Message-ID of the SEND message for which the Final Delivery Report is generated. Each `<leg>` element contains a mandatory ‘uri’ attribute containing the URI of the recipient (e.g. Alice, Bob, Cecile) and a ‘status’ attribute (e.g. 200, 9999) with or without a ‘max-size’ attribute.

If the MSRP message was completely delivered to the participant or if the MSRP message delivery to the participant failed, the `<leg>` element contains the ‘status’ attribute containing the MSRP status code of the respective recipient. If the MSRP message cannot be sent to the participant because the participant has negotiated the SDP attribute ‘`a = max-size`’ lower than the MSRP message size, the `<leg>` element contains the ‘status’ attribute with value ‘9999’ and the ‘max-size’ attribute contains the corresponding previously negotiated value.

# 8

# Conferencing

A conference is a conversation between multiple participants. There are many different types of conferences, including loosely coupled conferences, fully distributed multiparty conferences and tightly coupled conferences. In this chapter only the latter is described since it is the only one that is of interest to the Internet Protocol Multimedia Subsystem (IMS).

Conferencing is not just limited to audio; the popularity of video and text conferencing, better known as chatting, has been growing rapidly over the past few years. This popularity is due to conferencing's ability to transport files, enabling whiteboard sharing and of course provide simulating face-to-face meetings by exchanging video, all in real time.

## 8.1 IMS Conferencing Architecture and Principles

### 8.1.1 SIP Focus/Conferencing AS/MRFC

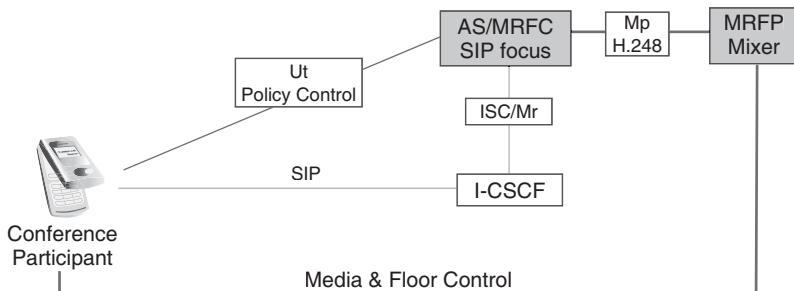
Figure 8.1 gives an overview of IMS conferencing architecture.

Tightly coupled conferences are hosted at a central point of control to which each conference participant has a connection.

The central point is referred to as a 'SIP focus' that is addressed by a public service identity (PSI) that relates to a conference. The focus is the endpoint for all SIP signalling dialogs towards all conference participants. Within IMS the SIP focus is an Application Server (AS) which is co-located with a Multimedia Resource Function Controller (MRFC), which is referred to as 'Conferencing AS/MRFC'.

### 8.1.2 Conference Mixer – MRFP

Whilst all the SIP signalling and also the basic conference control is performed by the Conferencing AS/MRFC, all the media streams that are related to the conference are terminated within the network by a so-called 'mixer'. Within IMS the mixer is a Multimedia Resource Function Processor (MRFP), which is controlled by the MRFC (which is part of the Conferencing AS/MRFC) by means of H.248/MEGACO protocol.



**Figure 8.1** IMS conferencing architecture

As a conference usually consists of more than two users, the MRFP needs at any time of the conference combine (mix) all incoming media streams from all participants and send the combined media stream back to the participants, so that every participant gets aware of the media sent by all other participants.

The MRFP can also provide transcoding. For example, if the conference participants use different audio or video codecs, the MRFC will ‘translate’ the media streams (this is called transcoding) into the codecs understood by the individual conference participants.

#### 8.1.3 Conference Participant

Users participating in a conference are called conference participants or just participant. Within a basic scenario, a participant will have at least a SIP dialog established with the conference AS/MRFC and a media connection (e.g. an audio stream) established with the MRFP.

Users can also gain further information about the ongoing conference by subscribing to the conference state event package that is defined in [RFC 4575]. The conference state event package informs the participant about the users currently online in a conference, which media these participants are using and other related information. A user subscribed to the conference state event package will also be notified whenever a participant joins or leaves the conference.

#### 8.1.4 Conference Moderator, Floor Control and Conference Policy Control

In some conferences the concept of a moderator is used. The moderator is a special participant who is allowed to control certain aspects of the conference.

A conference moderator can give the right to speak to a participant or withdraw it temporarily from a participant – this handling of the right to speak by a moderator is called floor control. In order to allow floor control, the conference moderator makes use of the Binary Floor Control Protocol (BFCP) as specified in [RFC 4582]. BFCP is used directly over the media connection, i.e. between the UE of the conference moderator and the MRFP.

A conference policy is a set of rules associated with a certain conference. These rules include directives on the lifespan of the conference, who can and who cannot join the conference (membership policy), definitions of roles available in the conference and the

responsibilities associated with those roles, and policies on who is allowed to request which roles. The conference policy also includes the media policy: the mixing characteristics of a conference.

### 8.1.5 Sidebars

A sidebar is a sub-conference to the existing main conference. Within a sidebar some of the participants can communicate, without the other participants being aware of this. Sidebars may occur in larger conferences, when some participants want to discuss a specific issue outside the main conference.

## 8.2 IMS Conferencing Procedures

### 8.2.1 Conference Creation

#### 8.2.1.1 Creation of a Hosted Conference/Conference Policy

Before participants can join a conference, the conference needs to be created at the conference AS/MRFC, which means, the parameters for the conference need to be configured. Such a configuration is called the conference policy and it can include many different elements, for example:

- the name of the conference, e.g. the subject that the participants will discuss;
- the address of the conference, i.e. the conference URI which is used to dial into the conference;
- the time when the conference starts;
- the maximum duration of the conference;
- the maximum number of users allowed for the conference;
- a list of users who are allowed to join the conference (white list);
- a list of users who are not allowed to join the conference (black list);
- the name/identity of the conference moderator;
- a list of users who should be called by the conference AS/MRFC in order to make them join the conference;
- media restrictions for the conference (e.g. a conference policy could forbid that instant messaging is used or at least forbid it for certain users).

There are different ways in which the conference policy is provided to the conference AS/MRFC. The network operator might, for example, offer users a web page, on which the conference policy can be set.

Within IETF a conference policy control protocol (CPCP) is currently under discussion, which will allow the conference moderator and specially privileged users to set up the conference policy automatically from the IMS terminal. CPCP will be used over the Ut interface and will be transported via XCAP protocol. It will also allow that the conference policy gets changed whilst the conference is ongoing, for example the list of users who should be called into the conference by the conference AS/MRFC could be extended during the ongoing conference by the moderator – the conference AS/MRFC would then set up a call to the indicated user, so that the user can join the conference.

### 8.2.1.2 Creation of an Ad-hoc Conference

A network operator can allow a user to create an ad-hoc conference, which has no specified start time and is automatically established as soon as the first user joins the conference. In order to do this, the user creating a conference must call a so-called conference-factory URI, which is a public service identity (PSI) that is exclusively used to create an ad-hoc conference.

Figure 8.2 shows how an ad-hoc conference gets created. The user can set up an ad-hoc conference by sending a SIP INVITE request to the conference-factory URI:

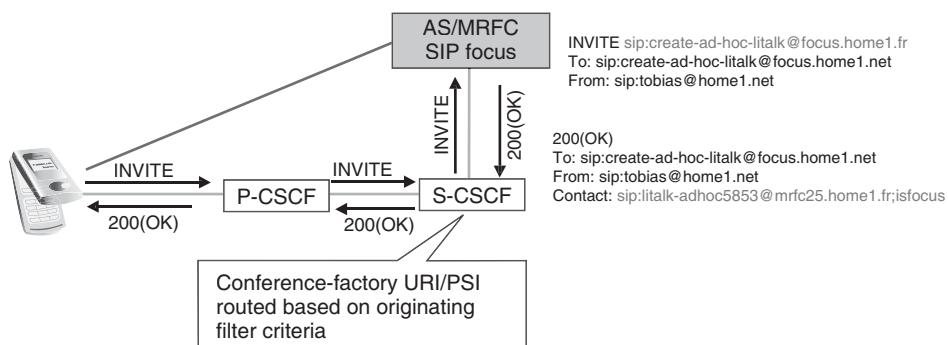
```
INVITE sip:create-adhoc-litalk@focus.home1.fr SIP/2.0
```

This SIP INVITE request will include in the body all the media streams that the user wants to establish for this conference.

The SIP INVITE request is forwarded to the calling users S-CSCF, where a filter criteria triggers the routing of calls destined for the indicated conference-factory PSI to the conference AS/MRFC of the served user. The S-CSCF of the calling user therefore will directly send the SIP INVITE request to the conference AS/MRF. A more detailed example for PSI routing can be found in Section 13.3.6.3.

When the conference AS/MRFC receives the SIP INVITE request, it immediately responds to the UE (in this example we assume that this is done by a 200 (OK) response), indicating the conference URI (note: not the conference-factory URI) in the Contact header of the SIP response. It also indicates in this address, that the conference AS/MRFC will act as a SIP focus for the conference by adding the ‘;focus’ parameter.

```
SIP/2.0 200 (OK)
Contact: sip:litalk-adhoc5853@mrfc25.home1.fr;focus
```



**Figure 8.2** Ad-hoc conference creation

The calling user does not need to do anything more, by receiving the 200 (OK) response, the user is the first active participant in the newly established ad hoc conference. Other participants can now join the conference as described in the next sections.

### 8.2.1.3 Creation of an Ad-hoc Conference with URI list

In the example above, the user created an ad-hoc conference and then waited for the other participants to join. Most likely the user wants the other participants to join immediately, once the conference has been created and therefore wants to instruct the conference AS/MRFC to set up calls to the other participants.

The conference creator can achieve this, by adding a list of the users, who should immediately be invited to the conference, to the SIP INVITE request. Such a so-called ‘URI-list’ (as described in [draft-ietf-sip-uri-list-conferencing]) can be attached within the body of the INVITE request.

Unfortunately, the body of the INVITE request already includes the SDP indication for the media that the calling user wants to use for the conference. In order to transport both bodies – the SDP and the URI list – the SIP INVITE request must indicate, that multiple bodies are included:

```
INVITE sip:create-adhoc-litalk@focus.home1.fr SIP/2.0
Content-Type: multipart/mixed;boundary="boundary1"

--boundary1
Content-Type: application/sdp

//SDP Information not shown here

--boundary1
Content-Type: application/resource-lists+xml
Content-Disposition: recipient-list

<?xml version="1.0" encoding="UTF-8"?>
<resource-lists xmlns="urn:ietf:params:xml:ns:resource-lists"
    xmlns:cp="urn:ietf:params:xml:ns:capacity">
    <list>
        <entry uri="sip:peter@home8.de" cp:capacity="to" />
        <entry uri="sip:kevin@home5.co.uk" cp:capacity="to" />
        <entry uri="sip:cathy@foreign.com" cp:capacity="to" />
    </list>
</resource-lists>
--boundary1-
```

From this we see, that the user wants to create the ad-hoc conference (based on the conference-factory URI in the request line) and that three additional users should be called directly into the conference, as described in Sections 8.2.2.2 and 8.2.2.3.

### 8.2.2 Joining a Conference

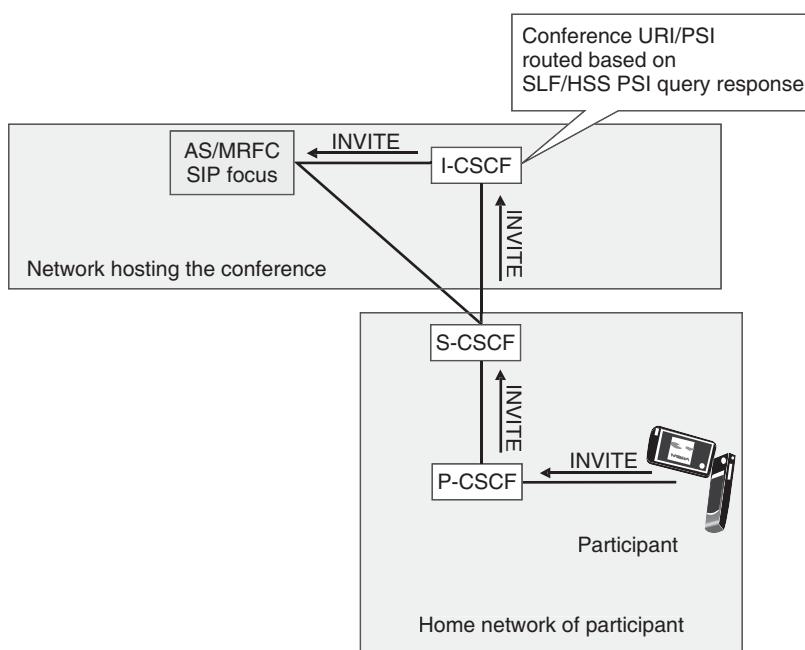
#### 8.2.2.1 User Calling into the Conference

Figure 8.3 shows how a user can dial into a conference.

Users can join a conference by calling the conference URI, i.e. by sending a SIP INVITE request towards it:

```
INVITE sip:litalk@focus.home1.fr SIP/2.0
From: "Theresa" <sip:theresa@home2.hu>;tag=xyz
To: "Literature Talk" <sip:litalk@focus.home1.fr>
```

As the conference URI is a PSI, the SIP INVITE request will be routed through the calling users home network, where all originating services for the calling user will be performed and then further on to the I-CSCF of the network in which the conference AS/MRFC is located. The I-CSCF will query the SLF/HSS in order to find out how to route towards the destination address, i.e. the conference URI. As the conference URI is a PSI, the HSS can respond directly with the address of the conference MRFC/AS and the I-CSCF can directly forward the request there. Once the dialog is established, the I-CSCF will drop out of the connection between the S-CSCF of the calling user and the conference AS/MRFC. A more detailed example of terminating PSI routing can be found in Section 13.3.4.



**Figure 8.3** User calling into a conference

The conference AS/MRFC will check the conference policy, whether the calling user is allowed to participate in the conference and also whether the conference is currently active. If the checks are successful, the conference AS/MRFC will accept the call by sending out e.g. a SIP 200 (OK) response to the new participant. The conference AS/MRFC will also instruct the MRFP via H.248/MEGACO procedures to set up the required media resources for the incoming call to the conference.

### 8.2.2.2 Conferencing AS/MRFC Calling a User into the Conference

Based on, for example, the conference policy it can also be that a conference AS/MRFC will actively call to specific users, in order to join them into the conference. In order to do so, the AS/MRFC will send out a SIP INVITE request towards the called user:

```
INVITE sip:theresa@home2.hu SIP/2.0
From: "Literature Talk" <sip:litalk@focus.home1.fr>;tag=abc
To: "Theresa" <sip:theresa@home2.hu>
```

If the called user accepts the call, the user will be joined into the conference as a participant.

### 8.2.2.3 Referring a User into a Conference

As an alternative, a user can also be transferred (which is also called referred) into a conference.

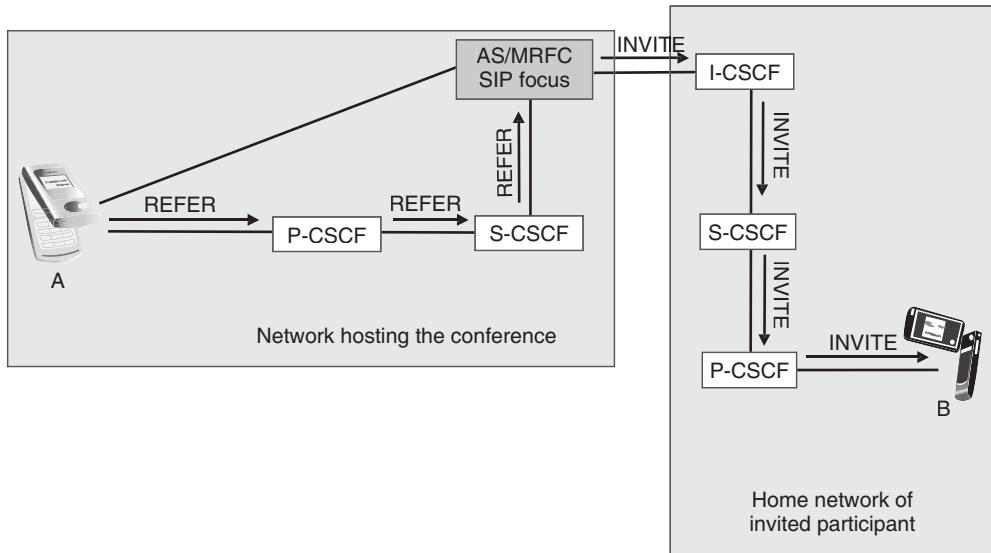
If we assume that Theresa is already participating in the above created conference and is missing Tobias there, she can set up a second call from her phone towards Tobias and tell him, that the conference is already active. In order to make joining easier for Tobias, Theresa will transfer him into the conference, which means she replaces the ongoing call between her and Tobias with a new connection towards the conference. Procedures for transferring calls are described, for example, in Section 9.3.10 and Section 12.10.

There is also the possibility that Theresa does not call Tobias directly, but simply sends a REFER request to the conference AS/MRFC, asking the focus to get Tobias into the conference, which is shown in Figure 8.4:

```
REFER sip:litalk@focus.home1.fr SIP/2.0
From: "Theresa" <sip:theresa@home2.hu>;tag=xyz
To: "Literature Talk" <sip:litalk@focus.home1.fr>;tag=abc
Refer-To: "Tobias" <sip:tobias@home1.fr>
```

As shown here, the Refer-To header instructs the conference AS/MRFC to set up a call with Tobias.

Based on the conference policy, the conference AS/MRFC will now either call Tobias directly, by sending an INVITE request to Tobias's phone, or it will send another REFER request to Tobias, in order to instruct Tobias to set up a call to the conference AS/MRFC.



**Figure 8.4** Referring users into a conference via conference AS/MRFC

```

REFER sip:tobias@home1.fr SIP/2.0
From: "Literature Talk" <sip:litalk@focus.home1.fr>
To: "Tobias" <sip:tobias@home1.fr>
Referred-By: "Theresa" <sip:theresa@home2.hu>
```

We see here that the Referred-By header indicates to Tobias that Theresa initiated the invitation to him. If Tobias accepts the SIP REFER request, his phone will automatically set up a call to the conference AS/MRFC, as described in Section 8.2.2.1.

### 8.2.3 Conference State Event Package

Every participant can subscribe to the conference state event package, in order to get more information about the other participants within a conference. In order to do so, the participants send a SIP SUBSCRIBE request to the conference URI, indicating the conference state event package and the duration of the subscription:

```

SUBSCRIBE sip:litalk@focus.home1.fr SIP/2.0
Event: conference
Expires: 3600
Allow: application/conference-info+xml
```

The conference AS/MRFC accepts the subscription by sending a 200 (OK) response to the SUBSCRIBE request, which will create a SIP dialog between the subscribing participant and the conference AS/MRFC. Immediately afterwards the conference AS/MRFC will send a notification to the participant, including the current conference status:

```
 NOTIFY sip:tobias@home1.fr SIP/2.0
Event: conference
Content-Type: application/conference-info+xml
<?xml version="1.0" encoding="UTF-8"?>
<conference-info xmlns="urn:ietf:params:xml:ns:conference-info"
  entity="sip:litalk@focus.home1.fr" state="full" version="1">
  <conference-description>

    <subject>Literature Talk</subject>

    <service-uris>
      <entry>
        <uri>http://www.home1.fr/litalk</uri>
        <purpose>web-page</purpose>
      </entry>
    </service-uris>
  </conference-description>

  <conference-state>
    <user-count>2</user-count>
  </conference-state>

  <users>
    <user entity="sip:tobias@home1.fr" state="full">
      <display-text>Tobias</display-text>
      <endpoint entity="sip:tobias@mobilepone.home1.fr">
        <media id="1">
          <display-text>audio</display-text>
          <type>audio</type>
          <status>sendrecv</status>
        </media>
      </endpoint>
    </user>

    <user entity="sip:theresa@home2.hu" state="full">
      <display-text>Theresa </display-text>
      <endpoint entity="sip:theresa@laptop87.home2.hu">
        <media id="1">
          <type>audio</type>
          <status>sendrecv</status>
        </media>
      </endpoint>
    </user>
  </users>
</conference-info>
```

The example shown here is not complete and only lists a very basic representation of all the information that can be transported within the conference state notification, such as:

- name of the conference;
- service URIs, as in this example a link to a webpage where further information on the subject of this conference can be found;
- detailed information about the participating users, such as their SIP addresses and names as indicated in their SIP INVITE requests, their contact addresses and media characteristics.

This information can be used at the subscribing participants UE to give the participant detailed information about the ongoing conference.

The subscription to the conference state event is automatically ended once the subscribing participant leaves the conference or the conference is ended.

#### 8.2.4 *Floor Control*

Floor control is an optional feature for conferences, by which the right to speak can be explicitly given to specific participants. Within IMS conferences, floor control is performed by means of BFCP, which allows the following functionalities:

- a conference participant can request the floor;
- a conference moderator can grant the floor based on a floor request or deny it;
- all conference participants get informed about the current status of the floor.

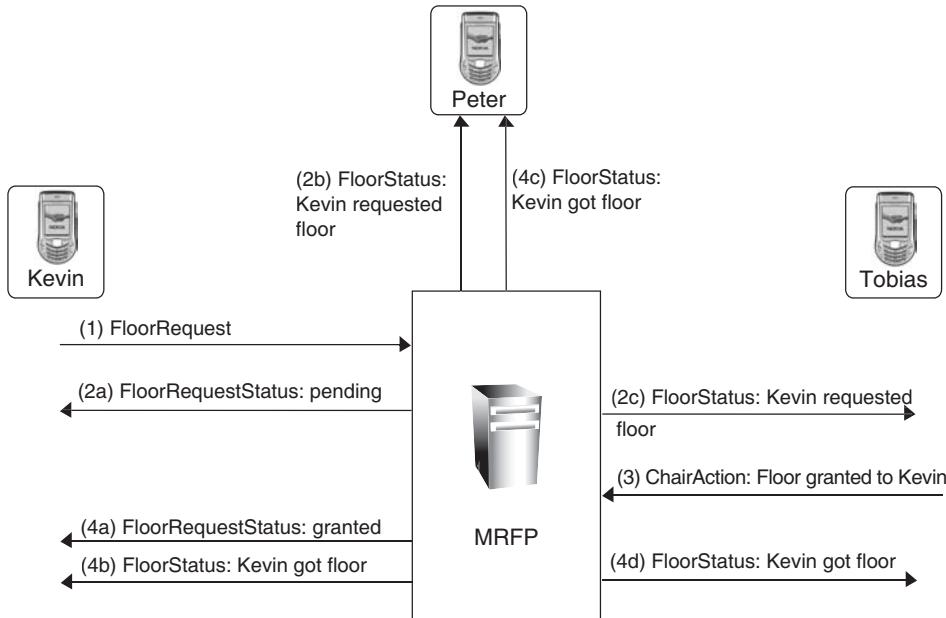
Let's assume there are three participants in a conference:

- Tobias, who is the conference moderator;
- Peter, who currently has the right to speak, i.e. he holds the floor;
- Kevin, who wants to speak.

We also assume that all three of them have sent a FloorQuery message to the MRFP when they joined the conference. The MRFP will therefore update the participants with the status of the floor by sending BFCP FloorStatus messages whenever the status of the floor changes.

In order to obtain the floor, Kevin sends a BFCP FloorRequest message to the MRFP. The MRFP will return a BFCP FloorRequestStatus message, indicating that the floor request is pending.

At the same time, the MRFP will send a new BFCP FloorStatus message to Tobias and Peter, indicating that Kevin requested the floor. Tobias, as the conference moderator, accepts the request from Kevin and therefore his phone sends a BFCP ChairAction message to the conference AS/MRFC, indicating that the floor has been granted to Kevin.



**Figure 8.5** Floor control with BFCP

The conference AS/MRFC now takes the floor away from Peter and gives it to Kevin and in parallel sends out:

- a BFCP FloorRequestStatus message to Kevin, indicating that the floor was granted;
- a BFCP FloorStatus message to all three participants, indicating that the floor was taken away from Peter and given to Kevin.

After this the conference participants will hear what Kevin is saying into the microphone of his phone. This example is shown in Figure 8.5.



# 9

# Multimedia Telephony

## 9.1 Introduction

We already have telephony so that we can talk to each other. We have video telephony so that we can see the other person or we can share what we see or have seen. We have messaging so that we can send instant notes, pictures, video clips etc. These can take place simultaneously but services typically use different applications in the device or bundled application is proprietary solution itself. IMS multimedia telephony service is the answer to build interoperating communication service that can provide a truly global service, in the spirit of good old telephony.

IMS multimedia telephony is a blended multimedia service suite. It allows users to establish communications between them and enrich that by enabling supplementary services. Typically it is a service using speech and speech combined with other media components, but the service is not limited to always include speech, it also caters for other media or combinations of media (e.g. text and video). IMS Multimedia Telephony service includes the following standardized media capabilities:

- full duplex speech;
- real time video (simplex, full duplex), synchronized with speech if present;
- text communication;
- file transfer;
- video clip sharing, picture sharing, audio clip sharing. Transferred files may be displayed/replayed on receiving terminal for specified file formats.

Based on these capabilities different types of communication use cases could be realized e.g. voice calls, voice call enriched with video (unidirectional or bidirectional), basic voice call enriched with sharing, basic voice call enriched with file transfer, basic voice call enriched with video and text communication, basic voice call enriched with video and text communication and file transfer, text communication (chat), File transfer (MSRP), sharing (image, video). To govern incoming and outgoing multimedia telephony

communication supplementary services mainly inherited from legacy PSTN, ISDN networks are standardized.

## 9.2 Multimedia Telephony Communication

### 9.2.1 SIP and IMS Multimedia Telephony

The basic communication between the users of IMS multimedia telephony is controlled via SIP procedures for establishing, handling and terminating a multimedia session. This will be shown in more detail in Chapter 12.

Due to the usage of SIP, IMS multimedia telephony can be used in several different ways:

- between users who all support the IMS multimedia telephony communication service, which allows them the full and rich experience of the service, including a fixed set of media and standardized supplementary services;
- between one or more users making use of the IMS multimedia telephony communication service and other users who use basic SIP terminals, supporting SIP style voice over IP (VoIP) services. In this case IMS multimedia telephony offers interworking towards the SIP style VoIP terminals, so that basic communication and most of the supplementary services are still available for all communication partners;
- between one or more users making use of the IMS multimedia telephony communication service and other users who are connected via e.g. CS terminals. In this case IMS multimedia telephony offers a perfectly adapted interworking between CS communications and the IMS service, so that both basic as well as supplementary services work in an ideal manner.

### 9.2.2 IMS Communication Service Identification (ICSI) and Telephony Application Server (TAS)

In addition to the basic SIP procedures, IMS multimedia telephony makes use of the IMS communication service identification (ICSI) mechanism, which offers a more efficient treatment of the service within IMS. Based on ICSI, SIP messages related to IMS multimedia telephony are:

- routed with preference towards those terminals of a called user, which support the IMS multimedia telephony service;
- routed automatically to the Telephony Application Server (TAS) of the user's home network operator. The TAS provides all the network based control for IMS multimedia telephony and in particular performs those supplementary services which are executed within the network, such as e.g. communication diversion;
- charged based on a specific charging model, that the network operator can apply for IMS multimedia telephony;
- interworked towards the CS network, whereby all the additional capabilities of IMS multimedia telephony are mapped best to the relating CS capabilities.

Detailed examples for the usage of ICSI can be found in Sections 11.9.3 and 12.3.9.

An example of interworking between CS and IMS multimedia telephony can be found in Section 13.3.3.

## 9.3 Supplementary Services

The following supplementary services are supported in 3GPP Release 7 multimedia telephony service. Planned 3GPP Release 8 enhancements are briefly introduced at end of this section.

- Originating Identification Presentation (OIP) service provides the terminating party with the identity of the originating party.
- Originating Identification Restriction (OIR) service enables the originating user to prevent presentation of its identity information to the terminating user.
- Terminating Identification Presentation (TIP) service provides the originating party with the possibility of receiving identity information in order to identify the terminating party.
- Terminating Identification Restriction (TIR) is a service offered to the connected party which enables the connected party to prevent presentation of the terminating identity information to the originating party.
- Communication Hold (HOLD) service enables a user to suspend media within a session, and resume that media at a later time.
- Conference (CONF) service enables a user to participate in and control a simultaneous communication involving a number of users.
- Communication Diversion services (aka communication forwarding) service allows the user to re-direct an incoming request that fulfils certain provisioned or configured conditions to another destination.
- Communication Barring (CB) service allows users to selectively block session attempts.
- Explicit Communication Transfer (ECT) provides a party involved in a communication to transfer that communication to a third party.
- Message Waiting Indication (MWI) service enables the application server to indicate to the subscriber, that there is at least one message waiting.

### 9.3.1 *Communication Barring*

Communication barring service allows users to selectively block session attempts. Barring service can be further divided into three main classes: Incoming Communications Barring (ICB), Outgoing Communication Barring (OCB) and Anonymous Communication Rejection (ACR). The ICB is a service that rejects incoming communications that fulfil certain provisioned or configured conditions on behalf of the terminating user. The ACR is a particular case of the ICB service, that allows barring of incoming communications from an anonymous originator on behalf of the terminating user. The OCB is a service that rejects outgoing communications that fulfil certain provisioned or configured conditions on behalf of the originating user. For example, the user can bar originating video call attempts while roaming, bar originating phone calls to specific numbers between 8am and 4pm, bar anonymous terminating messaging sessions attempts or bar incoming calls from numbers +358401234567, +358501234567 and allow the rest of the calls. These

and a great number of other conditions and rules have been standardized and the user is able to configure these settings using XCAP protocol as described in Section 5.8.1. Once the rules are set then the TAS will enforce communication according to user's preferences.

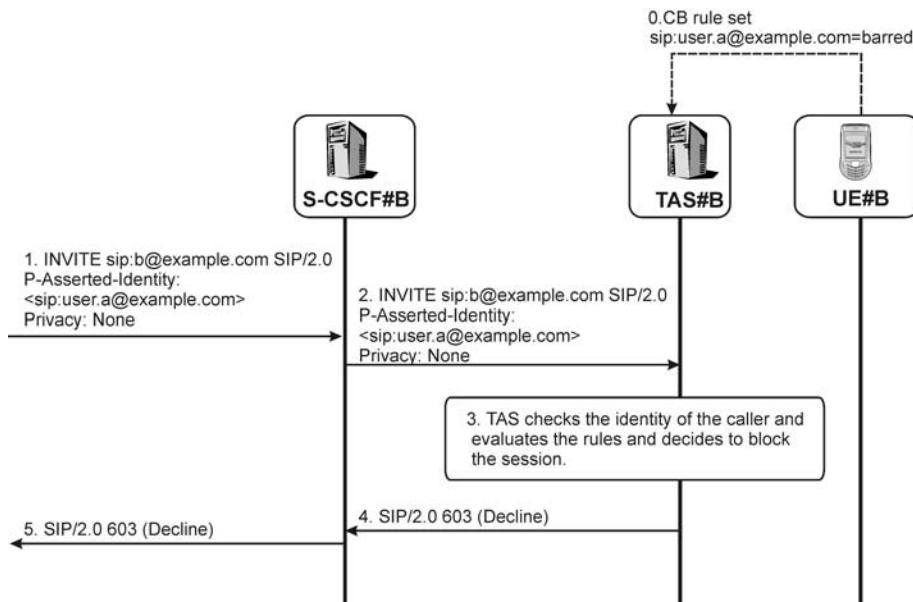
In Figure 9.1 User B uses the XCAP protocol to bar all communication requests from User A. When User A later makes a communication attempt towards User B the TAS providing service for them utilizes the stored information and based on the network asserted identity of the caller it blocks the request with appropriate SIP error response, 603 Decline which means that the user explicitly does not wish to take part in communication.

Figure 9.2 shows an example of outgoing call barring. First XCAP protocol is used to configure wanted policy to the network, here it is block all outgoing communication attempts towards domain 'expensive.com'. After that the TAS declines all communication attempts to all targets in that domain.

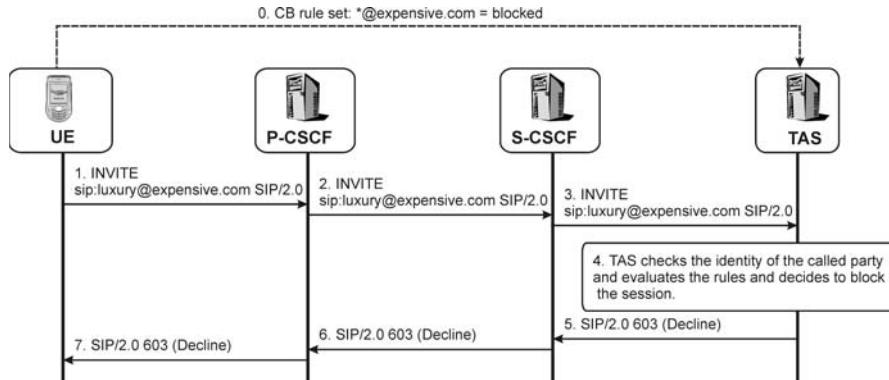
### 9.3.2 Communication Diversion

Communication diversion (aka communication forwarding) service allows user to re-direct an incoming request that fulfils certain provisioned or configured conditions to another destination. Baseline for this service is inherited from established diversion services in PSTN/ISDN networks. Namely the following services are included from old service world:

- Communication Forwarding Unconditional (CFU) service enables a served user to have the network redirect to another user communications which are addressed to the served user's address.



**Figure 9.1** Example of incoming communication barring supplementary service



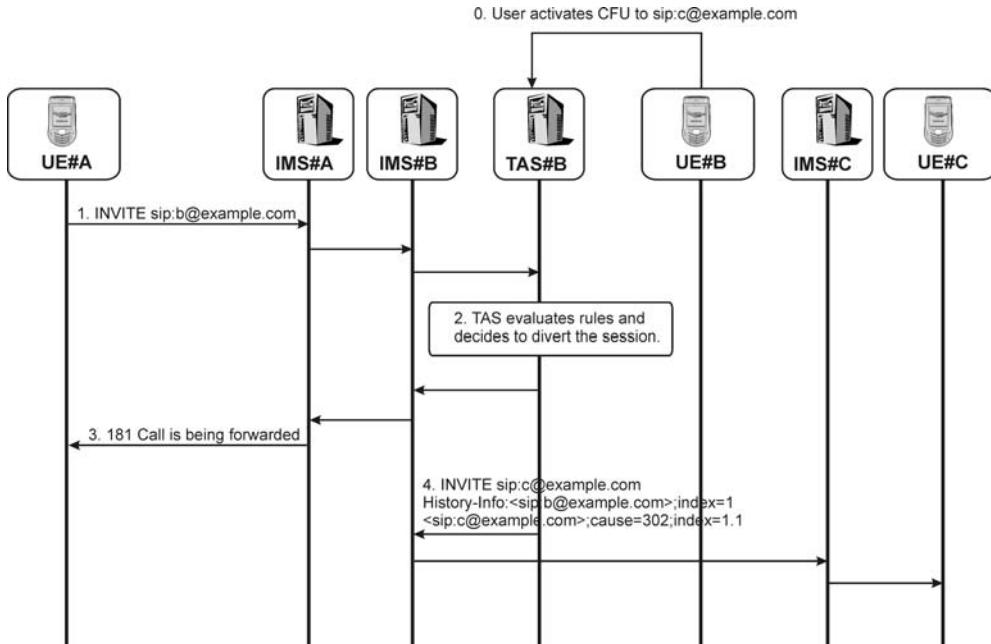
**Figure 9.2** Example of outgoing communication barring supplementary service

- Communication Forwarding Busy (CFB) service enables a served user to have the network redirect to another user communications which are addressed to the served user's address and meet busy.
- Communication Forwarding No Reply (CFNR) service enables a served user to have the network redirect to another user communications which are addressed to the served user's address, and for which the connection is not established within a defined period of time.
- Communication Forwarding on Not Logged-in (CFNL) service enables a served user to redirect incoming communications which are addressed to the served user's address, to another user (forwarded-to address) in case the served user is not registered (logged-in).
- Communication Deflection (CD) service enables the served user to respond to an incoming communication by requesting redirection of that communication to another user (before ringing and after ringing).
- Communication diversion on mobile subscriber not reachable (CFNRC) service enables a user to have the network redirect all incoming communications, when the user is not reachable (e.g. there is no IP connectivity to the user's terminal), to another user.

In addition to previously listed conditions new conditions are defined as follows:

- Communication Forwarding depending on called user's presence status.
- Communication Forwarding depending on the calling user's identity or lack of identity.
- Communication Forwarding depending on media including in the incoming session.
- Communication Forwarding depending on time of the call.

For example the user could create rules to divert incoming audio call to CS domain, divert incoming video call when user's presence status is 'silent' to video mail box, divert MSRP session to client associated to fixed device, divert unanswered session to 3rd party if no answer in 20 seconds. These and a great number of other possibilities are enabled by standards and the user is able to configure these settings using XCAP protocol as described in Section 5.8.2. Once the rules are set then the TAS will enforce communication according to user's preferences.

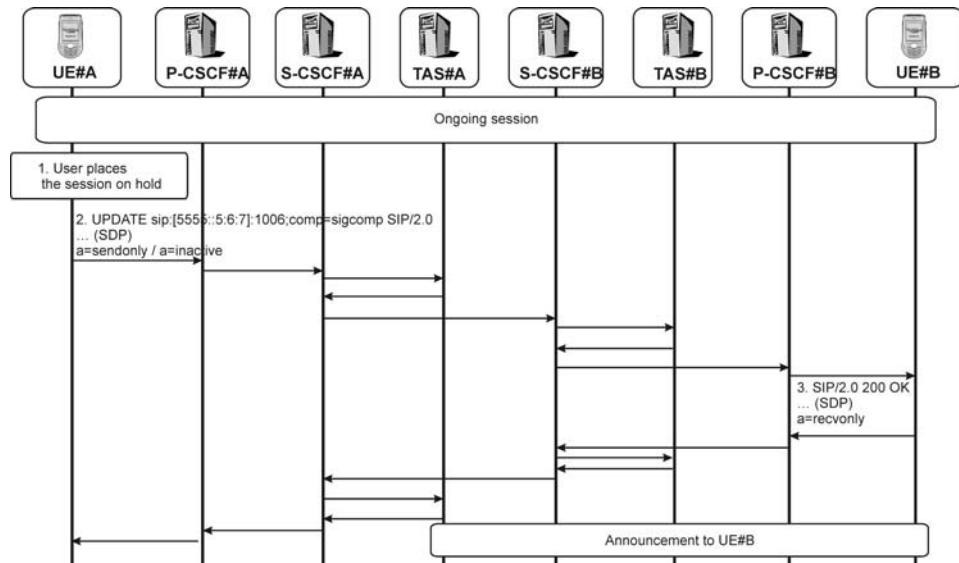


**Figure 9.3** Example of communication diversion supplementary service

Figure 9.3 shows an example when User B wants to divert all incoming communication to User C (`c@example.com`). First the device uses XCAP protocol to set-up diverting rules to the TAS. Once the TAS serving User B receives an incoming communication attempt it checks if the served user has uploaded any settings for an incoming session treatment and based on the stored information it makes a decision to divert the request to User C. The sent request to User C will contain additional information showing that the call was originally targeted to User B and it additionally conveys the reason for diversion (these are carried in the History-Info header). Moreover the TAS informs the caller that the communication is being forwarded (SIP Response 181 Call is being forwarded).

### 9.3.3 Communication Hold

The Communication Hold supplementary service enables a user to suspend the media stream(s) of an established IP multimedia session, and resume the media stream(s) at a later time. When a user wishes to place communication on hold the UE sends either UPDATE or RE-INVITE request towards other party. The Attribute line in the SDP of the request having value ‘inactive’ or ‘sendonly’ indicates that the sender is not willing to receive media stream(s) from the other party as shown in Figure 9.4. When a user wished to resume held media stream(s) the UE sends UPDATE or RE-INVITE request towards the other party. Attribute line in the SDP of the request having value ‘sendrecv’ or ‘recvonly’ indicates that the sender is again willing to receive media stream(s) from the other party.



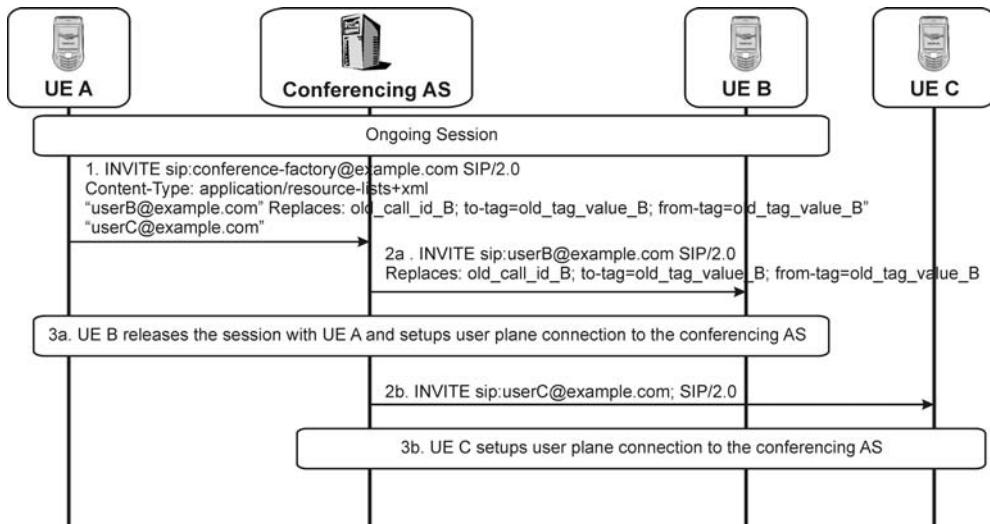
**Figure 9.4** Example of communication hold supplementary service

### 9.3.4 Conference

Conference service enables a subscriber to communicate with one or more subscribers simultaneously. This supplementary service re-uses a solution developed for standalone IMS conferencing and the service is covered thoroughly in Chapter 8 therefore this service is described here only in brief. Figure 9.5 shows an example of how a user can transform ongoing 1-1 session to multimedia conference. This can be seen as an equal service to existing GSM supplementary services known as multiparty conference (MPTY).

For example User A and User B are having a conversation as peer-to-peer via the IMS and then they realize that they wish to add an additional user, User C. User A takes action and uses her UE to create an ad-hoc conference with three participants by sending an INVITE to SIP-URI known as conference-factory URI (e.g. `sip:conference-factory@example.com`). This request contains intended participants: User B and User C. In addition, it contains instructions towards the UE B that this new incoming request will replace the existing session between UE A and UE B. This INVITE gets routed using normal IMS routing principles to an application server providing the conferencing functionality. When the application server gets this request it forms two independent session requests towards UE B and UE C (Steps 2a and 2b). Once the UE B receives new INVITE it learns that this is a request to join a conference and to release the existing session with UE A. Similarly the UE C receives the INVITE and learns that this is an invitation to a conference. After additional SIP signalling and user-plane setup Users A,B,C have joined the conference and can start discussing and sharing multimedia content.

However, it is important to acknowledge that IMS multimedia telephony conference service does not require an existing IMS multimedia telephony session prior to conference creation, so this is a major differentiator compared to the existing GSM service. Examples of out-of-blue conference sessions are given in Chapter 8.



**Figure 9.5** Example of conference supplementary service

### 9.3.5 Message Waiting

Message Waiting Indication (MWI) service enables the application server to indicate to the subscribe, that there is at least one message waiting in the message account. To activate this service the UE sends SIP SUBSCRIBE to the network having suitable value in Expire header e.g. one day. To terminate the message waiting indication service the UE sets Expire header value to 0 or lets subscription to expire.

```
SUBSCRIBE sip:tobias@home1.fr SIP/2.0

Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;comp=sigcomp;branch=4uetb
Route: <sip:[5555::a:b:c:d]:7531;lr>
Route: <sip:orig@scscf1.home1.fr;lr>
From: "Tobias" <sip:tobias@home1.fr>;tag=sipuli
To: "Tobias" <sip:tobias@home1.fr>
Event: message-summary
Expires: 86400
Accept: application/simple-message-summary
Contact: <sip:[5555::1:2:3:4]:1357>
Content-Length: 0
```

The subscription is intended for an event named ‘message-summary’, that uniquely defines that the user is interested in receiving message waiting indications; this is identified in the Event header of the request.

The request URI identifies the user whose message account information is requested and, therefore has to be set to subscriber’s public user identity or public service identity of the message account depending on the operator’s policy on how to access the MWI service and must also be indicated in the To header.

The Accept header indicates that only information of the type ‘application/simple-message-summary’ can be processed by the UE for this subscription, which is a simple text-based format for message waiting indications.

The application server like TAS receives this SUBSCRIBE request and will check whether the requesting user is allowed to subscribe to this particular message account. As Tobias is subscribing to his own account in this case, this is allowed. Therefore, the application server will immediately:

- return a 200 (OK) response for the SUBSCRIBE request, indicating that the subscription was successful;
- generate information regarding the current state of the message account. This initial information only contains a summary of message account. Further notifications may contain extended information (e.g. most important headers of the message To, From, Subject, Date, Priority);
- send the generated information in a NOTIFY message toward the subscriber (in this case Tobias’s UE).

```
NOTIFY sip:tobias@home1.fr SIP/2.0
From: <sip:tobias@home1.fr>;tag=31415

To: <sip:tobias@home1.fr>; tag=sipuli
Subscription-State: active;expires= 86399
Event: message-summary
Content-Type: application/simple-message-summary
Content-Length: (...)
Messages-Waiting: yes
Message-Account: sip:tobias@home1.fr
Voice-Message: 2/1 (1/0)
Video-Message: 0/1 (0/0)
```

Not all the information that is included in the NOTIFY request is shown here – the above headers are only those that are necessary to understand the nature of the message-summary event.

In this example it is assumed that the message account at the moment of subscription has three voice messages (two new and one old, with one new message being urgent), one old video message.

This solution is based on SIP Event Framework as defined in RFC3265 and the Event package defined for message waiting indication RFC3842. 3GPP has defined some restrictions in MWI specification 3GPP TS 24.406 compared to RFC3842. 3GPP TS 24.406 supports limited coding of the message types (voice-message, video-message, fax-message, pager-message, multimedia-message, text-message, None) and limited information of stored messages (e.g. To, From, Subject, Date, Priority).

It should be noted that 3GPP has not standardized how to fetch actual messages. For example, retrieving voice/video messages a user could dial her voice/video mail number. For text and image messages complete solution as described in Section 7.5.4 could be used.

### 9.3.6 Originating Identification Presentation

In mobile networks and ISDN networks users are able to see the identity of the caller on their display before answering the incoming call. This is enabled by a service called calling line identification presentation service (CLIP). The same functionality in IMS is provided with Originating Identification Presentation (OIP).

The public user identity is conveyed within three different headers within a SIP INVITE request:

- the From header, which can be set to any value the calling user wants to set it to;
- the P-Preferred-Identity header, in which the calling user can indicate one of its public user identities, i.e. the identity that it wants to be used for the call – see Section 12.2.4.3;
- the P-Asserted-Identity header, in which the authorized identity of the user is transported.

```
INVITE sip:theresa@home2.hu SIP/2.0
From: "Tobias" <sip:tobias@home1.fr>;tag=xyz
P-Preferred-Identity: <sip:tobias@home1.fr>
```

When setting up the call, the user indicates the public user identity in the P-Preferred-Identity header. The P-CSCF checks, based on its subscription to the user's registration state information (see Section 11.13.7), whether the indicated identity is a valid public user identity of the user:

- If the identity is valid, it replaces the P-Preferred-Identity header with the P-Asserted-Identity header, including the same value;
- If the identity is not valid, it removes the P-Preferred-Identity header and puts the value of the default public user identity (see Section 11.13.4) into the P-Asserted-Identity header.

```
INVITE sip:theresa@home2.hu SIP/2.0
From: "Tobias" <sip:tobias@home1.fr>;tag=xyz
P-Asserted-Identity: <sip:tobias@home1.fr>
```

The P-Asserted-Identity header, is used to convey a trusted identity of the caller, which is not only used for the OIP supplementary service, but also for identification of the authenticated user within the IMS network, e.g. for provisioning of dedicated services.

The P-Asserted-Identity header will be available in the terminated UE when the SIP signalling has traversed through a trusted signalling network and no operator has disabled OIP from the user. In case a CSCF detects that the next hop is untrusted, the CSCF will remove the P-Asserted-Identity header, which prevents the OIP supplementary service.

The From header may as well include the caller's public user identity and can be presented to the called user. Different to the P-Asserted-Identity header, the From header will always be delivered to the terminating phone, regardless of whether the SIP signalling is conveyed inside or outside any trust domain.

The From header is not checked and asserted by the network and therefore can include any value that the calling user wants to set it to. For example, both users could be members of an online community – the calling user then could indicate in the From header the nickname, that she uses in the community. Although this nickname is not known by the IMS of the calling user, the From header would still be transported to the other end, as it is not checked. Nevertheless, the nickname would not appear in the P-Asserted-Identity header, as it is always set to a valid public user identity of the user.

```
INVITE sip:theresa@home2.hu SIP/2.0
From: "MyNickname" <sip:mynickname@example.com>;tag=xyz
P-Asserted-Identity: <sip:tobias@home1.fr>
```

### 9.3.7 Originating Identification Restriction

Originating Identification Restriction (OIR) is a service that an originating user can use in order not to expose the user's identity to other parties. Users can activate this on a request basis or set default setting to the TAS within the user's IMS network.

In order to restrict the presentation of the identity in the From header, the From header must be set to 'anonymous'. If the calling user wants to apply OIR only for a specific call, the user can set the From header immediately to 'anonymous' when sending out the SIP INVITE request.

```
INVITE sip:theresa@home2.hu SIP/2.0
From: "anonymous" <sip:anonymous@example.com>;tag=xyz
```

It might well be that the user has subscribed to the OIR service in general and wants the service always to be invoked, regardless of what the originating UE indicates in the From header. In this case, the TAS of the calling user will apply the service, by changing the From header from the value set by the user to 'anonymous'. In order to achieve this, the TAS will act as a SIP Back-to-Back User Agent (B2BUA) – see e.g. Section 13.3.5.

The P-Asserted-Identity header nevertheless cannot be set to 'anonymous', as it is essential for the user's identification within the network, e.g. for provisioning of specific services. Therefore an additional SIP header called 'Privacy' is used in order to indicate that the originating user does not want the trusted identity to be shown to the called user. To achieve OIR for a single call, the calling user will include the 'Privacy' header set to the value 'id' in the initial INVITE request, which will force the terminating P-CSCF, i.e. the SIP proxy which sends the INVITE request to the called user, to remove the P-Asserted-Identity header from the SIP request.

```
INVITE sip:theresa@home2.hu SIP/2.0
P-Asserted-Identity: sip:tobias@home1.fr
Privacy: id
```

If the calling user has activated the OIR supplementary services on a permanent basis, the TAS will include the 'Privacy: id' header within the SIP INVITE request.

A user can permanently activate the OIR service by sending a XCAP message to the TAS, indicating that presentation should be restricted. The user can also deactivate

the service, by sending a XCAP message to the TAS, indicating that presentation is allowed.

### 9.3.8 Terminating Identification Presentation (TIP)

As long as the called user doesn't request otherwise, the called user's identity will be shown to the calling user.

When the calling user sets up the call, the public user identity of the called user must be set within the request URI of the SIP INVITE request, in order to enable the IMS network to route the call correctly. In addition to this, the calling user sets the SIP To header to either the public user identity of the called user or to an alternative identity of the called user, e.g. the user's nickname within an online community.

```
INVITE sip:theresa@home2.hu SIP/2.0
To: "YourNickname" <sip:nickname2@example.com>
```

This information cannot be influenced by the called user, as it is set before the called user is reached. Nevertheless, the called user will be identified by a P-Asserted-Identity header, which is sent in the first SIP response (for simplification we assume here that this is a 200 (OK) response) towards the originating user. The P-Asserted-Identity header is set by the P-CSCF of the called user, based on the value received in the P-Preferred-Identity header from the called terminal in the SIP response (see Section 9.3.6).

```
SIP/2.0 200 OK
To: "YourNickname" <sip:nickname2@example.com>
P-Asserted-Identity: sip:theresa_public2@home2.hu
```

As long as all IMS entities between the two users are trusting each other, the P-Asserted-Identity header will stay within the 200 (OK) response and will be delivered to the calling user's terminal, where it can be presented to the called user.

### 9.3.9 Terminating Identification Restriction (TIR)

The Terminating Identification Restriction (TIR) supplementary service enables the called user to restrict the presentation of the terminating public user identity in the same way as the originating user can restrict the presentation of the calling public user identity (as described for the OIR supplementary service – see Section 9.3.7).

In order to restrict the presentation of the identity, the called user adds a Privacy header with the value 'id' into every SIP response that it sends and that conveys the P-Preferred-Identity header. The P-CSCF of the called user then will include the P-Asserted-Identity header for the called user and will send the Privacy header forward unchanged. The P-CSCF of the calling user will remove the P-Asserted-Identity header, based on the presence of the 'Privacy: id' header and will not forward the terminating identification to the called user.

```
SIP/2.0 200 OK
To: "YourNickname" <sip:nickname2@example.com>
```

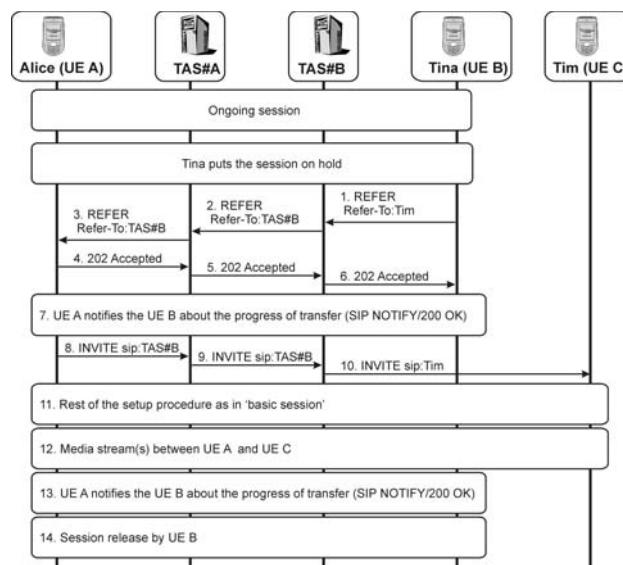
P-Asserted-Identity: sip:theresa\_public2@home2.hu  
 Privacy: id

The terminating user can also subscribe on a permanent basis to the TIR supplementary service, by sending a XCAP message to the called user's TAS, indicating that presentation of the terminating identification is restricted. Based on this, the TAS will add the 'Privacy: id' header in every response from the called user which includes the P-Asserted-Identity header.

### 9.3.10 Explicit Communication Transfer

Explicit Communication Transfer (ECT) service enables the user to leave the existing communication session and connect the other communication party to the another user. When a user wishes to transfer the ongoing communication she has basically two options: transfer the communication to a new party without knowing whether they will accept the communication request (this is also known as blind transfer) or first contact a new party and inform them that soon they will get a request to communicate with another person (this is also known as consultative transfer).

Figure 9.6 depicts on ECT example where user Alice has called a service desk and is asking assistance with a new device she bought yesterday. Tina, who took the call at the service desk is not able to provide a decent answer but she knows that her colleague must know the answer and decides to transfer the call to him (Tim) using ECT supplementary service. In a blind transfer case Tina's UE constructs SIP REFER request and sets Alice's contact information to Request-URI field and sets Tim's identity in Refer-To header and in addition Referred-by header could be added to transport Tina's identity.



**Figure 9.6** Example of explicit call transfer

---

```
REFER sip: alice@example.com SIP/2.0
Refer-To: sip:tim@servicedesk.com
Referred-By: sip:tina@example.com
```

The application server providing service for Tina intercepts the request and modifies the request in such a way that causes Alice to create a new session towards this application server instead of making direct connection with Tim. The main reason for doing this trick is charging. It is expected that Alice is responsible for paying the new leg towards Tim and without having Alice's application server involved in the new communication it would be impossible to create proper charging records.<sup>1</sup> In addition, the AS verifies that Referred-By header contains the valid identity of Tina and ensures that Tina does not have communication barring active towards the new target. After modification the request towards Alice could look like this

```
REFER sip: unique.identifier1@example.com SIP/2.0
Refer-To: unique.identifier1@example.com
Referred-By: sip:tina@example.com
```

Once the request arrives to Alice's device it can ask Alice permission to establish a new session to given address (unique.identifier1@example.com). Here it is assumed that transfer is acceptable and the UE sends 202 Accepted response back and initiates a new session towards the AS.

```
INVITE sip: unique.identifier1@example.com SIP/2.0
```

The AS recognizes that this incoming INVITE is related to an earlier request received from Tina and it takes further action to send modified INVITE towards Tim as follows:

```
INVITE sip: sip:tim@servicedesk.com SIP/2.0
Referred-By: sip:tina@example.com
```

Tim's UE gets the request and the session will be established between Alice and Tim via the AS providing the service to Tina. The REFER request will also create implicit subscription between Alice's UE and Tina's UE and this subscription is used to convey information on how the session transfer progresses. For example, it can be used to trigger session release from Tina's UE when the communication transfer has succeeded.

---

<sup>1</sup> Due to charging reasons consumers do not typically use ECT service because after service execution they do not have the means to stop communication. In other words they lose control of the call.

# Part III

## Detailed Procedures



# 10

## Introduction to Detailed Procedures

This part gives a detailed example of Session Initiation Protocol (SIP) and Session Description Protocol (SDP) related procedures in the Internet Protocol Multimedia Subsystem (IMS). Signalling procedures, protocol messages and their elements are described and explained based on an example of IMS registration and a subsequent IMS Multimedia Telephony session between two users. Additionally, it will be shown in detail how voice call continuity works.

The reader will see how IMS signalling works and how previously described concepts and architecture are realized at the protocol level. Nevertheless, this part concentrates on the straight forward examples and does not handle error or abnormal procedures in detail.

To give a better understanding of the procedures applied, each chapter of this part is split into several sections which concentrate on different concepts, such as routing, authentication or media negotiation. Each chapter will describe those parts of individual SIP and SDP messages that are necessary for their understanding. An overview section is included in each chapter to give an introduction to the basic operation. At the end of each chapter the related standards and specifications are listed, to allow the interested reader to obtain more detail by reading the base specifications.

### 10.1 The Example Scenario

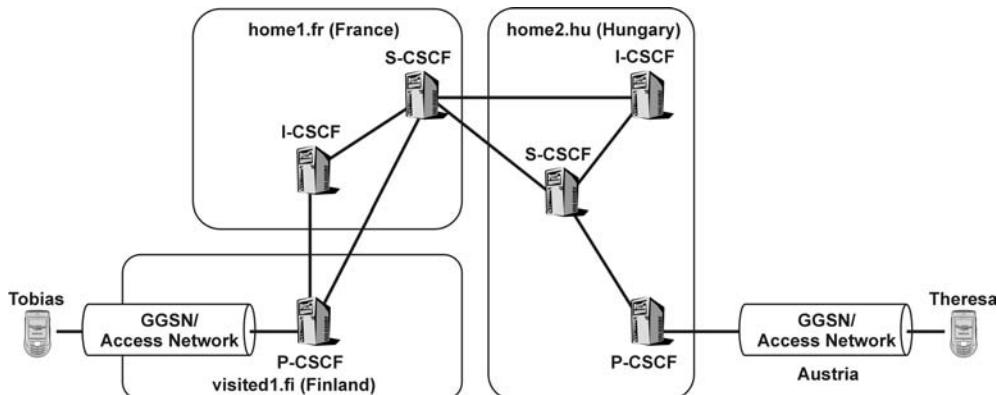
This part is based on the assumption that both users are attached to the General Packet Radio Service (GPRS), which is used as the example of access technology throughout.

Tobias, who is a student in France and currently visiting Finland, is calling his sister Theresa, who is working in Hungary and currently on a business trip to Austria (see Table 10.1 and Figure 10.1).

Tobias's home operator is located in France. As he is roaming in Finland, the Finnish operator provides the Proxy Call Session Control Function (P-CSCF), as the home operator and the Finnish operator have signed an IMS roaming agreement. Consequently,

**Table 10.1** Location of CSCFs and GPRS access for the example scenario

User	Home operator S-CSCF location	P-CSCF location	GPRS access
Tobias	France	Finland	Finland
Theresa	Hungary	Hungary	Austria



**Figure 10.1** The example scenario

the Gateway GPRS Support Node (GGSN) that Tobias is using is also located in Finland.

Theresa's home operator in Budapest has no IMS roaming agreement with the operator in Austria. Therefore, her terminal gets attached to the P-CSCF in her Hungarian home network, where the GGSN is also located. Theresa's access to the IMS is based on the GPRS-level roaming agreement between the operators of her home network and the visited network.

It is assumed that Theresa has already registered her SIP URI (uniform resource identifier), `sip:theresa@home2.hu`, as Tobias is just switching on his mobile phone. He wants to call his sister to show her one of the beautiful wooden buildings in Oulu and, therefore, points his camera, which is connected to the phone, toward the building. In parallel with this, his phone will also send a second video stream, showing his face to Theresa. The built-in camera of his phone records this second stream. However, Tobias first has to register his public user identity, `sip:tobias@home1.fr`, before he can call his sister.

In Chapter 13 the configuration is different, in order to better describe the procedures for IMS voice call continuity. The configuration will be described in detail in Section 13.1.

## 10.2 Base Standards

The following specifications define the basic procedures and architecture as used in the following chapters:

3GPP TS 23.228	IP Multimedia Subsystem (IMS).
3GPP TS 24.229	IP Multimedia Call Control Protocol based on SIP and SDP.
RFC3261	SIP: Session Initiation Protocol
RFC4566	SDP: Session Description Protocol
3GPP TS 24.930	Signalling flows for the session setup in the IMS based on SIP and SDP



# 11

## An Example of IMS Registration

### 11.1 Overview

Session Initiation Protocol (SIP) registration is performed in order to bind the Internet Protocol (IP) address that is currently used by the user and the user's public user identity, which is a SIP URI (uniform resource identifier). If Tobias wants to call Theresa, he will send a SIP INVITE request to her address – i.e., *sip:theresa@home2.hu*; he does not need to be aware of which terminal Theresa is using. The INVITE then gets routed to Theresa's registrar, which is located in *home2.hu*. This registrar became aware of Theresa's current terminal address during her registration and will replace the address *sip:theresa@home2.hu* with the registered contact, which is an IP address. Afterwards, the request can be routed to Theresa's terminal.

Therefore, even for nonIMS cases, Theresa needs to be registered at a SIP registrar so that her current terminal address can be discovered. The IP Multimedia Subsystem (IMS) couples more functionality with SIP registration procedures, which makes it necessary for Tobias to register as well, before he can call his sister.

The following procedures are performed during Tobias's IMS registration (see Figure 11.1):

- The UE gets configured with general IMS related configuration parameters. This configuration happens possibly only once per UE and does not need to be repeated for every registration – Section 11.2;
- A dedicated signalling Packet Data Protocol (PDP) context is established between Tobias's User Equipment (UE) and the Gateway GPRS Support Node (GGSN) in the case of General Packet Radio Service (GPRS) – Section 11.3;
- The UE discovers the address of the Proxy Call Session Control Function (P-CSCF), which it uses as a SIP outbound proxy during registration and for all other SIP signalling while it is registered – Section 11.4;
- The UE sends a REGISTER message to Tobias's home network to perform SIP registration for Tobias's public user identity – Section 11.5.3;

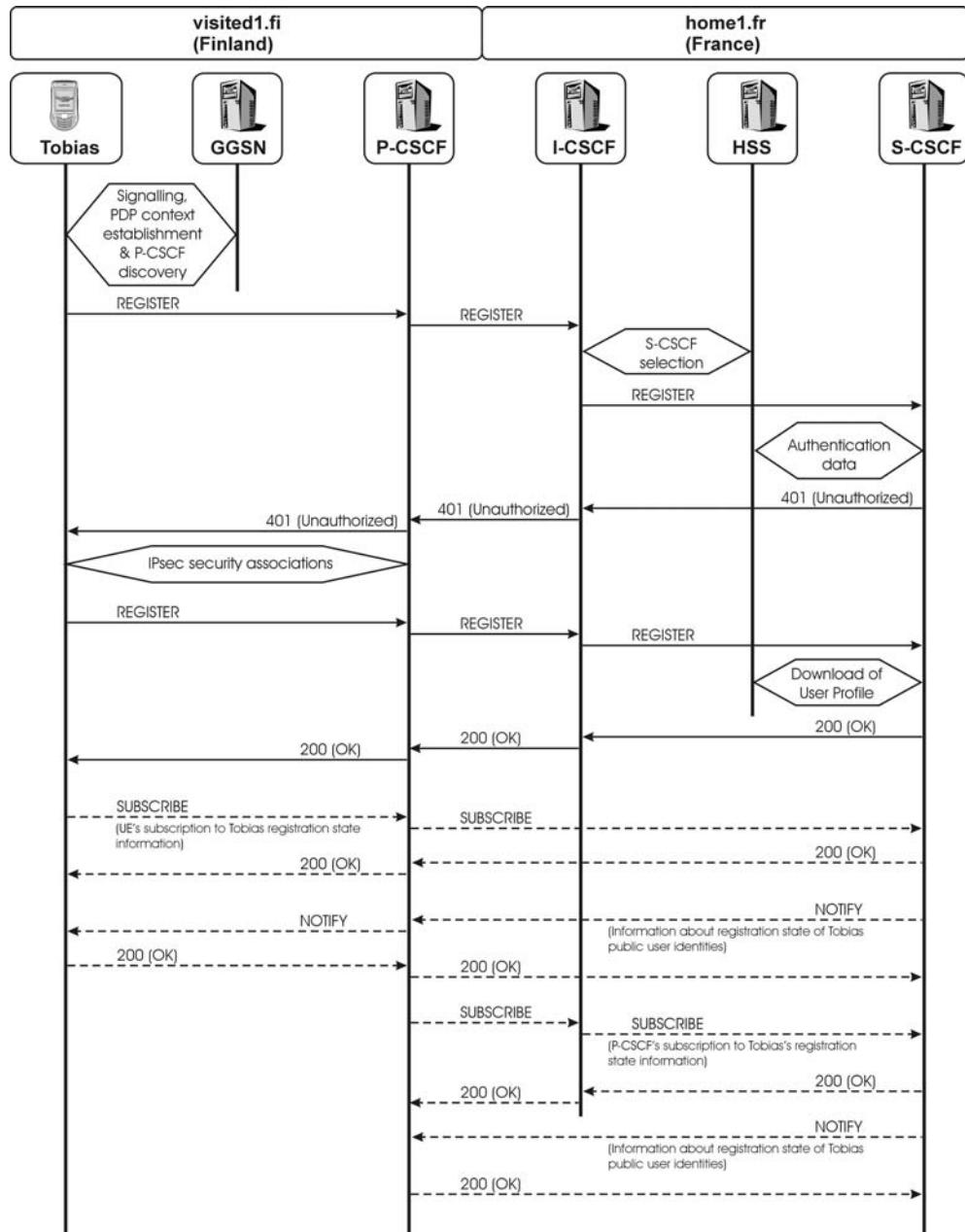


Figure 11.1 Initial registration flow

- The Interrogating-CSCF (I-CSCF) selects the Serving-CSCF (S-CSCF) that serves the user while it is registered – Section 11.5.5;
- The S-CSCF downloads the authentication data of the user from the Home Subscriber Server (HSS) – Section 11.6.4;
- The UE and the P-CSCF agree on a security mechanism – Section 11.8;
- The UE and the network (S-CSCF) authenticate each other – Section 11.6;
- IP Security (IPsec) associations between the UE and the P-CSCF are established – Section 11.7;
- SIP compression starts between the UE and the P-CSCF – Section 11.10;
- The UE learns the route to the S-CSCF – Section 11.5.8;
- The S-CSCF learns the route to the UE – Section 11.5.9;
- The S-CSCF downloads the user profile of the user from the HSS – Section 11.5.6;
- The S-CSCF registers the default public user identity of the user – Section 11.5.6;
- The UE registers the supported IMS communication services at the S-CSCF – Section 11.12;
- The S-CSCF may, based on the user profile, implicitly register further public user identities of the user – Section 11.13;
- The UE becomes aware of all public user identities that are assigned to Tobias and his current registration state – Section 11.13;
- The P-CSCF becomes aware of all public user identities that are assigned to Tobias and his current registration state – Section 11.13.

As a consequence of all these required basic actions, Tobias would not have been able to send the INVITE to his sister had he not registered earlier.

## 11.2 Initial Parameters and IMS Management Object

In order for Tobias's UE to construct the SIP REGISTER request, it needs to become aware of a set of information, some of which is provided on the UICC, with which the UE is equipped (see Section 11.13). In order to allow the network operator to configure the UE with further parameters and also give the operator the possibility of reading the configuration settings from the UE, the IMS Management Object (IMS MO) is used.

The IMS MO is transported to the UE via the OMA Device Management (OMA DM) protocol, which will not be further discussed here, as its usage is outside the IMS domain – it will only deliver the relevant configuration parameters.

The IMS MO provides the following settings to Tobias's UE:

- One or more pointers to GPRS access points configurations (ConRefs), which are configured to the phone by means of a separate OMA DM MO. The UE shall use one of these listed access points to connect to the 3GPP network in order to perform IMS registration on top of this connection – see Section 11.3;
- A preference of the operator, whether the UE should make use of a dedicated signalling PDP context for IMS or if a general PDP context is sufficient (PDP\_ContextOperPref). In this example the operator indicates that there is a preference for a dedicated signalling PDP context for IMS – see Section 11.3.

- A P-CSCF Address, which is only applicable for early IMS deployments in which IPv4 is used and where in some cases the network does not offer a P-CSCF discovery mechanism. In this example this field is not set, as the UE connects to the network via IPv6 to the 3GPP network and performs the P-CSCF discovery mechanism as described in Section 11.4.
- Operator specific values for the SIP Timers T1, T2 and T4 (Timer\_T1, Timer\_T2, Timer\_T4) which guard the retransmission of SIP transactions. In some cases, specific networks might want to change the default values of these timer, which is not assumed in this example.
- A list of IMS Communication Service Identifications (ICSIList) which indicates to the UE which ICSI values are supported by the network. The UE then shall only register those ICSI values which it supports and which are at the same time supported by the network – see Section 11.3. In this example the ICSI List contains three values:
  - one for the IMS Multimedia Telephony Communication Service (urn:urn-xxx:3gpp-service-ims.icis.mmtel);
  - one for a hypothetical IMS online game (urn:urn-xxx:other-vendor-service-ims.icsi.ongame);
  - one for a hypothetical IMS collaborative working service (urn:urn-xxx:other-vendor-service-ims.icsi.collawo), which is not supported by Tobias's UE.

Note that the ‘xxx’ values within these URNs will be replaced by numeric values, once these URNs have been registered with the Internet Assigned Number Authority (IANA).

In addition to these configuration parameters, the IMS MO allows the operator to read a set of information from the UE, i.e. when the UE receives an IMS MO, it will respond to it (via OMA DM protocol mechanisms) with the current settings and values available for the following fields:

- all the values available for the configuration fields described above, i.e. ConRefs, PDP\_ContextOperPref, P-CSCF\_Address, Timer\_T1, Timer\_T2, Timer\_T4 and ICSIList;
- the private user identity (Private\_user\_identity), all public user identities (Public\_user\_identity\_list) and the home network domain name (Home\_network\_domain\_name) – see Section 11.3.

### 11.3 Signalling PDP Context Establishment

Before Tobias's UE can start the IMS registration procedure, it needs to establish an IP connection with the network. In the case of GPRS such an IP connection is provided by either a dedicated signalling PDP context or a general purpose PDP context. The concepts and procedures for PDP context establishment and usage are described in Section 12.12.

Based on the configuration information in the IMS MO (Section 11.2) Tobias's UE knows

- which GPRS access point to use (from the ConRefs element in the IMS MO) to establish a GPRS connection;

- that a dedicated signalling PDP context for IMS must be established (from the PDP-ContextOperPref element in the IMS MO).

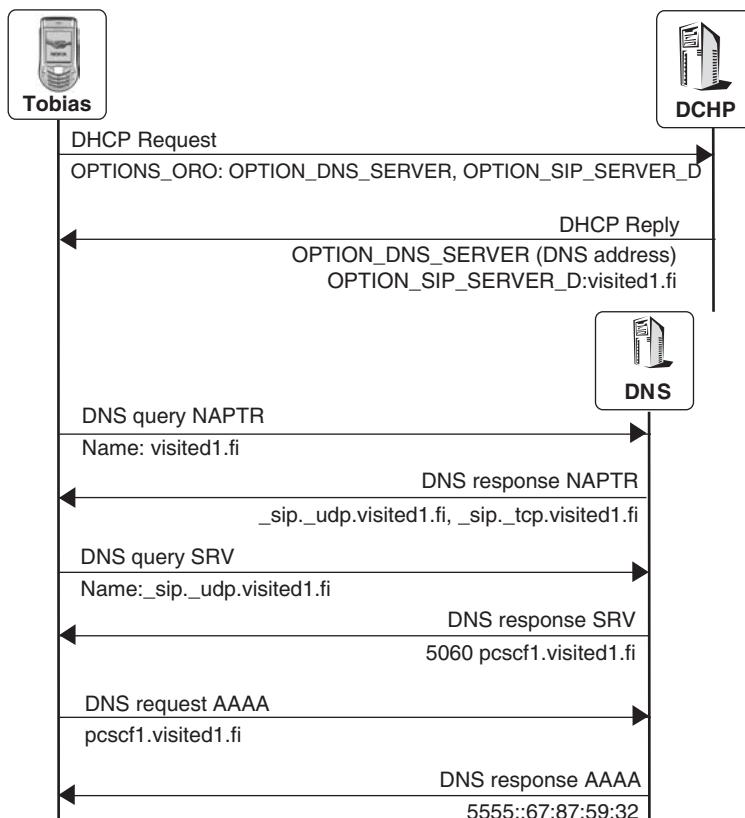
After the UE has established the signalling PDP context, it will be able to send SIP signalling over the air interface.

## 11.4 P-CSCF Discovery

### 11.4.1 Overview

The P-CSCF is the single entry point for all SIP messages that are sent from Tobias's UE to the IMS. Therefore, the P-CSCF address needs to be known by the UE before the first SIP message is sent. As this address is not pre-configured in our example, it needs to be discovered by the UE.

In the case of the GPRS the UE can request the addresses of a P-CSCF during the establishment of the general or signalling PDP context. The GGSN will then return the IPv6 prefix of an P-CSCF in response to the activate PDP context request.



**Figure 11.2** Discovering the P-CSCF via DHCP/DNS

Alternatively, the UE can choose to use Dynamic Host Configuration Protocol for IPv6 (DHCPv6) in order to discover the P-CSCF. If the P-CSCF address is returned from DHCP as a Fully Qualified Domain Name (FQDN) rather than an IP address, then the P-CSCF address will be resolved via the Domain Name System (DNS) as the address of any other SIP server.

#### *11.4.2 SIP and DNS Server Configuration via DCHPv6*

In this example we assume that Tobias's phone makes use of the DHCPv6 protocol in order to discover the P-CSCF. There are a lot of configuration options, that a UE can request from a DHCP server, by using the so-called DHCP Options. Also the communication between the UE and the DHCP server, over so-called link-local addresses, is very special, compared to the other IP-based communications we are looking at here. It also needs to be noted, that DHCP and DNS are encoded in a very different way from SIP and therefore the representation chosen in the examples here are provided in a readable form, but do not reflect the exact protocol messages as they are transported (in binary format) between the UE and the servers.

To keep the example simple and focused on the IMS and SIP needs, we will not look at the specific communication requirements of DHCP and will furthermore assume that Tobias's UE requests two types of information from the local DHCP server:

- the address of the DNS server;
- and the address of the P-CSCF.

In order to obtain this information, the UE sends out a DHCP REQUEST message, in which it includes the DHCP Options Request Option (OPTION\_ORO, code: 6), which itself includes a list of DHCP Option Codes, indicating those DHCP Options to UE requests the server to send.

```
DHCP REQUEST
OPTIONS_ORO:    OPTION_DNS_SERVERS (23)
                  OPTION_SIP_SERVER_D (21)
```

The first entry requests the address of a DNS Server, the second entry requests the domain name of a SIP server. There are two DHCP IPv6 OPTIONS for SIP defined, one for domain names (OPTION\_SIP\_SERVER\_D – code 21) and one for IP addresses (OPTION\_SIP\_SERVER\_A – code 22). In this example we assume that the UE only requests the domain name option.

The DHCP Server responds to the Request with with a DHCP Reply, in which it provides the requested information.

```
DHCP REPLY
OPTION_DNS_SERVERS:      5555::cc:dd:cc:dd, 5555::ff:ee:ee:ff
OPTIONS_SIP_SERVER_D:     visited1.fi
```

Tobias's UE now has two DNS servers configured and knows, that the domain 'visited1.fi' allows SIP services, i.e. it provides a P-CSCF. But nevertheless, the UE cannot route

directly to the domain name that was returned in the DHCP REPLY, it first needs to find out the host name of the P-CSCF and its IP address, to be able to send SIP messages there.

#### 11.4.3 DNS Naming Authority Pointer (NAPTR) Resolving

In order to find the required address out, the UE contacts the DNS Server (one of the list that got returned within the DHCP REPLY message) with a DNS query for a Naming Authority Pointer Resource Record (NAPTR RR), asking for further information about the received domain name:

```
DNS query - NAPTR
Name: visited1.fi
```

The DNS server responds with a list of two provided services in this domain:

```
DNS response - visited1.fi
IN NAPTR 100 10 "s" "E2U+sip" "" sip._udp.visisted1.fi
IN NAPTR 100 10 "s" "E2T+sip" "" sip._tcp.visited1.fi
```

The response indicates that the visited domain offers two different services for SIP, one over UDP ('E2U+sip') and one over TCP ('E2T+sip'). In this example we do not explain further the other parameters of the response. A more sophisticated example can be found in Section 13.3.6.2.

#### 11.4.4 Transport Protocol Selection and DNS Service (SRV) Resolving

The IMS puts no further restrictions on the transport protocol for SIP used between the UE and the P-CSCF. In this example it is assumed that the User Datagram Protocol (UDP) is the preferred transport protocol and is therefore chosen by the UE. The UE is free to choose the transport protocol, as the DNS NAPTR response does not include any preference for either UDP ('E2U+sip') or TCP ('E2T+sip'). The network operator can also put preferences for the service entries in the NAPTR response. An example of how this is done is shown in Section 13.3.6.2.

UDP will be used for the transport of SIP messages that are sent between Tobias's UE and the P-CSCF as long as these messages do not exceed 1,300 bytes. When they exceed this limit, the Transmission Control Protocol (TCP) must be used. Due to the fact that SIP also allows a lot of content in the SIP message body (e.g., pictures can be attached to the body of a MESSAGE request), it is likely that both UDP and TCP will be used in parallel while a user is registered.

The UE decides to use UDP for the connection towards the P-CSCF and therefore resolves the indicated service name further via DNS, by sending a DNS SRV query to the local DNS server:

```
DNS query - SRV
Name: _sip._udp.visited1.fi
```

The DNS Server now returns the host name of all SIP servers in the domain visited1.fi that support the desired service, i.e. SIP via UDP.

```
DNS response - _sip._udp.visited1.fi
IN SRV 0 0 5060 pcscf1.visited1.fi
```

The DNS SRV response may include more than one host name, which needs to be selected by specific procedures, which is also not shown here but in the example in Section 13.3.6.2. From this response the UE learns that there is one P-CSCF in the visited network, that supports SIP over UDP connections. The hostname of this P-CSCF is pcscf1.visited1.fi and it can be reached via UDP port 5060.

#### *11.4.5 DNS IPv6 Address Resolving*

Finally, the UE requires the IPv6 address of this P-CSCF, in order to be able to send IP packets (which transport the SIP messages) to the P-CSCF. It therefore once again needs to consult the DNS server in order to resolve the P-CSCFs host name to an IPv6 address:

```
DNS query - AAAA
Name: pcscf1.visited.fi
```

The DNS server responds to the AAAA request with the following information:

```
DNS response - pcscf1.visited.fi
IN AAAA 5555::67:87:59:32
```

With this the UE has collected all the information it needs to route the SIP REGISTER request to the first hop, the P-CSCF in the visited network.

#### *11.4.6 Related Standards*

Standards related to this section are:

- RFC 2782 A DNS RR for specifying the location of services (DNS SRV)
- RFC 2915 The Naming Authority Pointer (NAPTR) DNS Resource Record
- RFC 3263 Session Initiation Protocol (SIP): Locating SIP Servers
- RFC 3315 Dynamic Host Configuration Protocol for IPv6 (DHCPv6)
- RFC 3319 Dynamic Host Configuration Protocol (DHCPv6) Options for Session Initiation Protocol (SIP) Servers
- RFC 3646 DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6)

## **11.5 SIP Registration and Registration Routing Aspects**

### *11.5.1 Overview*

This section concentrates on the SIP routing aspects of Tobias's registration (see Table 11.1 and Figure 11.2).

Tobias's UE will first of all construct a REGISTER request, which it sends to the home domain of Tobias's operator. The relevant information is obtained from the IP Multimedia

**Table 11.1** Routing-related headers

Header	Function	Set up
Via	Routing of requests	By every traversed SIP entity, which puts its address to the Via header during the routing of the request
Route	Routing of requests	Initial requests: by the request-originating UE, which puts the P-CSCF (outbound proxy) address and entries of the Service-Route header Initial requests: by CSCFs, which find the next hop from the public user identity in the request URI (by querying DNS and HSS) or the received Path header Subsequent requests: by the request-originating UE, which put entries to the Route header as collected in the Record-Route header during initial request routing
Record-Route	Records the Route header entries for subsequent requests within a dialog	By CSCFs, which put their addresses into the Record-Route header if they need to receive subsequent requests within a dialog
Service-Route	Indicates the Route header entries for initial requests from the UE to the user's S-CSCF (originating case)	By the S-CSCF, which sends this header back to the UE in the 200 (OK) response for the REGISTER request
Path	Collects the Route header entries for initial requests from the S-CSCF to the user's P-CSCF (terminating case)	By the P-CSCF, which adds itself to the Path header in the REGISTER request and sends it to the S-CSCF

Services Identity Module (ISIM) application on Tobias's Universal Subscriber Identity Module (USIM), as described in Section 11.13.3. The request will traverse the P-CSCF and the I-CSCF, which – if not previously assigned – will select an S-CSCF for Tobias.

The S-CSCF will create, based on the information given in the REGISTER request, the binding between Tobias's public user identity and the IP address of Tobias's UE. This makes it possible for requests from other users to be routed from the S-CSCF to Tobias's UE. The S-CSCF will update the registration information in the HSS, download Tobias's

user profile and will, based on the received initial filter criteria from the HSS, inform any Application Servers (ASs) that are interested in Tobias's registration state.

During the registration procedures the UE will learn the direct route to the S-CSCF from the Service-Route header. After that, the I-CSCF will no longer need to be contacted when Tobias's UE sends out an initial request.

The S-CSCF will become aware of the address of Tobias's P-CSCF from the Path header. This is necessary as all initial requests that are destined for Tobias (e.g., an INVITE request) need to traverse the P-CSCF before they can be sent to the UE.

### 11.5.2 Constructing the REGISTER Request

After establishing the signalling PDP context and discovering the P-CSCF address, Tobias's UE can finally start to construct the initial REGISTER request:

```
REGISTER sip:home1.fr SIP/2.0
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=0uetb
Route: sip:[5555::a:f:f:e];lr
Max-Forwards: 70
From: <sip:tobias@home1.fr>;tag=pohja
To: <sip:tobias@home1.fr>
Contact: "Mobile Phone - Tobias"
<sip:[5555::1:2:3:4]:1357>;expires=600000
Call-ID: apb03a0s09dkjdfg1kj49111
CSeq: 25 REGISTER
Content-Length: 0
```

How the used public and private user identities as well as the registrar address are obtained from the ISIM is described in Section 11.13.3. The above message is not a complete IMS REGISTER request: there are some headers and parameters missing from it. It only includes the information required to explain the procedures in this section, as is the case with all the following messages.

The final destination of the request is the registrar, which is identified in the request URI as *sip:home1.fr* which is the domain name of the home network of Tobias as read from the ISIM.

In the To header we find the public user identity *sip:tobias@home1.fr*, as read from the ISIM, which is going to be registered. SIP registration takes place to tell the registrar that the public user identity *sip:tobias@home1.fr* will be reachable under the IP address that is indicated in the Contact header. This IP address includes the IPv6 prefix, which the UE got assigned during establishment of the dedicated signalling PDP context (see Section 11.3).

Also within the Contact header, the UE indicates that this binding of the IP address to the SIP URI is intended to last 600,000 seconds (nearly a week). In IMS the UE is forced to register for this length of time. Nevertheless, the network can adjust this time:

- During registration procedures by setting the expires value in the Contact header of the 200 (OK) response to the REGISTER request to a smaller value;
- After the user has registered, by making use of registration-state event notifications (e.g., Section 11.13.6 for network-initiated re-authentication).

The IP address of the UE also includes a port number (':1357') which will be further explained in Section 11.7.5.

The Contact header also includes the string 'Mobile Phone – Tobias' as a display name. This name is not mandatory, but will be helpful for advanced service scenarios, such as e.g. when a call gets transferred from one device to another as described in Section 12.10.

The UE puts its IP address into the Via header of the request. This ensures that all responses to this request will be routed back to the UE. A branch parameter that uniquely identifies the transaction is also put in the Via header. Every entity on the route will add its own Via header.

The P-CSCF, which was resolved in Section 11.4, is put in the Route header. The P-CSCF is the next hop to receive the REGISTER message, as it is the topmost – and only – entry of the Route header. The ;lr parameter indicates that the P-CSCF is a loose router, which means that it supports the SIP routing mechanisms as described in RFC 3261. Earlier SIP versions used a so called strict routing mechanism, which is not used within IMS.

The From header identifies the user who is performing the registration. We find in the From header the same public user identity as in the To header, as Tobias is performing a so-called first-party registration (i.e., he is registering himself).

Note that the From header includes a tag, while the To header does not. The recipient of the request (i.e., the registrar) will set the To tag when sending the response to the UE.

A Call-ID header is included, which, together with the value of the CSeq header, identifies the REGISTER transaction.

Finally, there is the indication that the REGISTER request does not carry any content, as the Content-Length header is set to 0.

The example shown on the previous page gives the header names in their long form. In order to avoid unnecessary signalling over the air interface, Tobias's UE may as well use the compact form, which would make the REGISTER request look like:

```
REGISTER sip:home1.fr SIP/2.0
v: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=0uetb
Route: sip:[5555::a:b:c:d];lr
Max-Forwards: 70
f: <sip:tobias@home1.fr>;tag=pohja
t: <sip:tobias@home1.fr>
m: "Mobile Phone - Tobias" <sip:[5555::1:2:3:4]:1357>;
    expires=600000
i: apb03a0s09dkjdfglkj49111
CSeq: 25 REGISTER
l: 0
```

To make reading of SIP messages more convenient, only the long form of the header names will be used in this example.

### 11.5.3 From the UE to the P-CSCF

Now Tobias's UE can send out the REGISTER request to the next hop, which is the topmost entry of the Route header (i.e., the P-CSCF). The request is sent via UDP, as the UE chose the 'E2U+sip' services from the DNS NAPTR response (see Section 11.4.4).

In order to reach the P-CSCF, the UE puts the following information in the lower layer messages:

- the UDP port is set to 5060, as it was indicated for pcscf1.visisted1.fi in the DNS SRV response in Section 11.4.4;
- the IP address is set to 5555::67:87:59:32, as it was resolved from pcscf1.visited1.fi in the DNS AAAA response in Section 11.4.5.

#### 11.5.4 From the P-CSCF to the I-CSCF

When receiving the initial REGISTER request the P-CSCF becomes aware for the first time that Tobias's UE is using it as a SIP outbound proxy. As Tobias is not authenticated at this moment, it can only act as a SIP outbound proxy and, therefore, tries to route the REGISTER request to the next hop.

The P-CSCF removes its own entry from the Route header. After doing so the Route header will be empty. The only routing-related information left now is the registrar address in the request URI, which points to Tobias's home network. In order to discover the address of a SIP proxy in Tobias's home network the P-CSCF needs to resolve the domain name (as given in the request URI) via the DNS. By using DNS NAPTR, SRV and AAAA queries, the P-CSCF will resolve the address of an I-CSCF in Tobias's home network.

Nevertheless, the P-CSCF will not put the address of the I-CSCF in the Route header, as it cannot be sure whether the I-CSCF will act as a loose router or not. Therefore, the P-CSCF will put the address of the I-CSCF as the destination address into the UDP packet that transports SIP requests.

Before sending the REGISTER message the P-CSCF also adds itself to the Via header, in order to receive the response to the request. It also adds a branch parameter to the Via header:

```
REGISTER sip:home1.fr SIP/2.0
Via: SIP/2.0/UDP sip:pcscf1.visited1.fi;branch=0pctb
Via: SIP/2.0/UDP [5555::a:b:c:d];branch=0uetb
Max-Forwards: 69
From: <sip:tobias@home1.fr>;tag=pohja
To: <sip:tobias@home1.fr>
Contact: "Mobile Phone - Tobias"
<sip:[5555::1:2:3:4]:1357>;expires=600000
Call-ID: apb03a0s09dkjdfg1kj49111
CSeq: 25 REGISTER
Content-Length: 0
```

#### 11.5.5 From the I-CSCF to the S-CSCF

The I-CSCF is the entry point to Tobias's home network and will receive every REGISTER request that is originated by Tobias's UE. As the I-CSCF has no knowledge about the assignments of user profiles to specific S-CSCFs, it needs to find out to

which S-CSCF it should forward the REGISTER request. Therefore it needs to query the HSS. In this example we assume that there is only one HSS in the network, so the I-CSCF does not need to query the SLF first. Examples of how the SLF is queried in order to find the relevant HSS can be found in Sections 13.3.6.2 and 13.3.5.2.

In order to obtain the S-CSCF address, the I-CSCF performs the User Registration Status Query procedures over the Cx interface. The detailed examples will show the most important Cx Diameter procedures and in Chapter 13 some Dx and Sh procedures based on Diameter will be shown as well. Nevertheless the Diameter protocol is not introduced in great detail here. The examples given in this book indicate the (3GPP specific) Diameter request and answer messages as well as the content of the relevant Attribute Value Pairs (AVPs). Interested readers will find good guidance in the referenced standards, where [RFC 3589] in particular gives a good introduction to the Diameter protocol and its procedures.

The I-CSCF sends out a Diameter User Authorization Request (UAR) to the HSS, including the following information:

- the ‘R’ command flag set to ‘1’, indicating that this is an Diameter request;
- the Command-Code set to ‘300’, indicating the Diameter ‘User Authorization’ command;
- the ‘Visited-Network-Identifier’ AVP (600), indicating the value contained in the P-Visited-Network header within the received SIP REGISTER request (see Section 11.11.2), in this case it would be the value ‘Kaunis Musta Kissä’;
- optionally the ‘User-Authorization-Type’ AVP (623). This AVP indicates the purpose of the received REGISTER request. If the expiration time, as indicated in the Contact header, is zero then a de-registration needs to be indicated here. As we are in the process of an initial registration, the value is set to 600 000 seconds and therefore the AVP is set to ‘REGISTRATION’ (0) – as this is the default value, the AVP could also be avoided;
- the ‘User-Name’ AVP (1) set to the private identity of Tobias, indicated in the username field within the Authorization header of the SIP REGISTER request (see Section 11.6.3), i.e. to ‘tobias\_private@home1.fr’;
- the ‘Origin-Host’ AVP (264) set the address of the querying I-CSCF, i.e. ‘icscf1.home1.fr’;
- the ‘Origin-Realm’ AVP (296) set to the domain name of the operator network in which the I-CSCF is located, i.e. ‘home1.fr’;
- the ‘Destination-Realm’ AVP (283) set to the home domain of the HSS, i.e. ‘home1.fr’, as this is the domain within which the user location information is queried;
- the ‘Destination-Host’ AVP (293) set to the address of the HSS, which needs to be locally configured within the S-CSCF, as no SLF query is performed.

Upon receiving this UAR, the HSS first checks whether the I-CSCF (as indicated in the ‘Origin-Host’ AVP) is allowed to query authorization status of Tobias, which it is of course allowed in this example. The HSS then detects that currently there is no S-CSCF assigned for Tobias and therefore leaves the decision of which S-CSCF to select for Tobias to the I-CSCF. It returns a Diameter Location Info Answer (LIA) with the following content:

- the ‘R’ command flag set to ‘0’, indicating that this is a Diameter answer;
- the Command-Code set to ‘300’, indicating the Diameter ‘User Authorization’ command;
- the ‘Result-Code’ AVP (268) set to ‘DIAMETER\_SUCCESS’ (2001), indicating that the query was successful;
- the ‘Server-Capabilities’ AVP (603), which includes a set of Mandatory-Capability AVPs (604) as well as Optional-Capability AVPs (605) – these capabilities are expressed as integer values and their meaning is specific to the operator network. The operator needs to make sure that all the listed mandatory and optional capabilities are understood at the I-CSCF, as the I-CSCF will select a S-CSCF for the ongoing registration based on this list. Based on the received list of mandatory and optional capabilities, the I-CSCF selects an S-CSCF for Tobias, in this case the S-CSCF with the address `sip:scscf1.home1.fr;lr`.

After putting its own entry in the topmost Via header, the I-CSCF sends the REGISTER request to the S-CSCF address that it either got from the HSS or that it selected:

```
REGISTER sip:home1.fr SIP/2.0
Via: SIP/2.0/UDP sip:icscf1.home1.fr;branch=0ictb
Via: SIP/2.0/UDP sip:pcscf1.visited1.fi;branch=0pctb
Via: SIP/2.0/UDP [5555::a:b:c:d];branch=0uetb
Route: sip:scscf1.home1.fr;lr
Max-Forwards: 68
From: <sip:tobias@home1.fr>;tag=pohja
To: <sip:tobias@home1.fr>
Contact: "Mobile Phone - Tobias"
<sip:[5555::1:2:3:4]:1357>;expires=600000
Call-ID: apb03a0s09dkjdfg1kj49111
CSeq: 25 REGISTER
Content-Length: 0
```

### 11.5.6 Registration at the S-CSCF

After receiving the initial REGISTER request, the S-CSCF will request Tobias to authenticate himself, as described in Section 10.6. This will result in another REGISTER request from Tobias. This second REGISTER request will include the same registration-related information and will also be routed in exactly the same way as the initial REGISTER request. Consequently, new CSeq numbers, branch parameters and a new From tag will be included in it. The second REGISTER received by the S-CSCF will look like:

```
REGISTER sip:home1.fr SIP/2.0
Via: SIP/2.0/UDP sip:icscf1.home1.fr;branch=3ictb
Via: SIP/2.0/UDP sip:pcscf1.visited1.fi;branch=2pctb
Via: SIP/2.0/UDP [5555::a:b:c:d];branch =1uetb
Route: sip:scscf1.home1.fr;lr
Max-Forwards: 68
From: <sip:tobias@home1.fr>;tag=ulkomaa
```

```
To: <sip:tobias@home1.fr>
Contact: "Mobile Phone - Tobias"
<sip:[5555::1:2:3:4]:1357>;expires=600000
Call-ID: apb03a0s09dkjdfglkj49111
CSeq: 47 REGISTER
Content-Length: 0
```

Assuming that the authentication procedures are successful, the S-CSCF will then register Tobias. This means the S-CSCF will create a binding for the public user identity that was indicated in the To header of the REGISTER request (`sip:tobias@home1.fr`) and the contact address (`sip:[5555::a:b:c:d]`). This binding will exist for exactly 600 000 seconds, which is the value that the UE entered into the ‘expires’ parameter of the Contact header, unless the S-CSCF decides to reduce this time due to local policy.

The S-CSCF also updates the information in the HSS to indicate that Tobias has now been registered and to download Tobias’s user profile (see Section 3.12). Therefore the S-CSCF performs the Diameter S-CSCF Registration Notification procedures over the Cx interface by sending a Diameter Server-Assigment Request (SAR) to the HSS, including:

- the ‘R’ command flag set to ‘1’, indicating that this is a Diameter request;
- the Command-Code set to ‘301’, indicating the Diameter ‘Server Assignment’ command;
- the ‘Public-Identity’ AVP (600), indicating the public user identity that is being registered, i.e. the URI contained in the To header of the REGISTER request: ‘`sip:tobias@home1.fr`’;
- the ‘Server-Name’ AVP (602) set to the SIP URI of the S-CSCF performing the SAR, i.e. ‘`sip:scscf1.home1.fr`’;
- the ‘User-Name’ AVP (1) set to the private identity of Tobias, indicated in the username field within the Authorization header of the SIP REGISTER request (see Section 11.6.3), i.e. to ‘`tobias.private@home1.fr`’;
- the ‘Server-Assigment-Type’ AVP (614) set to the value ‘REGISTRATION’ (1) as this is an initial registration. Other values of the Server-Assigment-Type AVP can for example be ‘RE\_REGISTRATION’ (2) when Tobias’s phone sends out another REGISTER request to keep the active registration alive (see Section 11.14) or ‘USER\_DEREGISTRATION’ (5) when Tobias sends another REGISTER request with an expiration time set to ‘0’ (see Section 11.14.1);
- the ‘UserData-Already-Available’ AVP (624) set to ‘USER\_DATA\_NOT\_AVAILABLE’ (0), indicating that the S-CSCF does not have locally available the service profile of Tobias;
- the ‘Origin-Host’ AVP (264) set the address of the S-CSCF, i.e. ‘`scscf1.home1.fr`’;
- the ‘Origin-Realm’ AVP (296) set to the domain name of the operator network in which the I-CSCF is located, i.e. ‘`home1.fr`’;
- the ‘Destination-Realm’ AVP (283) set to the home domain of the HSS, i.e. ‘`home1.fr`’, as this is the domain within which the user location information is queried;
- the ‘Destination-Host’ AVP (293) set to the address of the HSS, which needs to be locally configured within the S-CSCF, as no SLF query is performed.

The HSS now assigns the S-CSCF name to the registration set of Tobias's public user identities. This means, that as long as Tobias stays registered with this S-CSCF, every location query that is sent to the HSS that is asking how to route requests destined for Tobias further (see e.g. Section 12.3.3.4) will be responded to by the HSS with the address of the S-CSCF.

The HSS also sets Tobias's user profile to 'registered', which might lead to Sh-Notification messages towards attached Application Servers (see Section 2.3.7).

The HSS sends back a Diameter Server Assignment Answer (SAA) to the S-CSCF, containing:

- the 'R' command flag set to '0', indicating that this is a Diameter answer;
- the Command-Code set to '301', indicating the Diameter 'Server Assignment' command;
- the 'User-Name' AVP (1) set to the private identity of Tobias, i.e. to 'tobias\_private@home1.fr';
- the 'Result-Code' AVP (268) set to 'DIAMETER\_SUCCESS' (2001), indicating that the request was successful;
- the 'User-Data' AVP (606), which includes the user profile of Tobias (see Section 3.12)
- the 'Charging-Information' AVP (618), containing the addresses of the charging functions (see Section 11.12) within the following AVPs:
  - the 'Primary-Event-Charging-Function-Name' AVP (619) including the address of the primary online charging function (OCF – see Section 3.11) '5555::f66:e77:d88:c77', which will be distributed within the P-Charging-Function-Address header (see Section 3.11.4);
  - the 'Secondary-Event-Charging-Function-Name' AVP (620) including the address of the secondary OCF, which is stored at the S-CSCF;
  - the 'Primary-Charging-Collection-Function-Name' AVP (621) including the address of the primary charging data function (CDF – see Section 3.11 '5555::a55:b44:c33:d22', which will be distributed within the P-Charging-Function-Address header (see Section 12.7.4);
  - the 'Secondary-Charging-Collection-Function-Name' AVP (622) including the address of the secondary CDF, which is stored at the S-CSCF.
- optionally the 'Associated-Identities' AVP (632), containing the related private identities (note: not the public identities) that belong to the same user, in this case Tobias. Tobias can have several phones, each one equipped with a USIM/ISIM and he might have in addition a HTTP Digest username (private identity) and password – in these cases, the list of private identities are sent down to the S-CSCF.

### 11.5.7 The 200 (OK) Response

After receiving the SAM, the S-CSCF will send back a 200 (OK) response to the UE, to indicate that the registration procedure has succeeded:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP icscf1.home1.fr;branch=3ictb
```

```

Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=2pctb
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=1uetb
From: <sip:tobias@home1.fr>;tag=ulkomaa
To: <sip:tobias@home1.fr>;tag=kotimaa
Contact: "Mobile Phone - Tobias"
<sip:[5555::a:b:c:d]:1357>;expires=600000
Call-ID: apb03a0s09dkjdfglkj49222
CSeq: 47 REGISTER
Content-Length: 0

```

The S-CSCF has added a tag to the To header.

The response is routed back to the UE over all the CSCFs that received the REGISTER request; it manages to do this because CSCFs put their own address in the topmost Via header list when they receive REGISTER requests. Now, when receiving the 200 (OK) response, they just remove their own entry from the Via list and send the request forward to the address indicated in the topmost Via header.

The UE, when receiving this response, will know that the registration was successful.

#### 11.5.8 The Service-Route Header

We have seen that neither the UE nor the P-CSCF were aware of the address of the S-CSCF during the registration procedures; consequently, the I-CSCF had to be contacted to discover the S-CSCF address from the HSS.

In order to avoid the I-CSCF as an extra hop for every initial message sent from the UE, the S-CSCF will return its address in the Service-Route header in the 200 (OK) response to the REGISTER request:

```

SIP/2.0 200 OK
Service-Route: sip:orig@scscf1.home1.fr;lr

```

The UE, when receiving the 200 (OK) response, will store the entries in the Service-Route header. Whenever the UE sends out any initial request other than a REGISTER message, it will:

- include the addresses that were received in the Service-Route header within a Route header of the initial request; and
- include the P-CSCF address as the topmost Route entry in the initial request.

Examples of how initial requests are routed are given in Section 11.13.6 for a SUBSCRIBE request and in Section 0 for an INVITE request.

The S-CSCF in this example puts a user part ('orig') in its Service-Route entry as it needs to distinguish between two types of requests:

- requests originated from the served user (i.e., Tobias); and
- requests destined for Tobias's UE.

Whenever the S-CSCF receives an initial request (e.g., an INVITE request) it needs to determine whether this request is originated from or destined to the served user. The

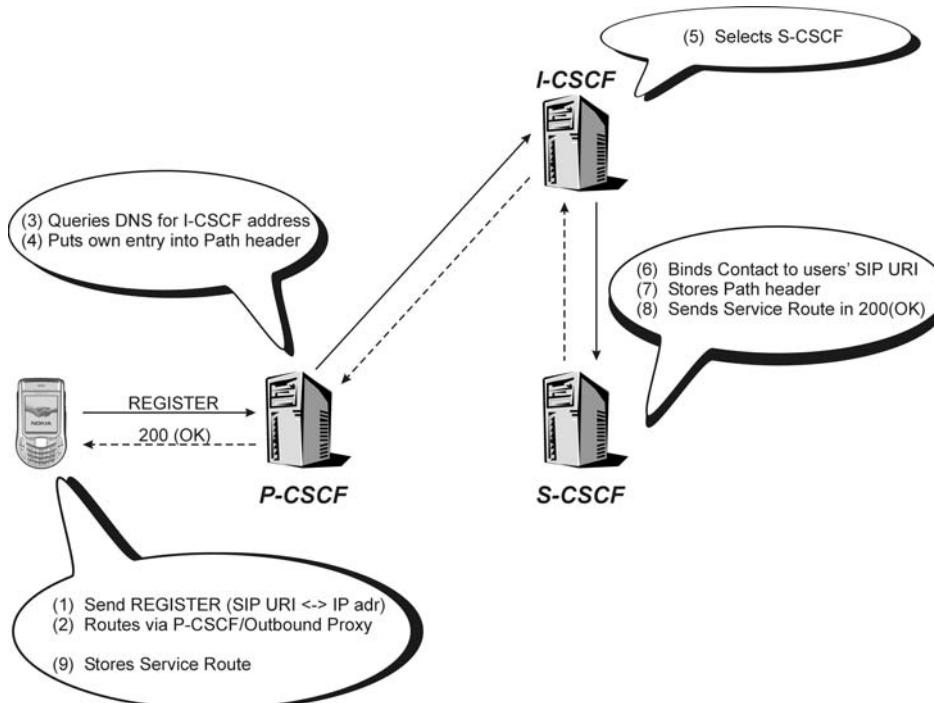
user part entry in the Route header makes it easy for the S-CSCF to discover whether a received request was originated from the served user, as Tobias's UE will include the S-CSCF's Service-Route entry as a Route entry within all requests that it originates.

### 11.5.9 The Path Header

The S-CSCF will receive all initial requests that are destined to Tobias, as it acts as his registrar. Normal SIP procedures allow the registrar to send requests directly to the UE. In the case of IMS this is not possible, because the P-CSCF needs to be contacted first; this is because the P-CSCF has established IPsec SAs with the UE that guarantee that all messages will be sent and received integrity-protected (see Section 11.7). Furthermore, the P-CSCF has an important role in media authorization (see Section 11.7.2) as it is the only network element in the IMS that has a direct connection to the GGSN.

Therefore, the S-CSCF needs to ensure that every request that is sent to the UE first traverses the P-CSCF. To make this possible, the P-CSCF includes its own address in every REGISTER request within a Path header:

```
REGISTER sip:home1.fr SIP/2.0
Path: sip:pcscf1.visited1.fi;lr
```



**Figure 11.3** Routing during registration

After successful registration of the user, the S-CSCF saves this P-CSCF address. Whenever a request for Tobias is received, the S-CSCF will include a Route header with the address that was received in the Path header. An example of routing an initial INVITE request toward the served user is given in Section 12.3.3.5.

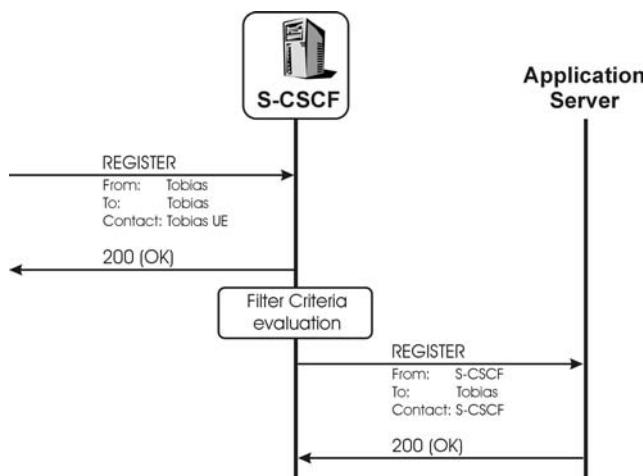
#### 11.5.10 Third-Party Registration to Application Servers

After successful registration the S-CSCF will check the downloaded filter criteria of the user (see Section 3.13). We assume that there is a presence server that provides its services to Tobias; this presence server needs to know that Tobias has now been registered and is, therefore, available. To inform the presence server about this, filter criteria have been set which trigger all the REGISTER requests that originate from Tobias's public user identity (Table 11.2).

Due to these filter criteria, the S-CSCF will generate a third-party REGISTER request (Figure 11.3) and send it to the presence server whenever Tobias performs a successful registration:

**Table 11.2** Filter criteria in Tobias's S-CSCF

Element of filter criteria	Filter criteria
SPT: session case	Originating
SPT: public user identity	<i>sip:tobias@home1.fr</i>
SPT: SIP method	REGISTER
Application server	<i>sip:presence.home1.fr;lr</i>



**Figure 11.4** Third party register by S-CSCF

This REGISTER request is destined to the presence server at presence.home1.fr, as indicated in the request URI. As no Route header is included, the request will be sent directly to that address.

The To header includes the public user identity of Tobias, as this is the URI that was registered.

The S-CSCF indicates its own address in the From header, as it is registering Tobias's public user identity on behalf of Tobias (i.e., as a third party). Furthermore, the S-CSCF indicates its own address within the Contact header. This ensures that the presence server never routes directly to Tobias's UE, but will always contact the S-CSCF first.

```
REGISTER sip:presence.home1.fr SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.fr;branch=99sctb
Max-Forwards: 70
From: <sip:scscf1.home1.fr>;tag=6fa
To: <sip:tobias@home1.fr>
Contact: <sip:scscf1.home1.fr>;expires=600000
Call-ID: las22kdoa45siewrf
CSeq: 87 REGISTER
Content-Length: 0
```

The presence server will send back a 200 (OK) response for this REGISTER request to the S-CSCF, but will not start acting as a registrar for Tobias. It will take the REGISTER request as an indication that Tobias has been successfully registered at the S-CSCF that is Tobias's registrar. If the presence server needs more information about Tobias's registration state (e.g., all other public user identities that have been implicitly registered for Tobias), it can subscribe to the registration-state information of Tobias in the same way as the UE and the P-CSCF do (see Section 11.13.6).

#### *11.5.11 Updating the User Profile*

The user profile, that the S-CSCF downloaded from the HSS (see Section 11.5.6), can be changed by the operator of Tobias's network at any time, also whilst Tobias is registered. If such a change in the user profile happens, the HSS informs the S-CSCF by performing the Diameter User Profile Update procedures over the Cx interface. The HSS therefore sends a Diameter Push-Profile Request (PPR) to the S-CSCF, containing:

- the 'R' command flag set to '1', indicating that this is a Diameter request;
- the Command-Code set to '305', indicating the Diameter 'Server Assignment' command;
- the 'User-Name', 'User-Data' and 'Charging-Information' AVPs in the same way as in the Diameter Server Assignment Answer (SAA) as described in Section 11.5.6;
- the 'Origin-Host' and Origin-Realm AVPs set the address and realm (home1.fr) of the HSS;
- the 'Destination-host' and 'Destination-Realm' AVPs the address (scscf1.home1.fr) and domain (home1.fr) of the S-CSCF.

The S-CSCF updates the user profile of Tobias accordingly and sends back a Diameter Push-Profile Answer (PPM), including only the ‘Result-Code’ AVP (268) set to ‘DIAMETER\_SUCCESS’ (2001), indicating that the request was successful. The new user profile will be immediately taken into use at the S-CSCF for all new standalone transactions and dialogs originating from and destined to Tobias. For all existing transactions and dialogs, the user profile cannot be changed.

#### 11.5.12 Related Standards

Specifications relevant to Section 11.5 are:

- RFC3327 Session Initiation Protocol (SIP) Extension Header Field for Registering NonAdjacent Contacts.  
RFC3608 Session Initiation Protocol (SIP) Extension Header Field for Service Route Discovery during Registration.

Communication between the CSCFs and the I-CSCF over diameter is standardized in the following specifications:

- 3GPP TS 29.228 IMS Cx and Dx interfaces – signalling flows and messages.  
3GPP TS 29.229 IMS Cx and Dx interfaces based on the Diameter protocol.  
RFC3588 Diameter Base Protocol.  
RFC3589 Diameter Command Codes for Third Generation Partnership Project (3GPP) Release 5.

## 11.6 Authentication

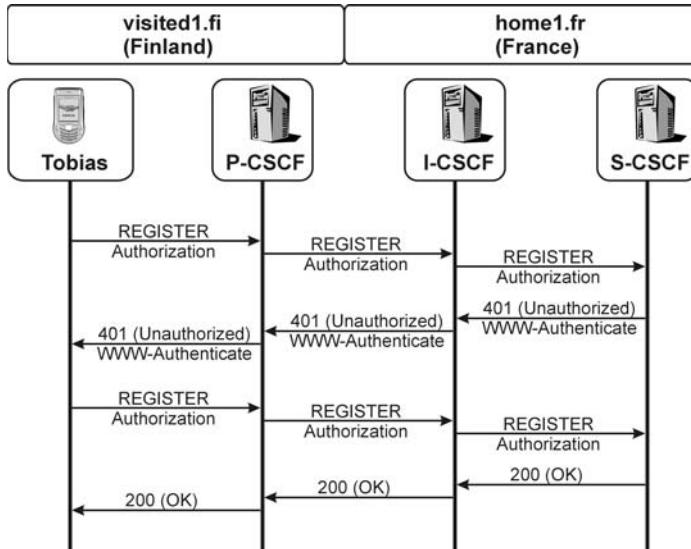
### 11.6.1 Overview

As shown in Section 3.21, the IMS is based on several security relations. Two of them – **authentication** between user and network, and the **SAs** between the UE and the P-CSCF – have an influence on SIP signalling (Figure 11.4). Authentication and SA establishment procedures in the IMS are directly coupled to SIP registration procedures.

In this example it is assumed that 3GPP AKA is used for the authentication of the user. In Section 11.16 an example for GPRS-IMS-bundled authentication (GIBA) will be given.

IMS authentication is based on a shared secret and a sequence number (SQN), which is only available in the HSS and the ISIM application on the Universal Integrated Circuit Card (UICC) card in Tobias’s UE. As the HSS never directly communicates with the UE, the S-CSCF performs the authentication procedures and all security-related parameters that are needed by the S-CSCF. The so-called Authentication Vector (AV) is downloaded by the S-CSCF from the HSS during registration.

In order to authenticate, Tobias sends his private user identity (in our example this is *tobias\_private@home1.fr*) in the initial REGISTER request. This private user identity is



**Figure 11.5** Authentication information flows during IMS registration

stored within the ISIM application and is only used for authentication and registration procedures.

When receiving this REGISTER request, the S-CSCF downloads the AV from the HSS. The AV does not include the shared secret and the SQN itself, but does include (among other parameters):

- a random challenge (RAND);
- the expected result (XRES);
- the network authentication token (AUTN);
- the Integrity Key (IK); and
- the Ciphering Key (CK).

In order to authenticate, the S-CSCF rejects the initial REGISTER request from the user with a 401 (Unauthorized) response, which includes (among other parameters) the RAND, the AUTN, the IK and the CK.

The P-CSCF, when receiving the 401 (Unauthorized) response, removes the IK and the CK from the response before sending it to the UE. The IK is the base for the SAs that get established between the P-CSCF and the UE immediately afterwards (see Section 11.7).

After receiving the response, the UE hands the received parameters over to the ISIM application, which:

- verifies the AUTN based on the shared secret and the SQN – when AUTN verification is successful the network is authenticated (i.e., the UE can be sure that the authentication data were received from the home operator's network);
- calculates the result (RES) based on the shared secret and the received RAND;

- calculates the IK, which is then shared between the P-CSCF and the UE and will serve as the base for the SAs.

Afterwards, the UE sends the authentication challenge response (RES) in the second REGISTER request back to the S-CSCF, which compares it with the XRES that was received in the AV from the HSS. If the verification is successful, the S-CSCF will treat the user as authenticated and will perform the SIP registration procedures (see Section 11.5.6).

Whenever the UE sends out another REGISTER request (i.e., due to either re- or de-registration), it will always include the same authentication parameters as included in the second REGISTER request, until the S-CSCF re-authenticates the UE.

### 11.6.2 HTTP Digest and 3GPP AKA

The Hypertext Transfer Protocol (HTTP) digest is specified in [RFC2617], and how it is used with SIP is described in [RFC3261]. The IMS on the contrary is part of the Third Generation Partnership Project/Universal Mobile Telecommunications System (3GPP/UMTS) architecture, which uses the 3GPP Authentication and Key Agreement (AKA) mechanism for authentication.

In order to achieve 3GPP AKA-based authentication within the IMS, [RFC3310] defines how 3GPP AKA parameters (as described above) can be mapped to HTTP digest authentication. Therefore, the signalling elements (SIP headers and parameters) used to transport 3GPP AKA information are identical to those used for the HTTP digest. Nevertheless, their meanings (i.e., their interpretation at the UE, the P-CSCF and the S-CSCF) are different.

In order to distinguish the 3GPP AKA authentication mechanism from other HTTP digest mechanisms (e.g., MD5), it was given a new algorithm value: “AKAv1-MD5”.

### 11.6.3 Authentication Information in the Initial REGISTER Request

Within the initial REGISTER request Tobias’s UE utilizes the HTTP Digest Authorization header to transport Tobias’s private user identity. In order to fulfil HTTP digest requirements, the UE includes the following fields in the Authorization header:

- The authentication scheme – set to the value ‘Digest’, as the 3GPP AKA is mapped to the HTTP digest mechanism;
- The username field – set to Tobias’s private user identity, which will be used by the S-CSCF and the HSS to identify the user and to find the corresponding AV;
- The realm and URI fields – set to the home domain of Tobias;
- The response and nonce fields – which are left empty. These fields are mandated by the HTTP digest, but not used in the initial REGISTER request.

The REGISTER now looks like:

```
REGISTER sip:home1.fr SIP/2.0
Authorization: Digest username="tobias_private@home1.fr",
realm="home1.fr",
nonce=" ",
```

```
uri="sip:home1.fr",
response=""
```

As the UE and the P-CSCF did not establish any kind of mutual security mechanism at the SIP signalling level, the P-CSCF cannot guarantee that the REGISTER request really does originate from Tobias: for example, a malicious user could have constructed the request and sent it to the P-CSCF, without the P-CSCF knowing. Therefore, the P-CSCF adds the integrity-protected field with the value ‘no’ to the Authorization header, before sending the request toward Tobias’s home network:

```
REGISTER sip:home1.fr SIP/2.0
Authorization: Digest username="tobias_private@home1.fr",
  realm="home1.fr",
  nonce="",
  uri="sip:home1.fr",
  response="",
  integrity-protected="no"
```

#### 11.6.4 S-CSCF Downloads the Authentication Vector (AV) from the HSS

When receiving this REGISTER request, the S-CSCF downloads the AV from the HSS by performing the Diameter Authentication procedures over the Cx interface by sending a Diameter Multimedia-Auth Request (MAR) to the HSS, containing:

- the ‘R’ command flag set to ‘1’, indicating that this is a Diameter request;
- the Command-Code set to ‘303’, indicating the Diameter ‘Multimedia-Auth’ command;
- the ‘Public-Identity’ AVP (600), indicating the public user identity that is being registered, i.e. the URI contained in the To header of the REGISTER request: ‘sip:tobias@home1.fr’;
- the ‘Server-Name’ AVP (602) set to the SIP URI of the S-CSCF performing the MAR, i.e. ‘sip:scscf1.home1.fr’;
- the ‘User-Name’ AVP (1) set to the private identity of Tobias, indicated in the user-name field within the Authorization header of the SIP REGISTER request, i.e. to ‘tobias.private@home1.fr’;
- the ‘SIP-Number-Auth-Items’ AVP (607) set to ‘5’, indicating that the S-CSCF wants to download five consecutive AVs for this user;
- the ‘SIP-Auth-Data-Item’ AVP (612), including the ‘SIP-Authentication-Scheme’ AVP (608) set to ‘Unknown’, as the S-CSCF does not know (from the received REGISTER message and the included Authorization header as set by the user – see Section 11.6.3) what authentication scheme will be used by the user;
- the ‘Origin-Host’ AVP (264) set the address of the S-CSCF, i.e. ‘scscf1.home1.fr’;
- the ‘Origin-Realm’ AVP (296) set to the domain name of the operator network in which the S-CSCF is located, i.e. ‘home1.fr’;
- the ‘Destination-Realm’ AVP (283) set to the home domain of the HSS, i.e. ‘home1.fr’, as this is the domain within which the user location information is queried;
- the ‘Destination-Host’ AVP (293) set to the address of the HSS, which needs to be locally configured within the S-CSCF, as no SLF query is performed. Examples of

how the SLF is queried in order to find the relevant HSS can be found in Sections 13.3.6.2 and 13.3.5.2.

After receiving this MAR, the HSS first checks whether the S-CSCF is allowed to download authentication data related to Tobias. As the S-CSCF is allowed, the HSS returns a Diameter Multimedia-Auth Answer (MAA) to the S-CSCF, containing:

- the ‘R’ command flag set to ‘0’, indicating that this is a Diameter answer;
- the Command-Code set to ‘303’, indicating the Diameter ‘Multimedia-Auth’ command;
- the ‘Public-Identity’ AVP (608) set to the same value as received in the MAR, i.e. to Tobias’s public user identity;
- the ‘User-Name’ AVP (1) set to the private identity of Tobias;
- the ‘SIP-Number-Auth-Items’ AVP (607) set to ‘5’, indicating that HSS sends down five consecutive AVs to the S-CSCF in order to challenge this user;
- the ‘SIP-Auth-Data-Item’ AVP (612), including five times the following AVPs (one time per Authentication Vector):
  - the ‘SIP-Item-Number’ AVP (613) set to ‘1’ for the first AV (and consequently incremented by 1 for the other four AVs), indicating the sequence by which the AVs must be used in order to challenge the user;
  - the ‘SIP-Authentication-Scheme’ AVP (608) set to ‘Digest-AKAv1-MD5’ as this is the authentication scheme used towards the received private identity of Tobias. Note that the authentication scheme is depending on the private identity, as this shows whether Tobias is registering (as in this example) from a phone that is equipped with an USIM/ISIM or whether he is using standalone HTTP digest procedures for authentication;
  - the ‘SIP-Authenticate’ AVP (609) includes:
    - the random challenge (RAND);
    - the network authentication token (AUTN);
  - the ‘SIP-Authorization’ AVP (610) includes the expected result (XRES);
  - the ‘Confidentiality-Key’ AVP (625) includes the confidentiality key (CK);
  - the ‘Integrity-Key’ AVP (626) includes the integrity key (IK);
- the ‘Result-Code’ AVP (268) set to ‘DIAMETER\_SUCCESS’ (2001), indicating that the query was successful.

### 11.6.5 S-CSCF Challenges the UE

Based on the data in the AV, it returns the WWW-Authenticate header in the 401 (Unauthorized) response and populates its fields as follows:

- in the nonce field it has the RAND and AUTN parameters, which were present in the SIP-Authenticate AVP in the MAA. Both values are 32 bytes long and Base 64-encoded (the nonce field may include additional server-specific data);
- in the algorithm field it has the value ‘AKAv1-MD5’, as it was present in the SIP-Authentication-Scheme AVP in the MAA and which identifies the 3GPP AKA mechanism; and

- in the ik and ck extension fields it has the integrity and ciphering keys, as they were present in the Confidentiality-Key and Integrity-Key AVPs in the MAA. Note that these two fields are not part of the original definition of the WWW-Authenticate header, which is defined in [RFC3261]. These fields are defined in [3GPP TS 24.229].

The WWW-Authenticate fields look like:

```
SIP/2.0 401 Unauthorized
WWW-Authenticate: Digest realm="home1.fr",
    nonce=A34Cm+Fva37UYWpGNB34JP, algorithm=AKAv1-MD5,
    ik="0123456789abcdedcba9876543210",
    ck="9876543210abcdedcba0123456789"
```

After receiving the 401 (Unauthorized) response, the P-CSCF must remove and store the ik and ck fields from the WWW-Authenticate header, before sending the response toward the UE:

```
SIP/2.0 401 Unauthorized
WWW-Authenticate: Digest realm="home1.fr",
    nonce=A34Cm+Fva37UYWpGNB34JP, algorithm=AKAv1-MD5
```

#### 11.6.6 UE's Response to the Challenge

From the received AUTN parameter the ISIM application in Tobias's UE now discovers that it was really Tobias's home operator network that sent the 401 (Unauthorized) response. It can also derive from the AUTN that the SQN (sequence number) is still in sync between the HSS and the ISIM.

The received parameters as well as the shared secret allow the ISIM to generate the values for the response and hand them over to the UE. The UE adds the Authorization header to the second REGISTER request, including (among others) the following fields:

- The username field – which includes Tobias's private user identity;
- The nonce field – which is returned with the same value as it was received in the WWW-Authenticate header of the 401 (Unauthorized) response;
- The response field – which includes the authentication challenge RES that was derived by the ISIM from the received RAND and the shared secret.

The ISIM will also calculate the IK, which is also known by the P-CSCF. Based on this key (and other information – see Section 11.7) the UE and the P-CSCF establish IPsec SAs, over which the UE sends the second REGISTER request:

```
REGISTER sip:home1.fr SIP/2.0
Authorization: Digest username="user1.private@home1.fr",
    realm="home1.fr",
    nonce=A34Cm+Fva37UYWpGNB34JP, algorithm=AKAv1-MD5,
```

```
uri="sip:home1.fr",
response="6629fae49393a05397450978507c4ef1"
```

### 11.6.7 Integrity Protection and Successful Authentication

The P-CSCF is now in a position to discover whether the received REGISTER request was modified on its way from the UE to the P-CSCF, as it can now check its integrity. If this check is successful, the P-CSCF adds the ‘integrity-protected’ field with the value ‘yes’ to the Authorization header and sends the REGISTER request toward Tobias’s home network:

```
REGISTER sip:home1.fr SIP/2.0
Authorization: Digest username="user1_private@home1.fr",
realm="home1.fr",
nonce=A34Cm+Fva37UYWpGNB34JP, algorithm=AKAv1-MD5,
uri="sip:home1.fr",
response="6629fae49393a05397450978507c4ef1",
integrity-protected="yes"
```

The S-CSCF now compares the received RES from the UE and the XRES that was included in the SIP-Authorization AVP of the AV that was received in the MAA from the HSS. If these two parameters are identical, then the S-CSCF has successfully authenticated the user. Only after that, will it proceed with normal SIP registration procedures.

### 11.6.8 Related Standards

Specifications relevant to Section 11.6 are:

- |                |  |
|----------------|--|
| 3GPP TS 33.102 | Security architecture.   |
| 3GPP TS 33.203 | Access security for IP-based services.   |
| RFC2401        | Security Architecture for the Internet Protocol.   |
| RFC2403        | The Use of HMAC-MD5-96 within ESP and AH.  |
| RFC2404        | The Use of HMAC-SHA-1-96 within ESP and AH.  |
| RFC2617        | HTTP Authentication: Basic and Digest Access Authentication.   |
| RFC3310        | Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA). |

## 11.7 Access Security – IPsec SAs

### 11.7.1 Overview

Section 3.21.4 describes how access security works in principle. Security via the Gm interface is achieved by means of IPsec SAs, which require specific handling at the SIP signalling level. This section describes how the UE and P-CSCF negotiate the security mechanism, how IPsec-related parameters are exchanged and how SAs are established and handled.

As the establishment of IPsec SAs is based on authentication of the user, new SAs are established during every re-authentication process. Consequently, new pairs of IPsec SAs have to be established between the UE and the P-CSCF.

### 11.7.2 Establishing an SA During Initial Registration

The initial REGISTER request as well as the 401 (Unauthorized) response are sent between the UE and the P-CSCF without any kind of protection. These two messages transport information that allows the UE and the P-CSCF to negotiate the security mechanism and to agree on the parameters and ports that will be used for the SAs.

During the registration process two pairs of IPsec SAs are established between the UE and the P-CSCF. Unless otherwise stated, such a set of two pairs of SAs is referred to as a ‘set of SAs’, while a single or specific IPsec SA from these four is referred to as an ‘SA’.

The four IPsec SAs are not static connections (e.g., TCP connections). They can be regarded as logical associations between the UE and the P-CSCF that allow the secure exchange of SIP messages.

A set of SAs facilitates four ports:

- the protected client port at the UE (uc1);
- the protected server port at the UE (us1);
- the protected client port at the P-CSCF (pc1); and
- the protected server port at the P-CSCF (ps1).

These ports are negotiated between the UE and the P-CSCF during initial registration (Figure 11.5) by using the Security-Client, Security-Server and Security-Verify headers of the SIP Security Mechanism Agreement (see Section 11.8).

The set of SAs needs to be established with a shared key. Unfortunately, the P-CSCF knows nothing about the security parameters that are shared between Tobias’s ISIM application and the HSS in the home network. Therefore, the S-CSCF sends the IK and the CK to the P-CSCF within the WWW-Authenticate header in the 401 (Unauthorized) response. The P-CSCF must remove these two keys from the header and store them locally before sending the 401 (Unauthorized) response toward the UE. The IK is then used by the P-CSCF as the shared key for the set of SAs. The UE at the other end of the Gm interface calculates the IK from the received challenge in the 401 (Unauthorized) response and also uses it as the shared key (see Section 11.6.7).

By means of the IK, the P-CSCF and the UE can then establish the set of SAs between the four ports that were exchanged beforehand in the initial REGISTER request and its response:

- between uc1 and ps1 for sending SIP requests from the UE to the P-CSCF;
- between us1 and pc1 for sending SIP responses from the P-CSCF to the UE;
- between us1 and pc1 for sending SIP requests from the P-CSCF to the UE; and
- between uc1 and ps1 for sending SIP responses from the UE to the P-CSCF.

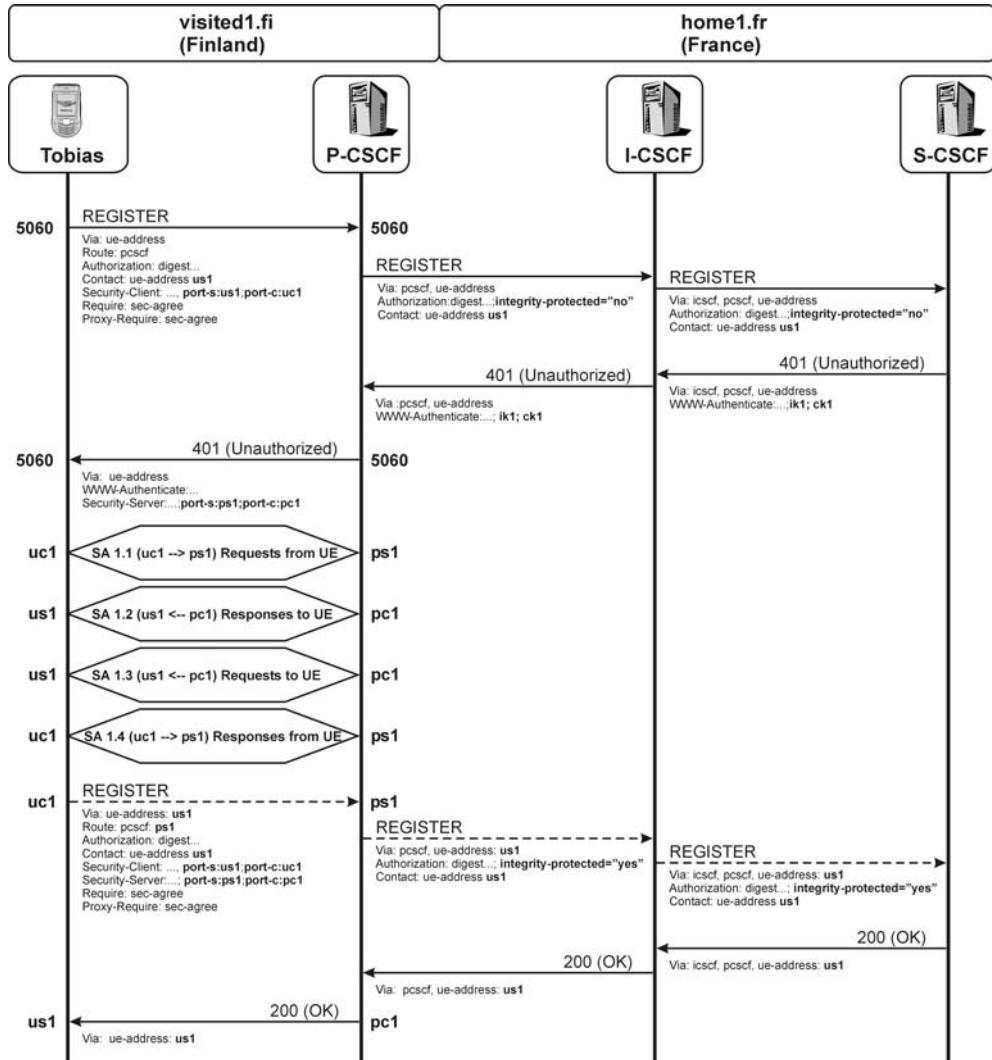


Figure 11.6 SA establishment during initial registration

After their establishment the set of SAs gets assigned a temporary lifetime. Although the UE will send all subsequent requests and responses via this temporary set of SAs, the set of SAs cannot be taken into use until the authentication procedure between the UE and the S-CSCF has been finished. This is done in order to ensure that the security mechanism between the UE and the P-CSCF is based on successful authentication of the user.

When sending the 200 (OK) response to the UE, the P-CSCF will update the lifetime of the set of SAs by giving it the lifetime of the registration (as indicated in the expires value of the Contact header) plus 30 seconds. The UE will do the same after receiving the 200 (OK) response.

In the case of initial registration (as described here), both ends (i.e., P-CSCF and UE) will immediately afterwards take this set of SAs into use. This means that the P-CSCF will send all SIP messages that are directed toward the UE via the established set of SAs. The UE will in the same way send all SIP messages via the established set of SAs.

### 11.7.3 Handling of Multiple Sets of SAs in the Case of Re-authentication

We have now seen how the first set of SAs is established during initial registration. As the establishment of a set of SAs is based on the authentication data that are sent from the S-CSCF in the 401 (Unauthorized) response, every re-authentication will generate a new set of SAs between the UE and the P-CSCF. Re-authentication procedures are described in Section 11.14. After successful re-authentication the UE and the P-CSCF will maintain two sets of SAs (Figure 11.6):

- the set of SAs that was already established and taken into use before the re-registration took place, which is now called the ‘old set of SAs’; and
- a new set of SAs that was established based on re-authentication, which is now called the ‘new set of SAs’.

The major complication in this situation is that the P-CSCF cannot be sure whether the 200 (OK) response for the second REGISTER request has been received by Tobias’s UE, as SIP defines no acknowledgement mechanism for received responses for any request other than an INVITE. If the UE has not received the 200 (OK) response for the second REGISTER, then it will not take the new set of SAs into use. Therefore, it has to wait until the UE sends a new request on the new set of SAs before it can take them into use. This means that, as long as the P-CSCF does not receive a request from the UE on the new set of SAs, it will:

- send incoming requests to the UE over the old set of SAs (i.e., from its protected client port pc1 to the UE’s protected server port us1); and
- keep both sets of SAs active until one or both of them either expires or a new request from the UE is received.

In our example we assume that the UE has received the 200 (OK) for the second REGISTER request and, therefore, is aware that the authentication procedure was successful and the new set of SAs can be used. Unfortunately, the P-CSCF does not know this and will send incoming requests to the UE over the old set of SAs; therefore, the UE also needs to maintain both sets of SAs.

When the UE needs to send out a new request, it will send it by means of the new set of SAs, which will confirm to the P-CSCF that the new set of SAs can be taken into full use (Figure 11.7). Furthermore, at this moment the old set of SAs will not be immediately dropped, as the UE might have received or sent a request over it, which the remote end has not yet responded to. Therefore, the old set of SAs is kept for another  $64 \cdot 3^{T_1}$  seconds (usually 128 seconds in an IMS environment), before it is dropped.

Note also that the UE cannot take the new set of SAs into use by sending a response – e.g., a 200 (OK) response – for a request – e.g., a MESSAGE request – that was received

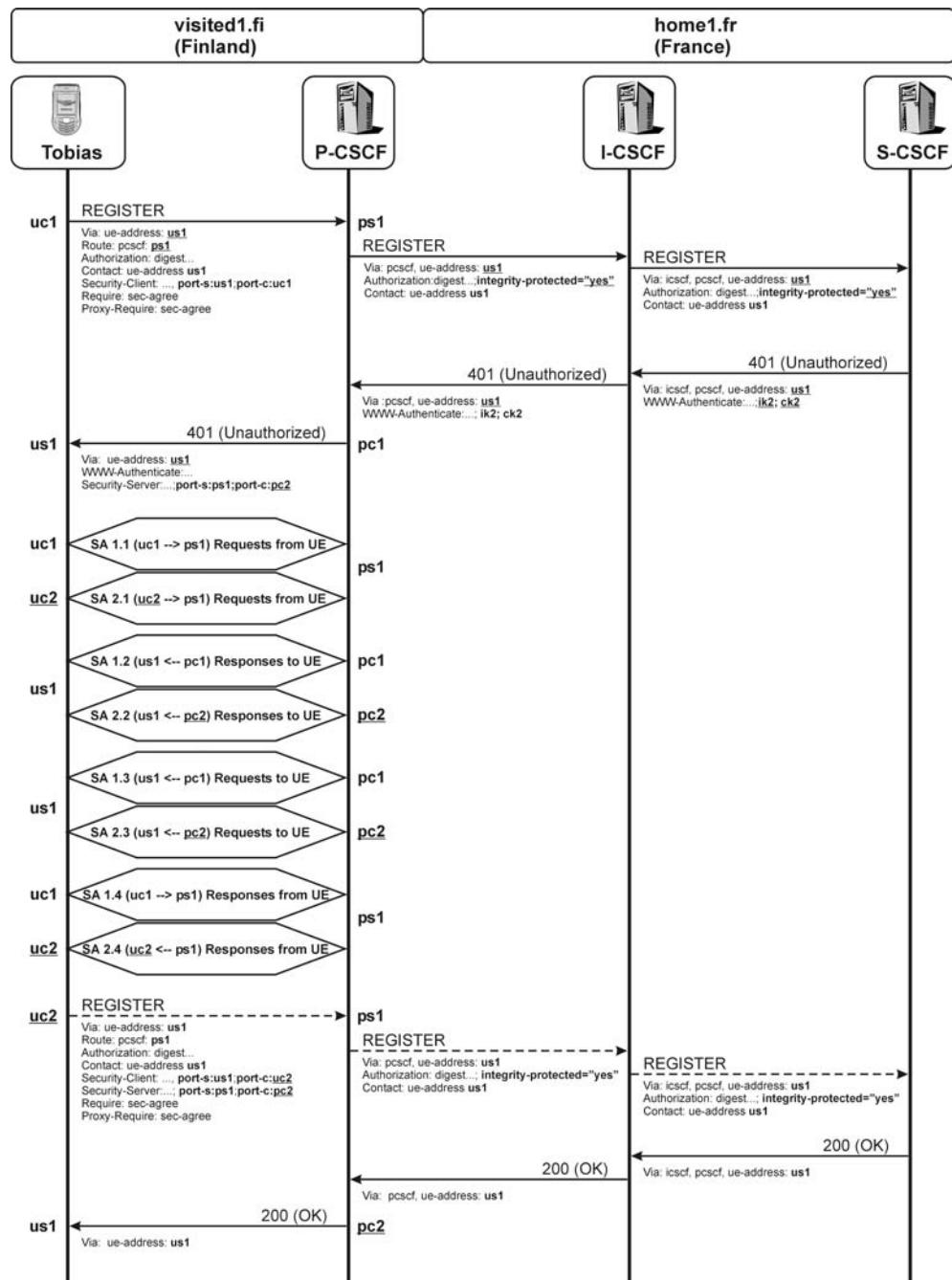


Figure 11.7 Two sets of SAs during re-authentication

over the old set of SAs. The UE is forced either by the Via header of the P-CSCF or due to a TCP connection to send the response to the same port and over the same set of SAs as the request was received.

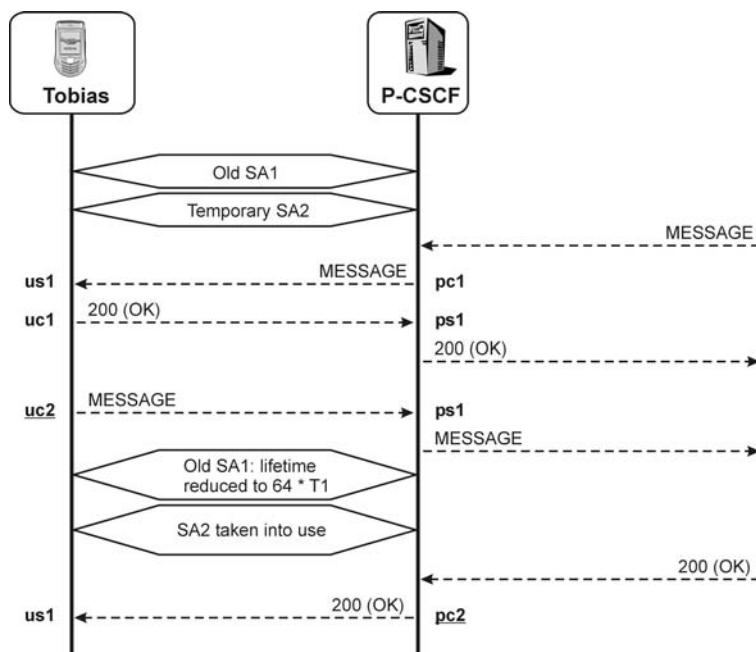
Whenever a set of temporary SAs is established the UE will drop all other SAs, other than the one over which it sent the last REGISTER request. Consequently, the UE never needs to handle more than two sets of SAs at the same time.

#### 11.7.4 SA Lifetime

During an ongoing authentication procedure the lifetime of a temporary set of SAs is restricted to 4 minutes. This guarantees that the authentication procedure can be finished. After successful authentication the lifetime of the new set of SAs is set to:

- Either the expiration time of the concluded registration plus 30 seconds. The expiration time of the registration is indicated in the expires parameter that is returned in the Contact header of the 200 (OK) response to the REGISTER;
- Or, if another set of SAs does already exist, to the lifetime of that already-existing set of SAs as long as its lifetime is longer than the expiration time of the just-concluded registration plus 30 seconds.

Whenever a re-registration takes place and is successful the P-CSCF and the UE have to update the lifetime of all existing SAs with the expiration time of the concluded



**Figure 11.8** Taking a new set of SAs into use and dropping an old set of SAs

re-registration plus 30 seconds, if that value is bigger than the already-assigned lifetime of the SAs.

Consequently, the SAs between the UE and the P-CSCF will be kept 30 seconds longer than Tobias is registered to the IMS network.

When the P-CSCF becomes aware that Tobias is no longer registered (e.g., by receiving a NOTIFY with Tobias's registration-state information which indicates network-initiated de-registration – see Section 11.15.3), the P-CSCF will drop all SAs toward the UE after  $64 * T_1$  seconds.

### 11.7.5 Port Setting and Routing

Special attention has to be paid when it comes to the usage of SA ports, as they heavily influence the routing between the P-CSCF and the UE. As shown in Figure 11.6, Tobias's UE:

- will send all requests from its protected client port (2468);
- expects all responses to be received on its protected server port (1357);
- expects all requests to be received at its protected server port (1357);
- will send all responses to received requests from its protected client port (2468).

The P-CSCF, on the other hand:

- will send all requests toward the UE from its protected client port (8642);
- expects to receive all responses from the UE at its protected server port (7531);
- expects to receive all requests from the UE at its protected server port (7531); and
- will send all responses toward the UE from its protected client port (8642).

To ensure that all requests are sent via IPsec SAs:

- The UE will set its protected server port as part of its address:
  - in the Contact header of every request (including all REGISTER requests);
  - in the Via header of every request, besides the initial REGISTER.
- The UE will set the protected server port of the P-CSCF as part of the outbound proxy (i.e., P-CSCF) address in the Route header of every initial request that it sends.
- The P-CSCF will set its protected server port as part of its address:
  - in the Record-Route header of every initial request that is sent toward the UE;
  - in the Record-Route header of every response that carries the P-CSCF's Record-Route entry toward the UE (for detailed setting of port numbers in the Record-Route header see Section 12.3.4.3).

#### 11.7.5.1 Port Setting During Registration

For example, Tobias's UE initially registers with the following information:

```
REGISTER sip:home1.fr SIP/2.0
Via: sip:[5555:1:2:3:4]:1357;branch=0uetb
```

```

Route: <sip:[5555::a:b:c:d];lr>
Security-Client: digest, IPsec-3gpp; alg=hmac-sha-1-96
    ;spi-c=23456789 ;spi-s=12345678
    ;port-c=2468; port-s=1357
Contact: "Mobile Phone - Tobias" sip:[5555::1:2:3:4]:1357

```

This means that the UE:

- Is going to establish IPsec SA with:
  - port 2468 as the protected client port (port-c parameter of the Security-Client header);
  - port 1357 as the protected server port (port-s parameter of the Security-Client header).
- Expects all incoming requests to be routed to its protected server port (port value in the Contact header);
- Will send this initial REGISTER request to the unprotected port 5060 of the P-CSCF, as no port value is given in the Route header;
- Will await all responses to this initial REGISTER request on unprotected port 5060, as no port value is given in the Via header.

The 401 (Unauthorized) response that is received afterwards by the UE will look like this:

```

SIP/2.0 401 Unauthorized
Via: sip:[5555:1:2:3:4]:1357;branch=0uetb
Security-Server: tls ;q=0.2, IPsec-3gpp; q=0.1
    ;alg=hmac-sha-1-96
    ;spi-c=98765432 ;spi-s=87654321
    ;port-c=8642 ;port-s=7531

```

This means that the P-CSCF is going to establish an IPsec SA with:

- port 8642 as the protected client port (port-c parameter of the Security-Server header); and
- port 7531 as the protected server port (port-s parameter of the Security-Server header).

After this exchange the UE and the P-CSCF will set up the temporary set of SAs and the UE will then send the second REGISTER request already protected, which then will look like:

```

REGISTER sip:home1.fr SIP/2.0
Via: sip:[5555:1:2:3:4]:1357;branch=1uetb
Route: <sip:[5555::a:b:c:d]:7531;lr>
Contact: "Mobile Phone - Tobias" sip:[5555::1:2:3:4]:1357

```

Note that the Security-Client and Security-Verify headers are also included in this request (see Section 11.8), but, as they no longer have any influence on SA establishment and routing, they are not shown here. This means that the UE:

- expects all incoming initial requests to be routed to its protected server port (port value in the Contact header);
- sends this REGISTER request already over the temporary IPsec SA (i.e., to the protected server port of the P-CSCF – port value in the Route header); and
- expects all responses to this REGISTER request to be sent via the temporary IPsec SA (i.e., on its protected server port 1357 – port value in the Via header).

### 11.7.5.2 Port Setting During Re-authentication

As said before, every re-authentication will result in a new pair of IPsec SAs. When exchanging the security parameter indexes and protected port numbers for the new set of SAs according to the SIP Security Mechanism Agreement, the P-CSCF and the UE only change their protected client ports:

- the UE receives requests and responses for both sets of SAs via its protected server port (us1);
- the P-CSCF receives requests and responses for both sets of SAs via its protected server port (ps1);
- the UE uses a new protected client port (uc2) for sending requests and responses toward the P-CSCF over the new set of SAs; and
- the P-CSCF also uses a new protected client port (pc2) for sending requests and responses toward the UE over the new set of SAs.

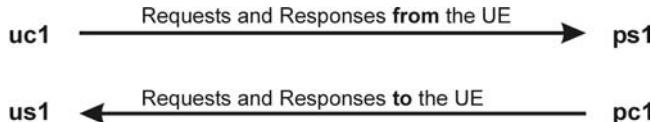
This is due to the fact that two sets of SAs must not use the same port parameters. Furthermore, if the protected server ports change, this would cause major problems and would mean that:

- the UE would need to perform re-registration, as its registered contact includes the protected server port;
- the UE would need to send re-INVITE on all established sessions, as its contact information that was sent to the remote end includes the protected server port;
- the P-CSCF would receive from the UE all subsequent requests to every already-established dialog (including all subscriptions of the UE) on the P-CSCF's old, protected server port, as there is no possibility in SIP to change the route information for an already-established dialog.

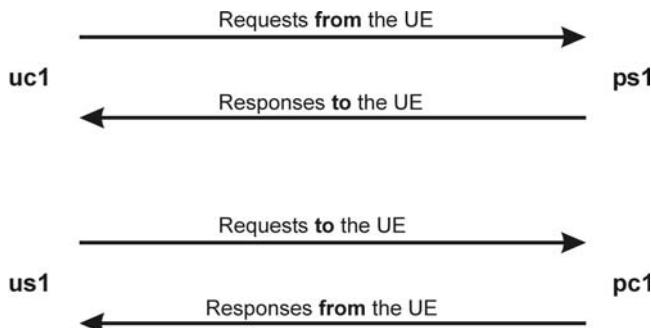
This list is not complete, but it shows that changing the protected server port would cause a lot of problems for SIP routing. Therefore, it is essential that this value is not changed as long as the user stays registered.

### 11.7.5.3 Port Settings for SIP Requests other than REGISTER

The setting of the protected ports in non-REGISTER requests is described in more detail in Section 3.7.



**Figure 11.9** Request and response routing between UE and P-CSCF over UDP



**Figure 11.10** Request and response routing between UE and P-CSCF over TCP

#### 11.7.5.4 Usage of Ports with UDP and TCP

The previous sections showed how requests and responses are routed via one or more sets of SAs. In the chosen example, only UDP was used as a transport protocol. For TCP, however, there is a slight difference in these procedures.

When a request is sent out via UDP (Figure 11.8) the Via header indicates the IP address and port number to which all related responses should be routed. When TCP is used to send the request (Figure 11.9) the information in the Via header is overridden and the response is routed back to the same address and port that the request was received from. This draws attention to the nature of TCP as a connection-oriented transport protocol. By applying this rule it is ensured that no additional TCP connection needs to be opened to send the response to a request that was received via TCP. This causes the routing of SIP messages between the P-CSCF and the UE to behave differently. The UE will set its protected server port (us1) in the Via header of every request that it sends out, regardless of whether UDP or TCP is used. All requests will originate from the UE's protected client port (uc1).

In the case of UDP the responses to such a request will be sent to the UE's protected server port (us1), as indicated in the Via header.

In the case of TCP the responses to such a request will be sent to the UE's protected client port (uc1), as the request originated from there. The same is true in the other direction (i.e., for requests sent from the P-CSCF toward the UE and their responses).

#### 11.7.6 Related Standards

Specifications relevant to Section 11.7 are:

---

3GPP TS 33.102	Security architecture.
3GPP TS 33.203	Access security for IP-based services.
3GPP TS 33.210	Network Domain Security (NDS); IP network layer security.
RFC2401	Security Architecture for the Internet Protocol.
RFC2403	The Use of HMAC-MD5-96 within ESP and AH.
RFC2404	The Use of HMAC-SHA-1-96 within ESP and AH.
RFC2451	The ESP CBC-Mode Cipher Algorithms.

## 11.8 SIP Security Mechanism Agreement

### 11.8.1 Why the SIP Security Mechanism Agreement is Needed

The IMS in 3GPP Releases 5 and 6 makes use of IPsec as the security mechanism between the P-CSCF and the UE. IPsec is only one of several possible security mechanisms. IMS was designed to allow alternative security mechanisms over the Gm interface as well. Allowing such an openness usually creates backward compatibility problems because, for example, a Release 6-compliant UE would not be able to understand any alternative security mechanism, while it could be attached to a P-CSCF of a higher release that would already support alternatives to IPsec.

Therefore, the SIP Security Mechanism Agreement (Sip-Sec-Agree) was introduced to allow the UE and the P-CSCF to negotiate a common security mechanism for use between them. For current releases the only security mechanism is IPsec; however, in Rel-8 TLS and HTTP Digest will be introduced as well.

### 11.8.2 Overview

To make the example not too simple and boring, we assume that the UE supports IPsec and the HTTP digest, and that the P-CSCF supports IPsec and Transport Layer Security (TLS), with a preference toward TLS. It is not necessary for the reader of this chapter to have any knowledge of any of these mechanisms.

As we have seen, the initial REGISTER request is sent without any protection from the UE to the P-CSCF. To guarantee that a common security mechanism can be established, Tobias's UE advertises the mechanisms it supports in this initial REGISTER request within the Security-Client header, which includes a list of supported mechanisms.

The P-CSCF sends back a Security-Server header in the 401 (Unauthorized) response. The header includes a list of supported mechanisms from the P-CSCF end. Furthermore, the P-CSCF adds a preference (q-value) to each of the mechanisms.

Based on this information, both ends now know which common mechanisms are supported by the UE and the P-CSCF. If there is more than one common mechanism, the mechanism which was given the highest preference by the P-CSCF will be selected and applied. To guarantee that this mechanism can be established immediately, the P-CSCF will send further information in the 401 (Unauthorized) response to enable the UE to set up the mechanism: for example, in a non-IMS environment it could send a Proxy-Authenticate header when HTTP digest is the chosen mechanism.

As we saw in Section 11.7, the UE and the P-CSCF will then establish the security mechanism, which is in our case based on IPsec SAs. Afterwards, all messages between the two entities will be sent protected over these SAs.

Nevertheless, the initial REGISTER request and its response are still not protected. There is the slight chance that a malicious user has tampered with the messages or that an error has occurred over the vulnerable air interface.

As shown in earlier chapters, the second REGISTER request from the UE repeats all the information necessary for authentication and registration, both of which are performed with the S-CSCF. In order to guarantee that Sip-Sec-Agree-related information has not been changed as well, the UE:

- repeats the Security-Client header that it sent in the initial REGISTER in the second REGISTER request as well; and
- copies the content of the Security-Server header that was received in the 401 (Unauthorized) response from the P-CSCF into a Security-Verify header and sends it along with the second REGISTER request as well.

As long as the established Security-Association is used, the UE will always repeat the same Security-Verify header in every request that it sends to the P-CSCF.

During the exchange between the Security-Client (from the UE) and the Security-Server (from the P-CSCF) headers, the two ends also agree on some parameters for IPsec SAs: that is, they indicate to each other the protected client and server ports (port-c and port-s) as well as the security parameter indexes (SPIs: spi-c and spi-s).

#### *11.8.3 Sip-Sec-Agree-Related Headers in the Initial REGISTER Request*

In order to activate the agreement on the security mechanism, the UE includes the following information in the initial REGISTER request:

```
REGISTER sip:home1.fr SIP/2.0
Require: sec-agree
Proxy-Require: sec-agree
Security-Client: digest, IPsec-3gpp ;alg=hmac-sha-1-96
                 ;spi-c=23456789 ;spi-s=12345678
                 ;port-c=2468 ;port-s=1357
```

The Proxy-Require header includes the option tag ‘sec-agree’; this indicates that the next hop proxy, in this case the P-CSCF, must support the procedures for the Sip-Sec-Agree in order to process the request further. If the next hop proxy does not support Sip-Sec-Agree procedures, it would – based on handling of the Proxy-Require header, which is defined in [RFC3261] – send back a 420 (Bad Extension) response, including an Unsupported header with the option tag ‘sec-agree’. As the P-CSCF in our example is fully IMS Releases 5 and 6-compliant, it of course supports SIP SA procedures and will not send this response to the UE.

Also, a Require header is included, indicating the sec-agree option tag. This is mandated to be included by [RFC3329], which defines the Sip-Sec-Agree. The Require header is used in the same way as the Proxy-Require header, but by the remote UE (not the proxy). It is there just in case a request (in this case the REGISTER request) is sent directly from the sending UE to the final destination (the S-CSCF), which would not look at the Proxy-Require header at all; this would mean that no negotiation of the security

mechanism would take place. The `Require` header forces the receiving end to perform the sec-agree procedures.

As the P-CSCF is able to perform Sip-Sec-Agree procedures, it removes the sec-agree option tags from the `Require` and `Proxy-Require` headers before sending the request toward Tobias's home operator network.

Tobias's UE sends the list of supported security mechanisms to the P-CSCF within the `Security-Client` header. The P-CSCF will discover, based on the information given in this header, that Tobias's UE supports two security mechanisms: one is the HTTP digest ('digest') and the other is IPsec as used by 3GPP ('IPsec-3gpp'). These two mechanisms are separated by commas in the header. The list of parameters (separated by ';') for the latter includes:

- the algorithm (alg parameter) – used for IPsec encryption and protection (in this case it is the HMAC SHA 1-96 algorithm, which is defined in [RFC2404]);
- the protected client port (port-c) and the protected server port (port-s) – used from the UE's end for IPsec SAs; and
- the SPI – used for the IPsec SA that relates to the protected client port (spi-c) as well as the SPI used for the IPsec SA that relates to the protected server port (spi-s).

The P-CSCF will also remove the `Security-Client` header before sending the `REGISTER` request further.

Note that the IPsec-3gpp security mechanism is only relevant for the IMS. The example given here uses digest and TLS as possible additional security mechanisms in Sip-Sec-Agree-related headers. This is only done to explain the procedures behind the negotiation process.

#### *11.8.4 The Security-Server Header in the 401 (Unauthorized) Response*

When receiving a 401 (Unauthorized) response from the S-CSCF for a `REGISTER` request, the P-CSCF includes a list of supported security mechanisms in a `Security-Server` header in the response:

```
SIP/2.0 401 Unauthorized
Security-Server: tls ;q=0.2, IPsec-3gpp; q=0.1 ;alg=hmac-sha-1-96
;spi-c=98765432 ;spi-s=87654321
;port-c=8642 ;port-s=7531
```

In this example the P-CSCF supports two security mechanisms: 3GPP-specific usage of IPsec and TLS. It even gives a higher preference to TLS: should the UE also support TLS, this would be chosen to protect the messages between the UE and the P-CSCF.

Furthermore, the P-CSCF sends IPsec-related information about SPIs and protected client and server ports in the same way as the UE.

At the point of sending out the 401 (Unauthorized) response to the UE, the P-CSCF is already aware that the IPsec will be used as the security mechanism, as it knows that this is the only mechanism that is supported by both the UE and itself.

### 11.8.5 Sip-Sec-Agree Headers in the Second REGISTER

After receiving the 401 (Unauthorized) response the UE is able to set up IPsec SAs. When this has been done, it can use these SAs to send the second REGISTER request over it. In this REGISTER request it now includes the following related information:

```
REGISTER sip:home1.fr SIP/2.0
Require: sec-agree
Proxy-Require: sec-agree
Security-Verify: tls ;q=0.2, IPsec-3gpp ;q=0.1
    ;alg=hmac-sha-1-96
    ;spi-c=98765432 ;spi-s=87654321
    ;port-c=8642 ;port-s=7531
Security-Client: digest, IPsec-3gpp ;alg=hmac-sha-1-96
    ;spi-c=23456789 ;spi-s=12345678
    ;port-c=2468 ;port-s=1357
```

Once again the Require and Proxy-Require headers with the sec-agree option tag are there. They serve the same purpose as in the initial REGISTER (see Section 11.8.3) and will be repeated in every REGISTER request that is sent from the UE. The P-CSCF will always remove them before sending the request on, in the same way as it did for the initial REGISTER request. If either the Proxy-Require or the Require header (or both) are found empty after the sec-agree option tag has been removed, the P-CSCF will also remove this or these empty headers.

The Security-Verify header includes a copy of the received Security-Server header. The Security-Client header is simply re-sent as in the initial REGISTER request.

The P-CSCF will compare the two Security-Client headers that were received in the initial and this second REGISTER request and see whether they match. It will also compare the content of the Security-Server header that it sent in the 401 (Unauthorized) response and with the content of the Security-Verify header that it received in this second REGISTER request.

Before sending the REGISTER request any further, the P-CSCF will remove the Security-Client and Security-Server headers from it.

### 11.8.6 Sip-Sec-Agree and Re-Registration

The S-CSCF can decide to re-authenticate the UE during any re-registration procedure, and by doing so it will force the UE and the P-CSCF to establish a new set of IPsec SAs, as these IPsec SAs are based on the IK, which changes during each re-authentication procedure (see Section 11.14.2). Establishing a new set of IPsec SAs also means that a new set of spis and new protected client and server ports are negotiated.

When sending the new REGISTER request for re-registration the UE cannot be sure whether the S-CSCF will request re-authentication. Therefore, it will add in every new REGISTER request a new Security-Client header with new values for the spis and the protected client and server ports:

```
REGISTER sip:home1.fr SIP/2.0
Require: sec-agree
```

```

Proxy-Require: sec-agree
Security-Verify: tls ;q=0.2, IPsec-3gpp ;q=0.1 ;alg=hmac-sha-1-96
    ;spi-c=98765432 ;spi-s=87654321
    ;port-c=8642 ;port-s=7531
Security-Client: digest, IPsec-3gpp ;alg=hmac-sha-1-96
    ;spi-c=23456790 ;spi-s=12345679
    ;port-c=2470 ;port-s=1357

```

Note that the values for the spis and the protected client port number have changed in the Security-Client header, in order to allow the set-up of a new set of SAs, should the S-CSCF re-authenticate the UE. The protected server port of the UE has not changed and will be kept throughout the user's registration (see Section 11.7.5).

The content of the Security-Verify header is sent unchanged, as it is a copy of the latest received Security-Server header.

Both the P-CSCF and the UE will know, at the moment of receiving the response to this REGISTER request from the S-CSCF, whether new IPsec SAs have to be established: that is, whether a 401 (Unauthorized) response is received or whether a 200 (OK) response is received.

When a 401 (Unauthorized) response is received from the S-CSCF, the P-CSCF will add a new Security-Server header to the response, providing new values for its protected ports and new spis:

```

SIP/2.0 401 Unauthorized
Security-Server: tls ;q=0.2, IPsec-3gpp ;q=0.1 ;alg=hmac-sha-1-96
    ;spi-c=98765434 ;spi-s=87654322
    ;port-c=8644 ;port-s=7531

```

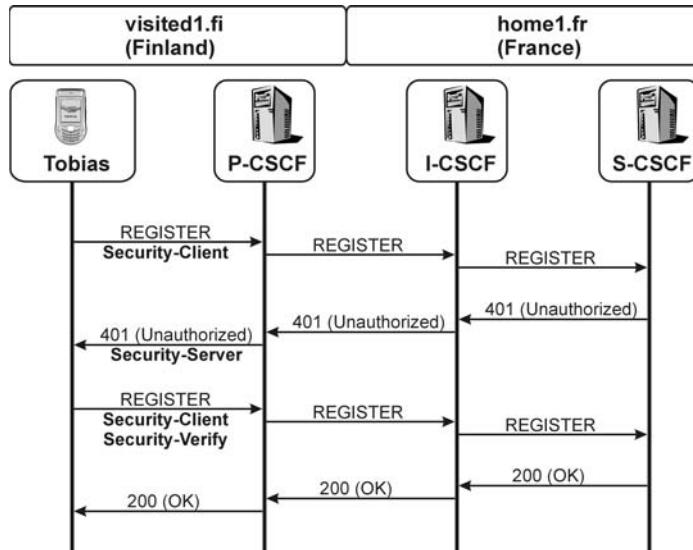
Also, the P-CSCF will not change the value of its protected server port (7531). Consequently, the UE and the P-CSCF will now establish a new set of temporary SAs (see Section 11.7.3). The REGISTER request, which includes the response to the re-authentication challenge (see Section 11.6), will be sent over this new, temporary set of SAs and will include the following headers:

```

REGISTER sip:home1.fr SIP/2.0
Require: sec-agree
Proxy-Require: sec-agree
Security-Verify: tls ;q=0.2, IPsec-3gpp ;q=0.1 ;alg=hmac-sha-1-96
    ;spi-c=98765434 ;spi-s=87654322
    ;port-c=8644 ;port-s=7531
Security-Client: digest, IPsec-3gpp ;alg=hmac-sha-1-96
    ;spi-c=23456790 ;spi-s=12345679
    ;port-c=2470 ;port-s=1359

```

Once again, as during the initial registration procedure (Figure 11.10), the second REGISTER request repeats the Security-Client header that was sent in the latest REGISTER request (with the new values) and copies in the Security-Verify header the values of the Security-Server header that was received in the last 401 (Unauthorized) response. This second REGISTER request within the re-registration procedure no longer carries any information related to any previously established set of SAs.



**Figure 11.11** Sip-Sec-Agree during initial registration

#### 11.8.7 Related Standards

Specifications relevant to Section 11.8 are:

- |                |   |
|----------------|---|
| 3GPP TS 33.203 | Access security for IP-based services.                                  |
| RFC2246        | The TLS Protocol Version 1.0.   |
| RFC2617        | HTTP Authentication: Basic and Digest Access Authentication.            |
| RFC3329        | Security Mechanism Agreement for the Session Initiation Protocol (SIP). |

## 11.9 IMS Communication Service Identification and other Callee Capabilities

### 11.9.1 Overview

IMS allows a user to have not only one, but several terminals active at the same time, of which all use the same public user identity (SIP AOR) of the user (Multiple Terminals – see Chapter 10). Each of these terminals might support specific capabilities and some of them might be restricted in the services they offer to a user. For example a mobile phone might be capable of sending and receiving video streams during SIP sessions. On the other hand, a desktop phone might not support this capability and is also not being used in mobile scenarios.

If we assume that Tobias would be registered from such a mobile phone as well as from a desktop phone, it would be useful that he indicates these phone-specific capabilities to the IMS network, so that incoming calls to him can be routed to the terminal that best matches the calling parties preferences (caller preferences) for the call.

In Section 12.3.9 it will be shown how a calling user can express preferences, in order to route a call towards terminals of the called user, that support one or more specific capabilities. In order to route the call to those called-party terminals, the capabilities need first to be registered within in the network. In this chapter we see how Tobias's phone can indicate its capabilities (callee capabilities) to the network during registration.

Callee capabilities and caller preferences are expressed by means of so-called feature tags. Feature tags are used to select the terminal of the called user in order to satisfy in the best possible way the preferences of the calling user for the call.

Within IMS feature tags are used for expressing the IMS Communication Service Identification (ICSI) and the IMS Application Reference Identification (IARI) (see Section 11.9.3).

### *11.9.2 Feature Tags: Callee Capabilities*

During registration, Tobiases UE wants to indicate the following callee capabilities to the network:

- the UE supports sending and receiving of audio streams (audio);
- the UE supports sending and receiving of video streamed (video);
- the UE is a mobile phone (mobility); and
- the SIP methods which the UE supports to be received (methods = ).

As these feature tags express device capabilities, they are indicated within the Contact header, i.e. the registered IP address of the terminal takes the related feature tags as parameters. Due to this, the Contact header of the initial REGISTER message looks as follows:

```
REGISTER sip:home1.fr SIP/2.0
Contact: <sip:[5555::1:2:3:4]:1357>
;audio
;video
;mobility
;methods="INVITE,BYE,ACK,OPTIONS,CANCEL,NOTIFY,
MESSAGE,PRACK,UPDATE"
```

The Contact header now includes a long list of parameters, which are all separated by ‘;’ from each other. The first three parameters (‘;audio ;video ;mobility’) are feature tags without specific values, i.e. they express the functionality of the UE directly.

The last parameter nevertheless indicates a tag-value-list, i.e. the listed methods are all supported by the UE. This means that an incoming call, that demands the usage of one or more of these listed methods to be supported, will be delivered to this UE.

### *11.9.3 IMS Communication Service Identification (ICSI) and IMS Application Reference Identification (IARI)*

SIP feature tags are also used for expressing IMS Communication Service Identifiers (ICSI) and IMS Application Reference Identifiers (IARI).

As shown in Section 12.3.9, the ICSI supports two different purposes within IMS: service identification within the network and routing of the request to a terminal, that supports the desired service. When indicating the ICSI as a feature tag during IMS registration, it is only used for the second purpose, i.e. it enables the routing of requests towards the terminal that is being registered.

The IARI value is only used for routing of a request to a terminal that supports the related service.

In this example we assume that Tobias's terminal supports three IMS Communication Services as well as one specific IMS application,

- IMS Multimedia Telephone Communication Service (urn:urn-xxx:3gpp-service-ims.icis.mmtel);
- an online game, for which an ICSI with the value urn:urn-xxx:other-vendor-service-ims.icsi.ongame is assumed to be defined;
- a service which allows remote controlling of devices via SIP, for which an ICSI with the value urn:urn-xxx:other-vendor-service-ims.icsi.remotecontrol;
- a specific application for alarming the firefighters of Tobias's home town, as he is a member of the voluntary firefighters. For this application an IARI is assumed to be defined with the value urn:urn-xxx:other-app-ims.iari.firefighter.

Note that the ‘xxx’ values within these URNs will be replaced by numeric values, once these URNs have been registered with the Internet Assigned Number Authority (IANA).

As we have seen in Section 11.2, the operator sent down a list of ICSI values within the IMS MO. Tobias's phone shall only make use of those ICSI values which are present in the list within the IMS MO and which the phone supports at the same time, which leaves only the ICSI values for IMS Multimedia Telephony and the online game. IARI values are not restricted by the operator and therefore Tobias can use the IARI value as listed above.

These ICSI and IARI values are defined as service URNs, which can be expressed within:

- a new SIP feature tag called ‘g.3gpp.icsi\_ref’ which takes the list of ICSI values as a tag-value-list; and
- a new SIP feature tag called ‘g.3gpp.iari\_ref’ which takes the list of IARI values as a tag-value-list, in the same way as the ‘methods’ feature tag, that was explained in Section 12.9.2.

Due to the definition of ICSI and IARI as URNs there is another problem when encoding the Contact header. The tag-value-list as defined in [RFC 3840] does not support the colon (‘:’) as a valid character. Therefore the colon within the ICSI and IARI URNs needs to be percentage-escaped, as defined in [RFC 3986]. This means, that every occurrence of the colon character is replaced with the string ‘%3A’ which is the percentage-escaped representation of the colon character. This leads to the following representation of the Contact header in the current example:

```
REGISTER sip:home1.fr SIP/2.0
```

```
Contact: <sip:[5555::1:2:3:4]:1357>
;audio
;video
;mobility
;methods=
"INVITE,BYE,ACK,OPTIONS,CANCEL,NOTIFY,MESSAGE,PRACK,UPDATE"
;g.3gpp.icsci.ref="urn%3Aurn-xxx%3A3gpp-service-ims.icis.mmtel,
urn%3Aurn-xxx%3Aother-vendor-service-ims.icsi.ongame"
;g.3pp.iari.ref=
"urn%3Aurn-xxx%3Aother-app-ims.iari.firefighter"
```

Note that for the sake of readability, the Contact header in most of the following examples within this book will not include any of the feature tags as described in the last two chapters. Also note that of the ICSI and IARI values given in this example, only the value of the IMS Multimedia Telephony Communication Service Identifier (urn:urn-xxx:3gpp-service-ims.icis.mmtel) is standardized and in use. The other ICSI and IARI values indicated above are invented by the authors.

#### 11.9.4 Related Standards and Links

Standards related to this section are:

RFC 3840

Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)

draft-drage-sipping-service-identification

A Session Initiation Protocol (SIP) Extension for the Identification of Services

draft-monrad-sipping-3gpp-urn-namespace

A Uniform Resource Name (URN) Namespace for the 3rd Generation Partnership Project (3GPP)

<http://www.3gpp.org/tb/Other/URN/URN.htm>

URN values maintained by 3GPP

## 11.10 Compression Negotiation

### 11.10.1 Overview

The ability to compress SIP messages over the air interface is essential for IMS. How Signalling Compression (SigComp) works is described in Section 3.18. This section shows how the UE and the P-CSCF indicate that they support SigComp and are both willing to use it.

The P-CSCF and an IMS UE must support SIP SigComp, but they are not mandated to use it. Therefore, they need a mechanism to express whether they are willing to apply SigComp.

[RFC3486] defines a new URI parameter ‘comp’, which can be set to ‘comp = SigComp’ by either the UE or a SIP proxy (in the case of IMS this applies only to the P-CSCF) in order to express its willingness to send or receive certain SIP messages compressed.

Tobias's UE will express its willingness to use SigComp with the P-CSCF that is already in the initial REGISTER request. The P-CSCF will give a similar indication in the 401 (Unauthorized) response. As these two SIP messages are sent without any protection, the P-CSCF and the UE will not create states (compartments) for SigComp at this point in time; this is to ensure that a malicious user – who wants, say, to start a Denial Of Service (DOS) attack against the P-CSCF – cannot overload the P-CSCF by forcing it to reserve memory for a huge number of unnecessary SigComp compartments.

State creation will only be done after an IPsec SA (see Section 11.7) between the UE and the P-CSCF has been established.

### *11.10.2 Indicating willingness to use SigComp*

The ‘comp’ parameter can be set:

- by the UE in the Contact header of the REGISTER request – this means that the UE is willing to receive every initial request that is destined for it compressed, as initial requests that are destined to the UE are routed based on the registered contact address;
- by the UE in the Contact header of any other initial request or the first response to an initial request – this means that the UE is willing to receive every subsequent request within this dialog compressed, as subsequent requests are routed based on the address in the Contact header of the initial request (from the originating end) or the first response to an initial request (from the terminating end);
- by the UE in the Via header of any request – this means that the UE is willing to receive all responses to this request compressed, as responses are routed based on the Via header in the related request;
- by the P-CSCF in its own entry to the Record-Route header that is sent toward the UE – this means that the P-CSCF is willing to receive subsequent requests within this dialog compressed, as subsequent requests are routed toward SIP proxies based on the entries in the Route header (which is generated from the Record-Route header); and
- by the P-CSCF in the Via header of any request – this means that the P-CSCF is willing to receive all responses to this request compressed, as responses are routed based on the Via header in the related request.

### *11.10.3 comp = SigComp Parameter During Registration*

The initial REGISTER request by the UE will include the following compression-related information:

```
REGISTER sip:home1.fr SIP/2.0
Via: SIP/2.0/UDP sip:[5555::1:2:3:4]:1357;comp=SigComp ;branch=0uetb
Route: sip:[5555::a:b:c:d];lr
Contact: "Mobile Phone - Tobias"
<sip:[5555::1:2:3:4]:1357;comp=SigComp>;expires=600000
```

The `comp = SigComp` parameter is included in the `Via` header and indicates that the UE is willing to receive all responses to this request compressed. Consequently, the P-CSCF may send the 401 (Unauthorized) response already compressed, but it will not create a state (i.e., a compartment) because of this.

The `comp = SigComp` parameter can also be found in the `Contact` header. This parameter will be included in every initial request that is received by the UE, as the S-CSCF will replace the request URI (which points to `sip:tobias@home1.fr`) of every initial request with the registered contact address (i.e., `sip:[5555::1:2:3:4]:1357; comp = SigComp`).

The 401 (Unauthorized) response from the P-CSCF does not include any further information on the P-CSCF's ability to perform `SigComp`. The P-CSCF address that was discovered before the initial registration (see Section 11.4) cannot be discovered with the `comp = SigComp` parameter. As SIP messages should only be sent compressed when the `comp = SigComp` parameter is set in the address of the next hop, the UE would therefore not send any initial request to the P-CSCF compressed.

Subsequent requests (such as ACK, PRACK, UPDATE or BYE) could be sent compressed, as the routing from the UE to the P-CSCF would be based on the `Record-Route` entry of the P-CSCF (see Section 12.3.3.2), in which the P-CSCF can include the `comp = SigComp` parameter. The same is true for responses sent from the UE to the P-CSCF, as they are routed based on the `Via` header entry of the P-CSCF, which is also set by the P-CSCF itself.

Although it is a requisite for the `comp` parameter to indicate whether compression is used, 3GPP TS 24.229 does not make a clear requirement on compression of the initial message. One possibility would be that the UE just sends every initial request compressed, as the P-CSCF must support `Sigcomp` in any event.

Therefore, the UE adds the `comp = SigComp` parameter to the P-CSCF address that was discovered previously. Therefore, it can send out the second REGISTER request already compressed:

```
REGISTER sip:home1.fr SIP/2.0
Via: SIP/2.0/UDP sip:[5555:1:2:3:4]:1357;comp=SigComp;branch=1uetb
Route: sip:[5555:a:b:c:d]:7531;lr;comp=SigComp
Contact: "Mobile Phone - Tobias"
<sip:[5555:1:2:3:4]:1357;comp=SigComp>;expires=600000
```

This REGISTER request is routed on the basis of the topmost Route header, which includes the P-CSCF address and the `comp = SigComp` parameter. As the parameter is already there, the UE can send the request compressed.

The 200 (OK) response to this REGISTER request will be sent from the P-CSCF to the UE on the basis of the `Via` header, and, as the UE also includes the `comp = SigComp` parameter, the P-CSCF will send it compressed.

#### 11.10.4 `comp = SigComp` Parameter in Other Requests

The handling of the `comp = SigComp` parameter in requests other than REGISTER is described in Section 12.4.

### 11.10.5 Related Standards

The comp parameter is defined in [RFC3486]: Compressing the Session Initiation Protocol (SIP).

## 11.11 Access and Location Information

### 11.11.1 P-Access-Network-Info

The P-Access-Network-Info header is a 3GPP-specific header and indicates to the IMS network over which access technology the UE is attached to IMS. In our example the access technology is GPRS. It also includes the Cell Global ID (CGI), which indicates the location of the user.

Tobias's UE will include the P-Access-Network-Info header in every request (besides ACK and CANCEL requests) and every response (besides responses to the CANCEL request) that it sends out, but only if that request or response is sent integrity-protected (i.e., via an SA, see Section 11.7).

The first time this header is sent out is, therefore, within the second REGISTER request, which is sent after the 401 (Unauthorized) response has been received by the UE. The header looks like:

```
REGISTER sip:home1.fr SIP/2.0
P-Access-Network-Info: 3GPP-UTRAN-TDD;utran-cell-id-
3gpp=234151D0FCE11
```

Tobias's S-CSCF will remove the P-Access-Network-Info header from every request or response that it sends toward another entity. The only exception from this rule are ASs that are in the same trust domain as the S-CSCF (see Section 11.5.10).

When the P-Access-Network-Info header is sent in an INVITE request that is sent for an emergency call, the P-CSCF and S-CSCF can determine from the Cell-ID which emergency centre is closest to the user and should be contacted. When writing this chapter the details for IMS emergency calls were still under discussion in 3GPP standardization groups. In the future there may be more applications that use the information contained in this header.

### 11.11.2 P-Visited-Network-ID

The P-Visited-Network-ID header indicates to Tobias's home network the identification of the network within which Tobias is currently roaming. The header is included by the P-CSCF to which Tobias's UE is attached. The information in this header will be used by the S-CSCF to check the roaming agreement with that visited network.

In this example it is assumed (see Section 10.1) that Tobias is roaming in Finland and is attached to the fictitious Finish operator Musta Kissa. As the P-CSCF is also provided by this operator, it will include a P-Visited-Network-ID header in every REGISTER request that it sends toward Tobias's home network. Within this header will be a string, from which the S-CSCF will recognize the visited network:

```
REGISTER sip:home1.fr SIP/2.0
P-Visited-Network-ID: "Kaunis Musta Kissa"
```

### 11.11.3 Related Standards

3GPP-specific SIP headers are defined in [RFC3455]: Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the 3rd-Generation Partnership Project (3GPP).

## 11.12 Charging-Related Information During Registration

The charging concept and the relevant entities in the network are described in Section 3.18. The current section only explains the handling and content of SIP headers that are related to charging during registration. Charging of IMS sessions is described in Section 12.7.

When the P-CSCF receives the initial REGISTER request, it creates an IMS Charging ID (ICID), which is valid for all IMS-related signalling as long as the user stays registered. The ICID value is transported from the P-CSCF to the S-CSCF in the P-Charging-Vector header:

```
REGISTER sip:home1.fr SIP/2.0
P-Charging-Vector:icid-value="AyretyU0dm+6O2IrT5tAFrbHLso=023551024"
```

The S-CSCF, when receiving this header, will store the ICID and will perform the charging procedures as described in Section 12.7. The P-Charging-Vector header is defined in [RFC3455]. Extensions to this header and its usage within the IMS are described in [3GPP TS 24.229].

## 11.13 User Identities

### 11.13.1 Overview

Tobias needs to register within his home network in order to be able to originate a call toward his sister. In the example so far he has used the SIP URI *sip:tobias@home1.fr* for registration. This is the user identity Tobias uses when he uses IMS services that are not work-related. Nevertheless, Tobias has a whole set of user identities that are registered at his operator in France, which are shown in Table 11.3.

**Table 11.3** Tobias's public user identities

Registration set	SIP URI	tel URL and related SIP URI (Alias)
1	<i>sip:tobias@home1.fr</i>	<i>tel:+44123456789 sip:+44123456789@home1.fr</i>
2	<i>sip:tobi@home1.fr</i>	<i>tel:+44123456111 sip:+44123456111@home1.fr</i>
3	<i>sip:gameMaster@home1.fr</i>	

As shown in Table 11.3, a registration set can consist of several SIP URIs and tel URLs. For every tel URL there must be a related SIP URI available in the network, i.e. the SIP URI user part takes the numeric value of the tel URL. These tel URLs and their related SIP URIs are Alias Public User Identities (Alias IDs), which means that they share exactly the same user profile within the HSS.

During the initial registration procedure, Tobias can explicitly register only one of these URIs, which in our example is *sip:tobias@home1.fr*. Nevertheless, the IMS allows implicit and explicit registration of further public user identities:

- Some of the above-listed identities might automatically (implicitly) be registered by the network during the initial registration phase;
- Others might stay unregistered until Tobias explicitly requests them to be registered.

When receiving the 200 (OK) response for the second REGISTER request, both Tobias's terminal and the P-CSCF discover Tobias's default public user identity, which is received as the first URI in the P-Associated-URI header.

To find out more about the registration status of the other public user identities that are assigned to Tobias, the UE automatically subscribes to the registration-state event information that is provided by the S-CSCF in the home network. It is mandatory that the UE performs this subscription immediately after the initial registration has succeeded, because:

- the UE needs to get the registration status of the associated URIs;
- the subscription enables the network (S-CSCF) to force the UE to perform reauthentication (see Section 11.14.2);
- the subscription enables the network (S-CSCF) to de-register the user (see Section 11.14.3).

In parallel, the P-CSCF makes a subscription to the user's registration-state information as well, mainly to be informed about network-initiated de-registration (see Section 11.14.3).

### *11.13.2 Public and Private User Identities for Registration*

The identities that go into the first REGISTER request are read from the ISIM, one of the applications contained on the UICC within the UE. Data read from the ISIM include:

- the private user identity of the user;
- the public user identity of the user, which is used for registration; and
- the address of the SIP registrar of the user.

The private user identity is only used for authentication, which is described in Section 11.6. The public user identity is the SIP URI that Tobias is going to initially register. There may be more public user identities available for Tobias, some of them may even be stored on the ISIM; however, at the beginning only one is explicitly registered.

If the UE is not equipped with an ISIM, it will derive the identities and the address of the registrar from the USIM application that also resides in the UICC. The USIM includes all user-related data that are needed for Circuit-Switched (CS) and Packet-Switched (PS) domain registration and authentication. This is described in more detail in Section 11.13.2.

Armed with these parameters the UE can fill in the following fields of the initial REGISTER request:

```
REGISTER sip:home1.fr SIP/2.0
From: <sip:tobias@home1.fr>;tag=pohja
To: <sip:tobias@home1.fr>
Authorization: Digest username="tobias_private@home1.fr",
    realm="home1.fr",
    nonce="",
    uri="sip:home1.fr",
    response=""
```

The public user identity, as read from the ISIM, is put into the To and From headers. The value of the username field of the Authorization header takes the value of the private user identity, and the address of the registrar is put into the request URI of the request as well as in the realm and uri fields of the Authorization header.

### 11.13.3 Identity Derivation from USIM

When Tobias registers, his UE takes the SIP URI *sip:tobias@home1.fr* from the ISIM application that is running on the UICC that he got from his operator and put into his UE. The ISIM always holds at least one valid public user identity.

However, IMS services can also be provided to users who own UICC cards on which no ISIM application and, therefore, no valid public user identity are present. Therefore, the UE needs to create a temporary public user identity from the data available from the USIM application (see Section 3.5.4) and use this temporary identity for registration.

As the temporary public user identity is constructed from security-related data on the USIM, it must not be exposed to any entity outside the IMS. Therefore, it is treated as a ‘barred identity’: that is, it is strongly recommended that the network reject any usage of this identity outside the user’s registration.

In this case the private user identity will be derived from USIM data as well. It will take the format of an International Mobile Subscriber Identity (IMSI) as the user part, followed by a host part, which includes the Mobile Country Code (MCC) and the Mobile Network Code (MNC), both of which are included in the IMSI: for example, the private user identity of Tobias could look something like: *2233099999999@ims.mnc33.mcc222.3gppnetwork.org*

The domain name of Tobias’s home network would be derived from the USIM as well and would appear just like the domain part of the user identity: that is, *ims.mnc33.mcc222.3gppnetwork.org*

### 11.13.4 Default Public User Identity/P-Associated-URI Header

If Tobias had used a temporary public user identity for his initial registration, he would now have the problem that he would be registered but could not perform any other action

(e.g., call his sister or subscribe to a service), as he is registered with an identity that he must not use further (barred identity). His terminal needs to know an identity that has been implicitly registered.

Whenever a user has successfully been authenticated and registered, the S-CSCF, therefore, sends in the 200 (OK) response for the REGISTER request the P-Associated-URI header, which lists all the SIP URIs and tel URLs (i.e., public user identities), which are associated but not necessarily registered for the user. Only the first URI listed in this header is always a valid, registered public user identity and can be used by the UE and the P-CSCF for further actions.

The P-Associated-URI in the 200 (OK) response to Tobias's REGISTER request looks like:

```
SIP/2.0 200 OK
P-Associated-URI: <sip:tobias@home1.fr>, <sip:tobi@home1.fr>,
                  <sip:gameMaster@home1.fr>,
                  <sip:+44123456789@home1.fr;user=phone>,
                  <sip:+44123456111@home1.fr;user=phone>
```

From this information Tobias knows that at least the public user identity *sip:tobias@home1.fr* is registered. He also becomes aware that there are two more SIP URIs and two more tel URIs that he can use, but he does not know whether they are currently registered or not.

As the P-Associated-URI is only defined to transport SIP URIs, it includes the tel URLs that are associated with Tobias (*tel:+44123456789* and *tel:+44123456111*) in the format of SIP URIs.

### 11.13.5 Assignment of a Globally Routable User Agent URI

So far we have seen two types of identities, those which are related to the user, e.g. the public user identity and then the private identity, which is related to the device. There are cases in which it is important that not only a specific user, but also a specific device can be addressed. An example for such a case is e.g. when a user wants to seamlessly transfer an ongoing call from the users fixed phone to a specific mobile phone (such an example will be shown in Section 12.10). Although the private identity identifies (at least in the cases when the phone is equipped with a UICC) the device, it must not be used outside the registration context, as it reveals security related information.

In order to fill this gap, the 'Globally Routable User Agent URI' (GRUU) was introduced. GRUUs allow the identification of the device in a way that the device can be addressed by SIP means.

There are two types of GRUUs:

- public GRUUs which expose the identity of the user, i.e. the SIP address of the user to whom the GRUU is assigned, can be easily determined from the GRUU, e.g. *sip:tobias@home1.fr;gr = jklhzzqw7as9asfd*;
- temporary GRUUs which hide the identity of the user, i.e. the GRUU has a SIP address which does not allow determination of the SIP address of the user, to whom the GRUU was assigned, e.g. *sip:98hzah4pmmn@home1.net;gr*.

An IMS terminal is mandated to request both a public as well as a temporary GRUU during registration. The GRUUs are assigned by the S-CSCF without any HSS involvement, GRUUs are also not stored within the HSS.

In order to request the assignment of a GRUU, Tobias's phone includes an instance-id parameter within the Contact header of the REGISTER requests it sends:

```
REGISTER sip:home1.fr SIP/2.0
Contact: "Mobile Phone - Tobias" <sip:[5555::1:2:3:4]:1357>
:+sip-instance=<urn:uuid: jklhzzqw7as9asfd>;expires=600000
```

If the registration is successful, the S-CSCF will assign both the public as well as the temporary GRUU and send them back within the Contact header of the 200 (OK) response to the REGISTER request:

```
SIP/2.0 200 OK
Contact: "Mobile Phone - Tobias" <sip:[5555::1:2:3:4]:1357>
;pub-gruu="sip:tobias@home1.fr;gr=urn:uuid: jklhzzqw7as9asfd"
;temp-gruu="sip:98hzah4pmmn@scscf1.home1.net;gr"
:+sip-instance=<urn:uuid: jklhzzqw7as9asfd>;expires=600000
```

As it is shown here, the GRUU relates to both, the public user identity under registration as well as the contact address, i.e. the address of the device, that is being registered. A GRUU is always bound to both, the user and the device.

A GRUU is identified by the 'gr' parameter in the SIP URI. The public GRUU assigned here is the SIP URI of Tobias, plus the 'gr' parameter set to the registered instance ID. From this the user, to whom the GRUU has been assigned, can easily be read.

The temporary GRUU uses in the user name a string, which is created by the S-CSCF and does not show any relation to Tobias or Tobias's phone. In order to make sure that this temporary GRUU can be routed without having to store the address in the HSS, the temporary GRUU shows the address of the assigning S-CSCF (scscf1.home1.fr) in the user part. This makes sure that e.g. an I-CSCF does not try to resolve the GRUU from the SLF/HSS (see Section 12.3.3.4), as it can directly forward it based on the information in the host part of the GRUU to the S-CSCF which assigned the GRUU. Also the temporary GRUU includes the 'gr' parameter, but this time it is empty, as the string in the username is globally unique and therefore no further distinction is needed.

Both the S-CSCF as well as Tobias's phone store both the public as well as the temporary GRUU.

#### *11.13.6 UE's Subscription to Registration-State Information*

After the initial registration and authentication has succeeded, Tobias's terminal sends out a SUBSCRIBE request with the following information:

```
SUBSCRIBE sip:tobias@home1.fr SIP/2.0
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;comp=sigcomp;branch=4uetb
Route: <sip:[5555::a:b:c:d]:7531;lr>
Route: <sip:orig@scscf1.home1.fr;lr>
```

```

From: "Tobias" <sip:tobias@home1.fr>;tag=sipuli
To: "Tobias" <sip:tobias@home1.fr>
P-Preferred-Identity: "Tobias" <sip:tobias@home1.fr>
Event: reg
Expires: 600000
Accept: application/reginfo+xml
Contact: "Mobile Phone - Tobias" <sip:[5555::1:2:3:4]:1357>
Content-Length: 0

```

Again, not all the information that is included in the SUBSCRIBE request is shown here – the above headers are only those that are necessary to understand the nature of the registration-state event subscription and the routing of the request.

The subscription is intended for an event named ‘reg’, which is the registration-state event package; it is identified in the Event header of the request.

The request URI identifies the user whose registration-state information is requested and, therefore, has to be set to a registered public user identity of Tobias and must be indicated in the To header.

In order to identify itself, Tobias’s UE sets the To and P-Preferred-Identity headers to a SIP URI that it knows is currently registered. This is:

- either the default public user identity that was received in the P-Associated-URI header (see Section 11.13.4);
- or the public user identity that was explicitly registered during initial registration as long as that was not a temporary public user identity (see Section 11.13.3). If no temporary public user identity was used, it is possible that this explicitly registered public user identity is identical to the default public user identity.

The relationship between the P-Preferred-Identity header, the P-Asserted-Identity header and the To header is described in Section 12.2. The To header does not include a tag, as SUBSCRIBE is an initial request and, therefore, the tag will be assigned by the remote end (i.e., in this case the S-CSCF).

The Expires header is set to the same value as the expiry time of the initial registration (i.e., 600 000 seconds which is about 7 days).

The Accept header indicates that only information of the type ‘reginfo+xml’ can be processed by the UE for this subscription, which is the Extensible Markup Language (XML) format for registration-state information.

The Contact header is set to the same contact information as used during registration: that is, the IP-Address of the UE which was assigned by the access network (see Section 11.3) and the protected server port that is used by the IPsec SA (see Section 11.7).

Finally, the Route headers are worth looking at: they include the route set that was received in the Service-Route header within the 200 (OK) response for the REGISTER request (see Section 11.5.8) and on top of it the address of the P-CSCF, which acts as an outbound proxy. This forces the SUBSCRIBE request to be routed first to the P-CSCF and then onward directly to the S-CSCF that was assigned during registration.

The P-CSCF, when receiving this SUBSCRIBE request from the UE, will check whether the information set in the P-Preferred-Identity header is a valid public user

identity of Tobias. If this is the case, then it replaces the P-Preferred-Identity header with the P-Asserted-Identity header:

```
SUBSCRIBE sip:tobias@home1.fr SIP/2.0
P-Asserted-Identity: "Tobias" <sip:tobias@home1.fr>
```

The S-CSCF, when receiving this SUBSCRIBE request, will check whether the user identified by the P-Asserted-Identity header is registered at the S-CSCF. Afterwards, it checks whether it can provide the requested registration-state information of Tobias to the subscribing user (Figure 11.11). As Tobias is subscribing to his own registration-state information in this case, this is allowed. Therefore, the S-CSCF will immediately:

- return a 200 (OK) response for the SUBSCRIBE request, indicating that the subscription was successful;
- generate an XML document of type reginfo, including the current registration-state information for the URIs that are associated with Tobias; and
- send the generated XML document in a NOTIFY message toward the subscriber (in this case Tobias's UE).

As the 200 (OK) response and the NOTIFY request are sent approximately at the same time, the NOTIFY request may be received at the terminal before the 200 (OK) response. In this exceptional case, the UE must be able to create the related subscription dialog based on the NOTIFY request: that is, it must not discard the information received in

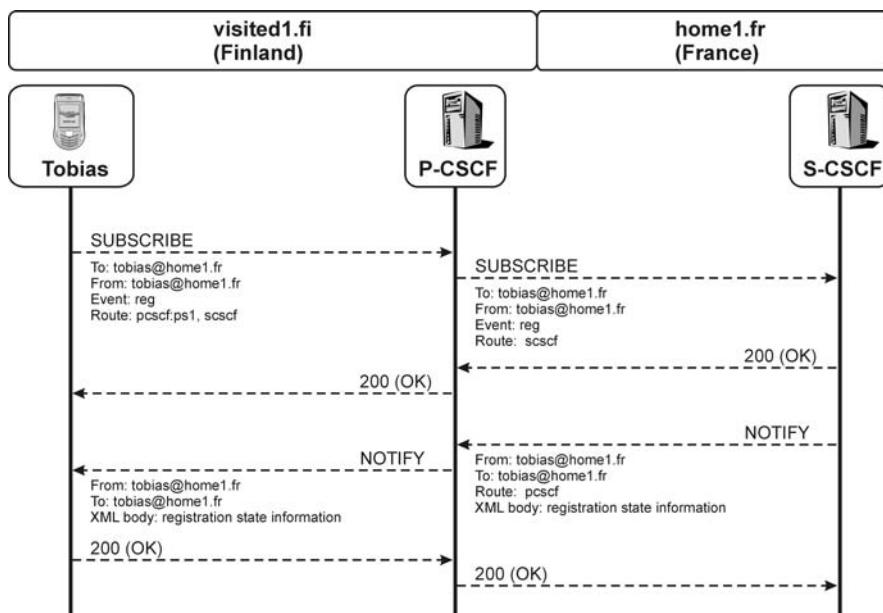


Figure 11.12 Tobias's subscription to his registration-state information

the NOTIFY request just because it did not receive a prior 200 (OK) response to the SUBSCRIBE request.

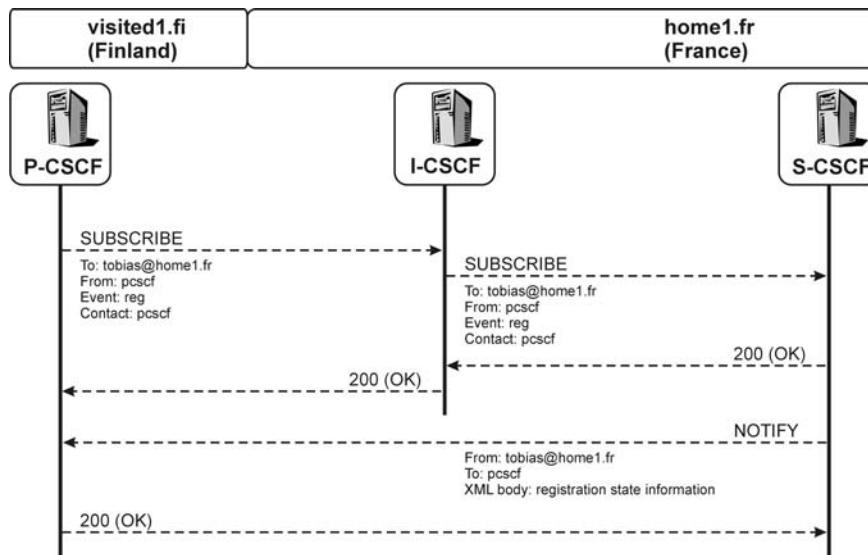
#### 11.13.7 P-CSCF's Subscription to Registration-State Information

The P-CSCF also needs to subscribe to Tobias's registration-state information and, therefore, creates a SUBSCRIBE request, which looks similar to the one that the terminal generates:

```
SUBSCRIBE sip:tobias@home1.fr SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.fi
From: <sip:pcscf1.visited1.fi>;tag=retiisi
To: "Tobias" <sip:tobias@home1.fr>
P-Asserted-Identity: <sip:pcscf1.visited1.fi>
Event: reg Expires: 600000
Accept: application/reginfo+xml
Contact: <sip:pcscf1.visited1.fi>
Content-Length: 0
```

The main difference here is that it is the P-CSCF that subscribes to the registration-state information of Tobias (Figure 11.12); therefore, it has to identify itself in the From header and the P-Asserted-Identity header. As the P-CSCF is a trusted entity (see Section 3.21.4.2) it immediately puts a P-Asserted-Identity header into the request.

As the P-CSCF did not save any routing information during the initial registration phase for its own routing purposes, it has no knowledge about the S-CSCF that was assigned for the user and, therefore, cannot include any Route headers. Consequently, it will route the request on the basis of the host part of the request URI, which is 'home1.fr' and can



**Figure 11.13** P-CSCF subscription to Tobias's registration-state information

be resolved via DNS to one or more I-CSCF addresses of Tobias's home network. The I-CSCF then queries the HSS for the address of the S-CSCF that is assigned for the URI `sip:tobias@home1.fr` and sends the request to the S-CSCF.

Note that with this SUBSCRIBE request a new dialog is created, this time between the P-CSCF and the S-CSCF. This dialog has no relation to the UE's subscription to the very same registration-state information; therefore, the S-CSCF will generate separate NOTIFY requests, including the registration-state information of Tobias, for the UE's and for the P-CSCF's subscription.

#### *11.13.8 Elements of Registration-State Information*

The S-CSCF generates a NOTIFY with Tobias's registration-state information immediately after a new subscription has been received and whenever the registration-state information changes (e.g., when a new public user identity becomes registered).

In this section we only look at the NOTIFY request and registration-state information that is received by Tobias's terminal immediately after the subscription. This information is identical to the information received by the P-CSCF at more or less the same time.

The NOTIFY request as received by Tobias's UE includes – among others – the following headers:

```
NOTIFY sip:[5555::1:2:3:4]:1357;comp=sigcomp SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.fr;branch=nosctb
Via: SIP/2.0/UDP pcscf1.visited1.fi:7531;branch=nopctb
From: "Tobias" <sip:tobias@home1.fr>;tag=peruna
To: "Tobias" <sip:tobias@home1.fr>;tag=sipuli>
Subscription-State: active;expires=599999
Event: reg
Content-Type: application/reginfo+xml
Contact: <sip:scscf1.home1.fr>
Content-Length: ( ... )
```

The things to note about this NOTIFY request are that:

- The To and From headers changed as this request was sent from the Notifier (S-CSCF) to the Subscriber (Tobias's UE). Although both headers have nearly identical content, their tags are different. The S-CSCF also has added a 'To' tag ('peruna'), which now appears in the From header;
- A Subscription-State header has been added, which indicates that the subscription is active and will expire after 599 999 seconds.

#### *11.13.9 Registration-State Information in the Body of the NOTIFY Request*

The registration-state information for the URIs associated with Tobias is included in the body of the NOTIFY request and shown in detail in Section 11.13.9. Registration-state information is a hierarchical list that consists of:

- The root element ‘reginfo’, which includes registration-state information that is associated with one user;
- One or more ‘registration’ sub-elements of the ‘reginfo’ root element. A ‘registration’ sub-element includes information about exactly one URI (i.e., one public user identity);
- Zero or more ‘contact’ sub-elements of every ‘registration’ sub-element. A ‘contact’ sub-element includes information about an address that has been registered (or de-registered) for the URI in the ‘registration’ sub-element;
- One ‘uri’ sub-elements of every ‘contact’ sub-element, indicating the contact address that was registered;
- Zeroe or one ‘display-name’ sub-elements of every ‘contact’ sub-element, indicating the display name that was set in the Contact header for the relevant contact address;
- Zero or one ‘gr:pub-gruu’ and zero or one ‘gr:temp-gruu’ sub-elements of every ‘contact’ sub-element, indicating the public and temporary GRUUs which got assigned to the device in during registration (see Section 11.13.5);
- Zero or more ‘unknown-parameter’ sub-elements of every ‘contact’ sub-element. An ‘unknown-parameter’ sub-elements includes further parameters which were present in the Contact header during registration (such feature tags).

Each registration sub-element can include the following attributes:

- The Address Of Record (AOR) attribute, which is followed by the URI for the public user identity;
- The ID attribute, which uniquely identifies the registration sub-element from among all the others;
- The state attribute of the registration sub-element, which indicates whether the indicated URI is in one of the following states:
  - ‘active’ (i.e., registered);
  - ‘terminated’ (i.e., de-registered);
  - ‘init’ (i.e., in the process of being registered, such as when an initial REGISTER request has been received, but authentication procedures have not yet been finished).

Each contact sub-element includes the registered contact address and can include the following attributes:

- The ID attribute, which uniquely identifies the contact sub-element from among all the others;
- The state attribute of the contact sub-element, which indicates whether the indicated contact – in relation to the URI of the registration sub-element – is in one of the following states:
  - ‘active’ (i.e., the URI is registered with this contact information);
  - ‘terminated’ (i.e., the binding between the URI and this contact information has just been removed).
- The event attribute of the contact sub-element, which indicates the event that caused the latest change in the contact state attribute. The events can be:

- registered – this event switches the contact address from the ‘init’ state to the ‘active’ state and indicates that the AOR has been explicitly registered (i.e., a valid REGISTER request has been received for this AOR and the related contact information is bound to it);
  - created – this event has the same meaning as the registered event, but indicates that the AOR has been implicitly registered (i.e., the binding was created automatically, such as when there is a received REGISTER request for another AOR);
  - refreshed – this event occurs when re-registration for an AOR takes place and may also occur implicitly (i.e., when re-registration for an associated AOR is performed);
  - shortened – this event occurs when the network shortens the expiry time of an AOR (e.g., to trigger network-initiated re-authentication, see Section 11.14.2);
  - deactivated – this event occurs when the binding is removed by the network (e.g., due to a network-initiated de-registration), allowing the user to perform a new initial registration attempt afterwards;
  - probation – with this event the network can de-register the user and request them to send a new initial registration after a certain time (dependent on the retry-after value);
  - unregistered – this event occurs when the user has explicitly unregistered the contact; and
  - rejected – this event occurs when the network does not allow the user to register the specific contact.
- Additional attributes, such as:
    - the expires attribute – which indicates the remaining expiration time of the registration for the specific contact address (it must be set for the shortened event, but is optionally set for other events);
    - the duration-registered attribute – which shows how long the contact is already registered;
    - the callid attribute – which indicates the value of the Call-ID header within the REGISTER request by which the contact address was registered;
    - the cseq attribute – which indicates the numerical value of the CSeq header within the REGISTER request by which the contact address was registered;
    - the retry-after attribute – which is only set for the probation event and indicates how long the UE should wait before it can try again to register.

#### 11.13.10 Example Registration-State Information

Tobias’s registration-state information is included in the body of the NOTIFY requests that the S-CSCF sends out to the UE and the P-CSCF. It includes first of all an XML document heading:

```
<?xml version="1.0"?>
<reginfo xmlns="urn:ietf:params:xml:ns:reginfo" version="0"
state="full">
```

The heading indicates the XML version in use (1.0). The registration information always starts with the root element, named ‘reginfo’, which includes a number of attributes:

- The xmlns attribute points to the Uniform Resource Name (URN) that defines the XML document and the XML namespace;
- The version attribute always starts with the value ‘0’ and is incremented by one every time a new (updated) version of the registration-state information is sent to the same recipient;
- The state attribute indicates that the following registration-state information is a full list of all the AORs that relate to Tobias. The first version (‘0’) of a reginfo document always needs to be sent as a complete list (‘full’) – subsequent information (starting from ‘1’) can be sent as ‘partial’ and will only include information that has changed since the last notification.

All the public user identities that relate to Tobias and their registration states are now listed in the document:

```

<registration aor="sip:tobias@home1.fr" id="a1" state="active">
<contact id="15" state="active" event="registered" duration-
registered="1" expires=599999
callid="apb03a0s09dkjdfglkj49111" cseq="25">

<uri>sip:[5555::1:2:3:4]:1357</uri>

<display-name>Mobile Phone - Tobias</display-name>
<unknown-paramter name="audio"/>

<unknown-paramter name="video"/>

<unknown-paramter name="mobility">

<unknown-paramter name="methods">
"INVITE, BYE, ACK, OPTIONS, CANCEL, NOTIFY, MESSAGE, PRACK, U
PDATE"
</unknown-paramter>

<unknown-paramter name="g.3gpp.icsi_ref">
"urn%3Aurn-xxx%3A3gpp-service-ims.icis.mmtel,
urn%3Aurn-xxx%3Aother-vendor-service-ims.icsi.ongame"
</unknown-paramter>

<unknown-paramter name="g.3gpp.iari_ref">
"urn%3Aurn-xxx%3Aother-app-ims.iari.firefighter"
</unknown-parameter>

<unknown-paramter="+sip-instance">
"&lt;urn:uuid: jklhzzqw7as9asfd&gt;" 
</unknown-paramter>

<gr:pub-gruu uri="sip:tobias@home1.fr
;gr=urn:uuid: jklhzzqw7as9asfd"/>

```

```

<gr:temp-gruu uri="sip:98hzah4pmmn@scscf1.home1.net
;gr"/>

</contact>
</registration>
```

The first AOR or URI is *sip:tobias@home1.fr*, which we already know from the above example. It is currently registered (state = ‘active’). The content of this registration sub-element is one contact sub-element, which shows the binding that was created by the S-CSCF between *sip:tobias@home1.fr* and the contact information *sip[55551234]* (‘uri’ sub-element). The event attribute is set to ‘registered’, which indicates that this AOR was explicitly registered with this contact address. In addition to this we see several other attributes, which give further information about the registered contact, such as e.g. the duration-registered attribute, which shows how long the contact has already been registered.

All further information and parameters explicitly registered with the contact are also shown here, i.e.:

- the display-name that was set by Tobias’s phone in order to easily identify the phone amongst all other ongoing registrations;
- the registered callee capabilities for audio, video, mobility and methods, as described in Section 11.9.2 in the unknown-parameter sub-element;
- the IMS Communication Service Identifications (ICSIIs) and IMS Application Reference Identification (IARI), as described in Section 11.9.3 in the unknown-parameter sub-element;
- the instance-id from the client, as described in Section 11.13.5 in the unknown-parameter sub-element;
- the public and temporary GRUUs that got assigned to the device as described in Section 11.13.5 in the gr:pub-gruu and gr:temp-gruu elements.

```

<registration aor="sip:+44123456789@home1.fr" id="a2"
    state="active">
    <contact id="16" state="active" event="created"
    duration-registered="1" expires=599999
    callid="apb03a0s09dkjdfglkj49111" cseq="25">
        <uri>sip:[5555::1:2:3:4]:1357</uri>

        <display-name>Mobile Phone - Tobias</display-name>

        <unknown-paramter name="audio"/>

        <unknown-paramter name="video"/>

        <unknown-paramter name="mobility">

        <unknown-paramter name="methods">
```

```

    "INVITE, BYE, ACK, OPTIONS, CANCEL, NOTIFY, MESSAGE, PRACK, U
    PDATE"
    </unknown-paramter>

    <unknown-paramter name="g.3gpp.icsi_ref">
    "urn%3Aurn-xxx%3A3gpp-service-ims.icis.mmtel,
    urn%3Aurn-xxx%3Aother-vendor-service-ims.icsi.ongame"
    </unknown-paramter>

    <unknown-paramter name="g.3gpp.iari_ref">
    "urn%3Aurn-xxx%3Aother-app-ims.iari.firefighter"
    </unknown-parameter>

    <unknown-paramter="+sip-instance">
    "&lt;urn:uuid: 12jklhzzqw7as9asfd&gt;"
    </unknown-paramter>

    <gr:pub-gruu uri="sip:+44123456789@home1.fr
    ;gr=urn:uuid:12jklhzzqw7as9asfd"/>

    <gr:temp-gruu uri="sip:55uwztagahna@scscf1.home1.net
    ;gr"/>
</contact>
</registration>
```

The next AOR is another SIP URI that was implicitly registered (event = ‘created’) with the same IP address as the first AOR. This implicit registration was made by the S-CSCF, based on the user profile of Tobias. Due to the implicit registration, all the callee capabilities (i.e. feature tags, including ICSI and IARI values) of the explicitly registered public user identity apply as well to all the implicitly registered user identities and therefore they are here reproduced as unknown-parameters.

Also a public and a temporary GRUU are assigned to the URI, but they are not the same as for the explicitly registered public user identity. The public GRUU adds the instance-id to the SIP URI of the registration element and the temporary GRUU has a different value. Separate GRUU values will be assigned for each implicitly registered public user identity.

For better readability of this section, the display-name, unknown-parameter and GRUU related sub-elements as well as some of the attributes will not be shown further in this example.

```
<registration aor="tel:+44123456789" id="a3" state="active">
    <contact id="17" state="active" event="created"
    duration-registered="1" expires=599999
    callid="apb03a0s09dkjdfglkj49111" cseq="25">
        <uri>sip:[5555::1:2:3:4]:1357</uri>
    </contact>
</registration>
```

The next AOR is a tel URL that was implicitly registered (event = ‘created’) with the same IP address as the first AOR. This implicit registration was made by the S-CSCF, based on the user profile of Tobias.

```
<registration aor="sip:tobi@home1.fr" id="b1" state="terminated">
</registration>
<registration aor="tel:+44123456111" id="b2" state="terminated">
</registration>
```

These two AORs are currently not registered (state = ‘terminated’) and, therefore, the registration sub-elements do not include any information at all. Finally, Tobias is also the game master of an online role-playing game. He takes his job in this game very seriously and is, therefore, always registered from a gaming console that has the address *sip:[5555::101:102:103:104]:1458*. The contact of the gaming console was explicitly registered (event<sub>4</sub> ‘registered’). Nevertheless, Tobias also wants to stay informed about the ongoing status of the game when he is online with his IMS UE; therefore, this AOR was also implicitly registered (event = ‘created’) by the S-CSCF when the REGISTER request for *sip:tobias@home1.fr* was received:

```
<registration aor="sip:gameMaster@home1.fr" id="c1" state="active">
    <contact id="45" state="active" event="registered">
        <uri>sip:[5555::101:102:103:104]:1458</uri>
    </contact>
    <contact id="19" state="active" event="created">
        <uri>sip:[5555::1:2:3:4]:1357</uri>
    </contact>
</registration>
</reginfo>
```

The last line of the registration-state information shows the tag that ends the XML document.

#### 11.13.11 Multiple Terminals and Registration-State Information

One or more public user identities can be registered from different terminals (i.e., different UEs). In our example it could be that Tobias also owns a simple paging device that uses his public user identity *sip:tobias@home1.fr*. This device would also need to perform registration procedures before being able to use IMS services. The registration of this device could take place over a different P-CSCF, but would end up in the same S-CSCF as the first registration. The S-CSCF then will perform the explicit as well as the implicit registrations. As the registered address belongs to a registration set of three different public user identities, all three of them are registered with the new contact.

After this paging device has registered, Tobias’s UE and his P-CSCF would receive another NOTIFY message indicating that additional contact information for the public user identity is now available: that is, information that relates to the first AOR in the body of the NOTIFY would include the following information:

```

<?xml version="1.0"?>
<reginfo xmlns="urn:ietf:params:xml:ns:reginfo" version="1"
state="partial"contact id="20" state="active" event="registered">
        <uri>sip:[5555::171:171:172:173]:1579</uri>
    </contact>
</registration>
<registration aor="sip:+44123456789@home1.fr" id="a2">
    <contact id="16" state="active" event="created">
        <uri>sip:[5555::1:2:3:4]:1357</uri>
    </contact>
    <contact id="21" state="active" event="created">
        <uri>sip:[5555::171:171:172:173]:1579</uri>
    </contact>
</registration>
<registration aor="tel:+44123456789" id="a3" state="active">
    <contact id="17" state="active" event="created">
        <uri>sip:[5555::1:2:3:4]:1357</uri>
    </contact>
    <contact id="22" state="active" event="created">
        <uri>sip:[5555::171:171:172:173]:1579</uri>
    </contact>
</registration>
</reginfo>

```

Always the second lot of contact information in the registration information relates to the paging device. Note that this is now the second lot of registration-state information that the UE receives. To make sure that no registration-state information was lost, the ‘version’ parameter is set to ‘1’ (the first lot of information had version = ‘0’). While the first NOTIFY included complete registration-state information, the UE will receive only information about changed registration elements, in this case for the AOR *sip:tobias@home1.fr*. Consequently, the state parameter is set to ‘partial’ (in the first lot of information it was set to ‘full’).

### 11.13.12 Related Standards

Specifications relevant to Section 11.13 are:

- 3GPP TS 23.003 Numbering, addressing and identification.
- RFC3265 Session Initiation Protocol (SIP)-specific Event Notification.
- RFC3325 Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks. An example flow is shown in Figure 11.18.
- RFC3455 Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the 3rd-Generation Partnership Project (3GPP).
- RFC3680 A Session Initiation Protocol (SIP) Event Package for Registrations.

## 11.14 Re-Registration and Re-Authentication

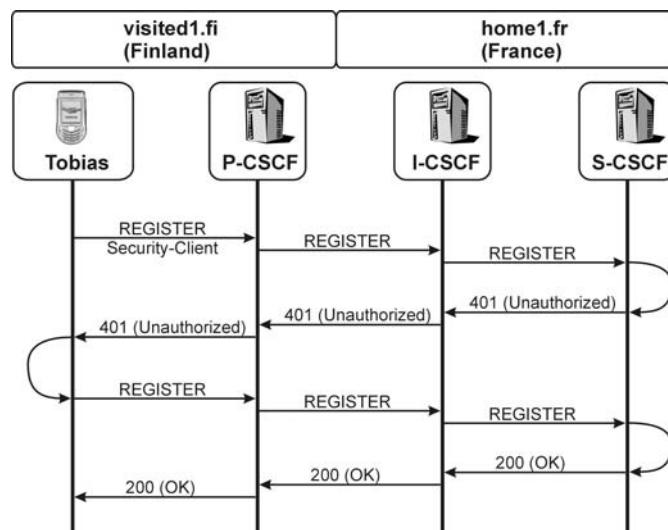
### 11.14.1 User-initiated Re-registration

Tobias's UE can at any time perform re-registration by sending a new REGISTER request (see Section 11.5) to the network (Figure 11.13). This happens when, say, the registration needs to be refreshed due to expiry of the registration time getting close. As the re-registration is handled in the same way as an initial SIP registration procedure, this is not further described here.

### 11.14.2 Network-Initiated Re-Authentication

The IMS UE registers its contact information for a time of 600 000 seconds, which means that the binding of the registered public user identities and the physical IP address is kept for around seven days in the S-CSCF. As user authentication procedures are directly coupled to registration procedures, this would mean that the S-CSCF has no means of re-authenticating the user within this time period. Certain conditions may, nevertheless, make it necessary for the S-CSCF to re-authenticate the UE.

To achieve this, the S-CSCF can reduce the expiry time of the user's registration. Let us assume that Tobias has already been registered for three hours and his home operator wants to perform a random re-authentication. The S-CSCF assigned to Tobias will reduce the expiry time of Tobias's registration to 600 seconds (exactly 10 minutes). Up to that moment Tobias's UE is unaware of the reduced registration time and would, therefore, not perform a re-registration, which is needed for re-authentication. To inform the UE about this, the S-CSCF makes use of the UE's subscription to the registration-state event package.



**Figure 11.14** User-initiated re-registration (without re-authentication)

The S-CSCF generates a NOTIFY request for the registration-state event package, in which it indicates that it shortened the registration time and sends this NOTIFY request to Tobias's UE. On receiving this request the UE will immediately update the registration expiry time information.

Furthermore, all other subscribers to Tobias's registration-state information (e.g., the P-CSCF and the subscribed ASs) will receive a NOTIFY request from the S-CSCF with the updated state information.

After half the indicated time has elapsed (i.e., 300 seconds), the UE will send out a REGISTER request. From then on, the normal registration procedures as described in Section 11.5 will take place, during which the S-CSCF can authenticate the user again (see Section 11.6).

#### *11.14.3 Network-Initiated Re-Authentication Notification*

The NOTIFY message (Figure 11.14) that is sent from the S-CSCF to the UE will include the following information:

```
<?xml version="1.0"?>
<reginfo xmlns="urn:ietf:params:xml:ns:reginfo" version="2"
state="partial"event="shortened"
expires="600"event="shortened"
expires="600"event="shortened"
expires="600"event="shortened"
expires="600"

```

All registration and related contact states are still set to 'active', but the latest event that occurred for contact is indicated as 'shortened'. The expires value shows that there are

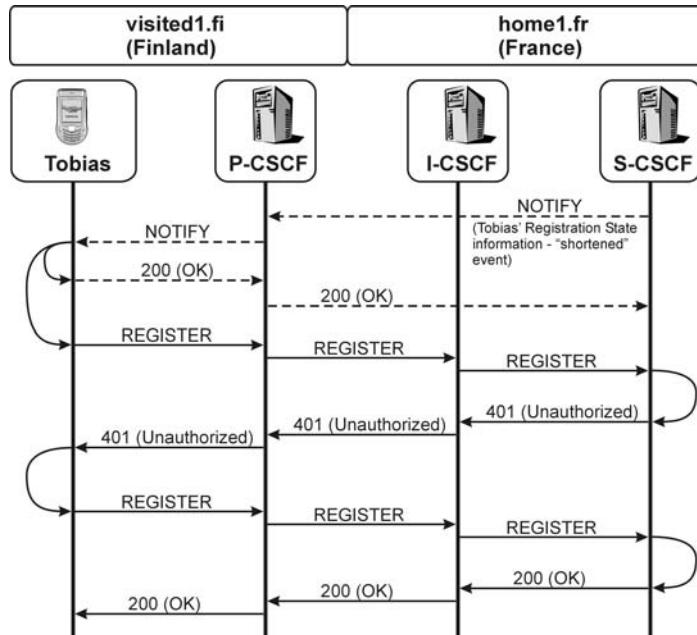


Figure 11.15 Network-initiated re-authentication

10 minutes left for the UE to re-register. In this document only partial registration-state information is delivered (state = ‘partial’ in the document header), as the rest of the registration-state information has not been changed.

As the S-CSCF sends partial information, the contact elements that were not shortened are not shown in this registration-state information.

#### 11.14.4 Related Standards

Specifications relevant to Section 11.13 are:

RFC3265 Session Initiation Protocol (SIP)-specific Event Notification.

RFC3680 A Session Initiation Protocol (SIP) Event Package for Registrations.

## 11.15 De-Registration

### 11.15.1 Overview

All things come to an end at some point, and this is also true for the registration of a user to the IMS. Tobias might want to be undisturbed after he called his sister and switches off his mobile phone. When doing so, his phone sends another REGISTER request to the S-CSCF, including all the information we have already seen, but indicating that this time it is for de-registration (Figure 11.15). The S-CSCF will then clear all the information it has stored for Tobias, update the data in the HSS and send a 200 (OK) response to Tobias’s UE.

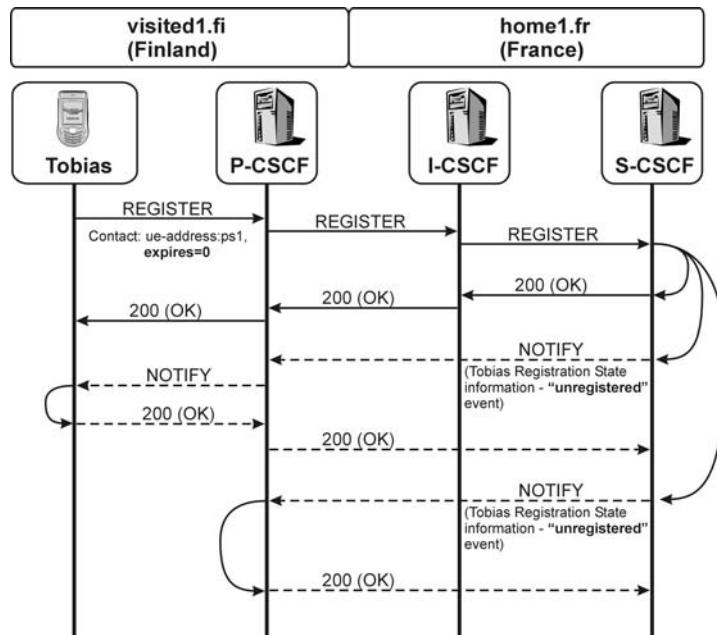


Figure 11.16 User-initiated de-registration

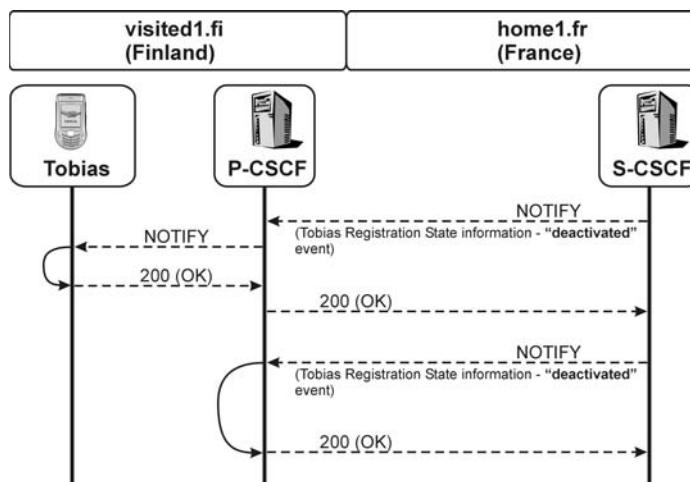


Figure 11.17 Network-initiated de-registration

Sometimes the network needs to de-register the user (Figure 11.16), e.g. the S-CSCF needs to be shut down for administrative reasons (see Section 11.15.3.2). In these cases the S-CSCF would simply send another NOTIFY message with registration-state information to Tobias's UE, this time indicating that he has been de-registered.

If the registration time expires and Tobias did not send any further REGISTER request (see Section 11.15.3.1), the registration is terminated and the NOTIFY requests are sent to all subscribers to the registration-state event information of Tobias, but not to the de-registered UE, as the binding has expired and therefore no messages can be sent towards it.

In both cases the S-CSCF will send NOTIFY requests to the P-CSCF and all other subscribers to Tobias's registration-state information, indicating that Tobias has been de-registered. By sending these NOTIFY requests the dialogs that were created during subscription to the registration-state event will also be terminated.

### 11.15.2 User-Initiated De-Registration

If Tobias decides to switch off his phone, the UE will send a REGISTER request to the network in order to de-register:

```
REGISTER sip:home1.fr SIP/2.0
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;comp=sigcomp;branch=99uetb
Route: sip:[5555::a:b:c:d]:7531;comp=sigcomp;lr
Max-Forwards: 70
From: <sip:tobias@home1.fr>;tag=ulkomaa
To: <sip:tobias@home1.fr>;tag=kotimaa
Authorization: Digest username="user1_private@home1.fr",
realm="home1.fr",
nonce=A34Cm+Fva37UYWpGNB34JP, algorithm=AKAv1-MD5,
uri="sip:home1.fr",
response="6629fae49393a05397450978507c4ef1",
integrity-protected="yes"
uri="sip:home1.fr",
Require: sec-agree
Proxy-Require: sec-agree
Security-Verify: tls ;q=0.2, IPsec-3gpp ;q=0.1 ;alg=hmac-sha-1-96
;spi-c=98765434 ;spi-s=87654322
;port-c=8644 ;port-s=7533
Security-Client: digest, IPsec-3gpp ;alg=hmac-sha-1-96
;spi-c=23456790 ;spi-s=12345679
;port-c=2472 ;port-s=1357
Contact: "Mobile Phone - Tobias" <sip:[5555::1:2:3:4]:1357;
comp=sigcomp>
;expires=0
Call-ID: apb03a0s09dkjdfglkj49222
CSeq: 49 REGISTER
Content-Length: 0
```

This is principally the same information that we have already seen in the other REGISTER requests; the main difference is that the expires value is set to 0, which means that the

user wants to de-register the binding between the public user identity (in the To header) and the IP address (in the Contact header).

This REGISTER request will be routed in exactly the same way as every other REGISTER request (i.e., it will not follow the stored Service-Route). Therefore, it will:

- traverse the P-CSCF – which checks for integrity protection and adds the integrity-protected = yes flag to the Authorization header;
- traverse the I-CSCF – which will ask the HSS for the S-CSCF address that was selected for the user; and
- finally, be received at the S-CSCF – where de-registration will take place.

The S-CSCF performs the Diameter Deregistration Notification procedures over the Cx interface by sending a Diameter Server-Assignment Request (SAR) to the HSS, including:

- the ‘R’ command flag set to ‘1’, indicating that this is a Diameter request;
- the Command-Code set to ‘301’, indicating the Diameter ‘Server Assignment’ command;
- the ‘User-Name’ AVP (1) set to the private identity of Tobias, indicated in the username field within the Authorization header of the SIP REGISTER request (see Section 11.6.3), i.e. to `tobias_private@home1.fr`, if the Authorization header was present in the REGISTER request;
- the ‘Server-Assignment-Type’ AVP (614) set to the value ‘USER\_DEREGISTRATION’ (5) as the deregistration is invoked by the user;
- the ‘Public-Identity’, ‘Server-Name’, ‘Origin-Host’, ‘Origin-Realm’, ‘Destination-Realm’ and ‘Destination-Host’ AVPs in the same way as described for the SAR in Section 11.5.6.

The HSS deletes the name of the S-CSCF from the registration data related to Tobias and sends back a Diameter Server Assignment Answer (SAA) to the S-CSCF, containing:

- the ‘R’ command flag set to ‘0’, indicating that this is a Diameter answer;
- the Command-Code set to ‘301’, indicating the Diameter ‘Server Assignment’ command;
- the ‘User-Name’, ‘Result-Code’, ‘User-Data’, ‘Charging-Information’ and ‘Associated-Identities’ AVPs in the same way as described for the SAM in Section 11.5.6. Note that the user profile and the charging function addresses are sent again in this answer message to the S-CSCF.

The S-CSCF sends back a 200 (OK) response to the UE, which also includes the expires header set to the value 0.

Afterwards, the S-CSCF will generate NOTIFY requests to all subscribers of the registration-state information of Tobias, including Tobias’s UE. Each of these NOTIFY requests will include the Subscription-State header set to the value ‘terminated’, which indicates that the subscription to the registration-state information of that user has been terminated. For example:

```
 NOTIFY sip:[5555::1:2:3:4]:1357;comp=sigcomp SIP/2.0
Subscription-State: terminated
```

The body of these NOTIFY requests will include Tobias's registration-state information:

```
<?xml version="1.0"?>
<reginfo xmlns="urn:ietf:params:xml:ns:reginfo" version="3"
state="partial"

```

Once again, this XML document includes a 'partial'-state notification, as it does not explicitly list those public user identities that have not been registered (see Section 11.13.6):

```
<registration aor="sip:tobias@home1.fr" id="a1" state="active"state="terminated" event="unregistered">
        <uri>sip:[5555::1:2:3:4]:1357</uri>
    </contact>
    <contact id="20" state="active" event="registered">
        <uri>sip:[5555::171:171:172:173]:1579</uri>
    </contact>
</registration>

<registration aor="sip:+44123456789@home1.fr" id="a2"
state="active"state="terminated" event="unregistered">
        <uri>sip:[5555::1:2:3:4]:1357</uri>
    </contact>
    <contact id="21" state="active" event="created">
        <uri>sip:[5555::171:171:172:173]:1579</uri>
    </contact>
</registration>

<registration aor="tel:+44123456789" id="a3" state="active"state="terminated" event="unregistered">
        <uri>sip:[5555::1:2:3:4]:1357</uri>
    </contact>
    <contact id="22" state="active" event="created">
        <uri>sip:[5555::171:171:172:173]:1579</uri>
    </contact>
</registration>
```

The public user identity *sip:tobias@home1.fr* and the public user identities of the same registration set are still active, as they are registered with Tobias's pager (see Section 11.13.10). Only the contact address of the mobile phone was set to terminated.

```
<registration aor="sip:gameMaster@home1.fr" id="c1" state="active"state="terminated" event="unregistered">
```

```

<uri>sip:[5555::1:2:3:4]:1357</uri>
</contact>
</registration>
</reginfo>
```

Finally, the gaming URI *sip:gameMaster@home1.fr* also remains registered, as another UE is still actively using it. Only the contact that was explicitly de-registered ended up being removed.

### 11.15.3 Network-Initiated De-Registration

#### 11.15.3.1 Cx communication Upon Registration Timeout

If the registration times out locally at the S-CSCF, i.e. Tobias did not send any further REGISTER message to refresh the registration expiration time, and if this is the only registration that Tobias has currently active (i.e. he is not registered from any other terminal) the S-CSCF and HSS perform the Diameter Deregistration Notification procedures over the Cx interface in the same way as described in Section 11.15.2, with the only exception that the ‘Server-Assignment-Type’ AVP (614) in the SAR from the S-CSCF to the HSS is set to:

- either ‘TIMEOUT\_DEREGISTRATION’ (4) if the S-CSCF will delete Tobias’s user profile – the HSS will then remove the S-CSCF address from Tobias’s user profile;
- or ‘TIMEOUT\_DEREGISTRATION\_STORE\_SERVER\_NAME’ (6) if the S-CSCF will keep Tobias’s user profile and will serve Tobias further – the HSS then will set Tobias’s user profile to ‘unregistered’ but will keep the S-CSCF assigned to Tobias’s user profile.

#### 11.15.3.2 Cx Communication Upon Administrative De-Registration

If administrative reasons in Tobias’s home network force Tobias to be deregistered, the HSS will inform the S-CSCF to deregister the user. The HSS therefore will perform the Diameter Network Initiated Deregistration by HSS procedures by sending a Diameter Registration-Termination Request (RTR) over the Cx Interface to the S-CSCF, containing:

- the ‘R’ command flag set to ‘1’, indicating that this is a Diameter request;
- the Command-Code set to ‘304’, indicating the Diameter ‘Registration Termination’ command;
- the ‘User-Name’ AVP (1) set to the private identity of Tobias, i.e. to ‘tobias-private@home1.fr’;
- the ‘Deregistration-Reason’ AVP (615) including:
  - the ‘Reason-Code’ AVP (616) set to the reason why the user is being de-registered, in this example we assume it is that the S-CSCF needs to be maintained and therefore needs to be removed for a while, so the code is ‘PERMANENT\_TERMINATION’ (0);

- optionally the ‘Reason-Info’ AVP (617) set to a free text string, that can be used at the S-CSCF e.g. for login purposes;
- the ‘Origin-Host’ and ‘Origin-Realm’ AVPs set to the HSS address and the operator network name in which the HSS is located (i.e. home1.fr)
- the ‘Destination-Host’ and ‘Destination-Realm’ AVPs set to the S-CSCF address (scscf1.home1.fr) and the operator network name in which the S-CSCF is located (home1.fr).

The S-CSCF will terminate the registration of Tobias and will respond to the HSS with a Diameter Registration-Termination Answer (RTA), including:

- the ‘Result-Code’ AVP (268) set to ‘DIAMETER\_SUCCESS’ (2001), indicating that the request for deregistration was successful;
- optionally the ‘Associated-Identities’ AVP (632), containing all the private user identities that are de-registered in addition to the private user identity indicated in the User-Name AVP in the RTR. This list is only indicated if Tobias is registered from several terminals and therefore has several private user identities registered at the same time.

### 11.15.3.3 Registration State Information to the User upon Network Initiated De-Registration

Whenever the network sees the need to de-register the user or some of the user’s identities, the S-CSCF will generate NOTIFY requests in the same way as described in Section 11.14.2, only the content of the XML document will look different. In this example we assume that Tobias has already de-registered from his mobile phone (Section 11.14.2) and is only active from his paging device:

```
<?xml version="1.0"?>
<reginfo xmlns="urn:ietf:params:xml:ns:reginfo" version="4"
state="partial"state="terminated" event="deactivated"state="terminated"state="terminated" f event="terminated"state="terminated"state="terminated" event="terminated"

```

```

<uri>sip:[5555::171:171:172:173]:1579</uri>
</contact>
</registration>

<registration aor="sip:gameMaster@home1.fr" id="c1"
  state="terminated">
  <contact id="45" state="terminated" event="deactivated">
    <uri>sip:[5555::101:102:103:104]:1458</uri>
  </contact>
</registration>
</reginfo>
```

All public user identities are now set to ‘terminated’, as the network consequently deregistered every registration that was active for Tobias, even those from other terminals. The event has changed to ‘de-activated’, which indicates that it was the network that de-registered, not the user.

#### *11.15.4 Related Standards*

Specifications relevant to Section 11.15 are:

RFC3265 Session Initiation Protocol (SIP)-specific Event Notification.

RFC3680 A Session Initiation Protocol (SIP) Event Package for Registrations.

## 11.16 GPRS-IMS-Bundled Authentication (GIBA)

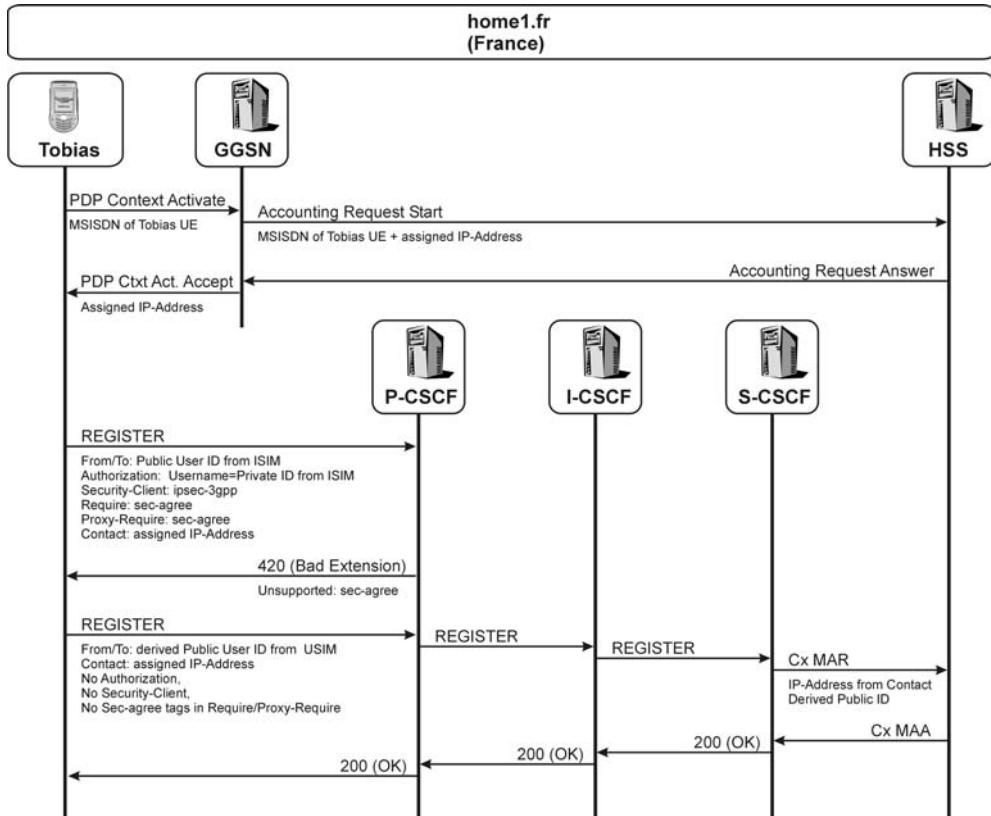
### *11.16.1 Example IMS Registration with Fallback to GIBA*

In Section 3.21.2.1 the general principles and requirements for early IMS security are described. Figure 11.17 shows how GIBA in principle when the network requires IP-based authentication and the UE supports both – i.e., 3GPP AKA as well as GIBA An example flow is shown in Figure 11.18.

When the UE establishes an IMS signalling PDP context, the GGSN creates a RADIUS ‘Accounting-Request START’ request towards the GGSN, in which it indicates the user’s Mobile Subscriber Integrated Services Digital Network (MSISDN) number (i.e., the phone number) as well as the IP address for the IMS-specific PDP context.

After establishing a signalling PDP context the UE will send out an initial REGISTER request, as described in the previous sections, including the Authorization header, a Security-Client header as well as the ‘sec-agree’ option tag in the Require and the Proxy-Require header:

```
REGISTER sip:home1.fr SIP/2.0
From: <sip:tobias@home1.fr>;tag=pohja
To: <sip:tobias@home1.fr>
Authorization: Digest username="tobias_private@home1.fr",
realm="home1.fr", nonce=" ",
```



**Figure 11.18** Example early IMS security flow

```

uri="sip:home1.fr", response=""
Security-Client: digest, IPsec-3gpp; alg=hmac-sha-1-96
    ;spi-c=23456789 ;spi-s=12345678
    ;port-c=2468; port-s=1357
Require: sec-agree
Proxy-Require: sec-agree
Contact: "Mobile Phone - Tobias" <sip:[5555::1:2:3:4]:1357>
    ;expires=600000
  
```

When the P-CSCF, which in this example only supports GIBA, receives this REGISTER request, it will reject it with a 420 (Unsupported) response, indicating that it does not support the Sip-Sec-Agree extension.

```

SIP/2.0 420 Unsupported
Unsupported: sec-agree
  
```

This rejection is used at the UE as an indication that GIBA has to be applied.

It is for this reason that the UE constructs a new REGISTER message, which does not include the Authorization and Security-Client headers nor the sec-agree option tags in the Require and Proxy-Require header.

The identities used in the REGISTER request have now to be derived from the USIM application in the UE. This means that even if the UE is equipped with an ISIM application (in the case of GIBA), it nevertheless needs to derive identities from the USIM (see Section 11.13.3). This is required as this second REGISTER request does not include the private user identity in the Authorization header. As a consequence, the user will always register a temporary public user identity in the case of GIBA.

```
REGISTER sip:33.222.IMSI.3gppnetworks.org SIP/2.0
From: <sip:222330999999999@33.222.IMSI.3gppnetworks.org>;tag=t2
To: <sip:222330999999999@33.222.IMSI.3gppnetworks.org>
Contact: "Mobile Phone - Tobias" <sip:[5555::1:2:3:4]:1357>
;expires=600000
```

This second REGISTER request will now be routed through the network in exactly the same way as described in the previous sections of this chapter.

The P-CSCF, when receiving this REGISTER request, will detect that GIBA is used, as the ‘sec-agree’ option tag is not included in the Require and Proxy-Require headers.

The S-CSCF will detect that GIBA is used, as no Authorization header is included in the received REGISTER request. It will, therefore, send the identity in the To header and the IP address from the Contact header to the HSS via the Cx interface.

The HSS checks whether the provided MSISDN is correlated with the IMSI provided in the user identity. Afterwards, it checks whether the IP address indicated by the S-CSCF was assigned by the GGSN to that specific MSISDN. As seen before, the HSS receives

**Table 11.4** GIBA registration scenarios

		Network supports		
UE supports	Only GIBA	Only 3GPP AKA	Both	
Only GIBA	GIBA	P-CSCF rejects “GIBA” REGISTER request –no IMS registration possible	UE initiates with “GIBA” REGISTER –network accepts	3GPP AKA
Only 3GPP AKA	P-CSCF rejects initial REGISTER request –no IMS registration possible	3GPP AKA		3GPP AKA
Both	P-CSCF rejects initial REGISTER request –UE sends “GIBA” REGISTER	3GPP AKA		3GPP AKA

from the HSS the IP address that was assigned to that MSISDN during PDP context establishment procedures. Only if this check succeeds will the HSS give to the S-CSCF the indication that the authentication was successful. When receiving this indication from the HSS, the S-CSCF will respond to the received REGISTER request with a 200 (OK) response. On completion of this procedure, the user is authenticated by IMS.

In order to guarantee that SIP messages sent into the IMS network were really originated by the authenticated UE, the GGSN will check the IP addresses within these messages against the registered contact address.

#### 11.16.2 GIBA Scenarios

Table 11.4 lists the possible scenarios for GIBA and 3GPP AKA. In the case when the UE only supports GIBA but the network only supports 3GPP AKA, the P-CSCF will reject the initial REGISTER request with a 421 (Extension Required) response, indicating that it requires the Sip-Sec-Agree procedures to provide full IMS security.

SIP/2.0 421 Extension Required

Require: sec-agree

As the UE in this case does not support 3GPP AKA, it will stop its attempt to register to the IMS network.

Table 11.4 shows two cases in which IMS registration is not possible. This underlines the early nature of GIBA. Only in networks that provide both mechanisms – GIBA and 3GPP AKA – will subscribers not suffer from service discontinuity.



# 12

## An Example IMS Multimedia Telephony Session

### 12.1 Overview

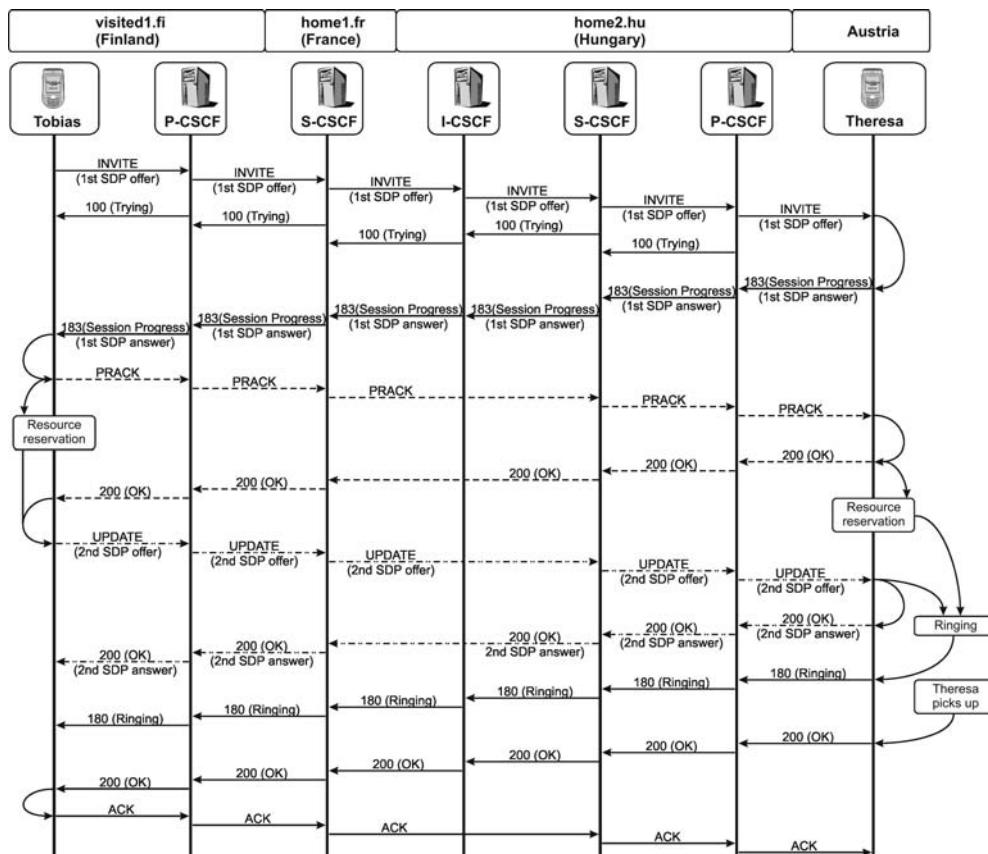
This chapter shows an example of a IMS Multimedia Telephony session between Tobias and his sister Theresa, who are both registered in their home networks and are both currently roaming in different countries (see Section 10.1).

The Session Initiation Protocol (SIP) and Session Description Protocol (SDP) are facilitated by the Internet Protocol Multimedia Subsystem (IMS) to ensure that Tobias and Theresa can talk to each other and see each other on the screens of their mobile phones. In order to achieve this within the wireless environment certain steps have to be taken:

- Tobias's UE needs to construct an INVITE request that includes a registered public user identity of Theresa in order to reach her – Section 12.2.4;
- All SIP messages must traverse the Proxy Call Session Control Functions (P-CSCFs) and the Serving-CSCFs (S-CSCFs) of both users – Section 0;
- All SIP messages are sent via the established IP security (IPsec) Security Associations (SAs) between User Equipment (UEs) and their P-CSCFs – Section 12.3.3.1;
- All SIP messages are sent compressed between UEs and their P-CSCFs – Section 12.4;
- The two UEs agree on the media streams that they will exchange. In the example case they will exchange a bidirectional audio stream, so that brother and sister can talk to each other, and a bidirectional media stream, so that they can also see each other – Section 12.5;
- The two UEs agree on a single codec for every media stream that they will exchange – Section 12.5;
- The networks will authorize the media for the session, so that the users can reserve the related resources – Section 12.6.6;

- Both UEs perform resource reservation (i.e., they set up the necessary media PDP contexts over which the media streams to and from the network will be transported) – Section 12.6;
- Theresa's UE will not get any indication that her brother is calling her before the resources for media sessions (i.e., the media PDP contexts) have been reserved at both ends, in order to be sure that media sessions can really be established – Section 12.6.4;
- The network elements will exchange charging information, so that media sessions can be billed correctly – Section 12.7;
- The S-CSCFs may initiate advanced services for their served users – Section 12.3.8;
- Theresa's UE will finally start to ring and Theresa will accept the session; this completes the session establishment phase.

After Tobias and Theresa have finished their call, they will hang up and one of their UEs will send a BYE request to the other UE – Section 12.8.1. The SIP message sequence for the example session will look like that shown in Figure 12.1.



**Figure 12.1** IMS session establishment call flow

Section 12.9 shows the set of most alternative session establishment flows, which all follow the principles for IMS session establishment, but due to resource reservation and interworking conditions vary in the SIP and SDP related signalling.

Sections 12.10 and 12.11 give examples for the specific routing of GRUUs and PSIs within the IMS.

Section 12.12 gives a short introduction to GPRS and its basic procedures, as GPRS is the access technology used within the example scenario.

## 12.2 Caller and Callee Identities

### 12.2.1 Overview

Section 11.13 described how an IMS user becomes aware during registration of the public user identities he can use and which of them are currently registered. Subsequently, the users – in our example Theresa and Tobias – will use these identities for different purposes. For every kind of dialog within the IMS – in this example the INVITE dialog – two identities are essential:

- A registered and authenticated public user identity of the calling user (Tobias) needs to be indicated in the request, in order to guarantee the identification of the user within his home network and for execution rights for extended services. This is provided in the P-Asserted-Identity header within the INVITE request;
- A registered and authenticated public user identity of the called user (Theresa) needs to be indicated in the request, in order to be able to contact the user and to execute services for her. This is provided in the request Uniform Resource Identifier (URI) of the INVITE request and in the P-Asserted-Identity header of the first response.

### 12.2.2 From and To Headers

The INVITE request that Tobias's UE sends toward Theresa includes the following headers that are related to either his or her identity:

```
INVITE sip:theresa@home2.hu SIP/2.0
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister" <sip:Theresa@sister.com>
P-Preferred-Identity: <sip:tobias@home1.fr>
Privacy: None
```

Obviously, the From and To headers can be set to any value the sender likes. We chose the wording in this example to clearly indicate that the values of these two headers in any request other than the REGISTER request have no influence on any IMS routing or security procedures – they can be freely set. The only information that is needed by the protocol itself are the tag parameters in these two headers.

Tobias's home network operator may have certain restrictions to some values that the To header can be set at. If this is the case, the home network can only reject the request if the setting of the From or To header does not fulfil the operator policy, because SIP does not allow any of these headers to be changed.

### 12.2.3 Identification of the Calling User: P-Preferred-Identity and P-Asserted-Identity

#### 12.2.3.1 Inclusion of the P-Preferred-Identity Header by the Originating UE

In the above example Tobias includes the P-Preferred-Identity header, which is optional. When present, it should include a registered public user identity of that user. In Section 11.13.6 we saw how Tobias became aware of all the public user identities that he can use. By means of the registration-state information to which his terminal subscribed, he also discovered which of these user identities he currently has registered.

If Tobias wanted to completely hide his identity from his sister, he would have needed to set the Privacy header to the ‘id’ value. This value would force Theresa’s P-CSCF to remove the P-Asserted-Identity header from the INVITE request, so that Theresa could only see the identity in the From header as the caller identification.

#### 12.2.3.2 Originating P-CSCF Includes the P-Asserted-Identity Header

Tobias’s UE will send out the INVITE request which is then first received by the P-CSCF. The P-CSCF checks whether the request was received over a valid IPsec SA. If the request was received unprotected (i.e., not over an SA), the P-CSCF will reject the request.

Afterwards, the P-CSCF inserts a P-Asserted-Identity header in the INVITE request, which replaces the received P-Preferred-Identity header, if one was received. The P-Asserted-Identity header is the only identity within an IMS dialog that is guaranteed to include a registered and authenticated public user identity of the user.

If a P-Preferred-Identity header is present, the P-CSCF will check whether the URI in the header is a currently registered public user identity of the user who sent in the request. It will discover whether it is a registered public user identity from the registration-state information it is subscribed to (see Section 11.13.7). The P-CSCF can ensure that a certain request was sent in by a specific user based on the SA it was received over (see Section 11.7). If both checks are successful, the P-CSCF will replace the P-Preferred-Identity header with a P-Asserted-Identity header that includes the same content.

If the P-Preferred-Identity header did not include a currently registered public user identity, then the P-CSCF will remove the header. In this case or when no P-Preferred-Identity header was received at all, the P-CSCF will add a P-Asserted-Identity header that includes the default public user identity of the user. How the default public user identity of the user is determined is described in Section 11.13.4:

```
INVITE sip:theresa@home2.hu SIP/2.0
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister" <sip:Theresa@sister.com>
P-Asserted-Identity: <sip:tobias@home1.fr>
Privacy: None
```

#### 12.2.3.3 Originating S-CSCF and P-Asserted-Identity Header

On receiving this INVITE request the S-CSCF of Tobias’s home network operator will identify Tobias by the information given in the P-Asserted-Identity header. The S-CSCF

will also check the authentication and registration state of the public user identity indicated in the header. Because of these checks, the header serves as the main identification of the user for the whole dialog. Application Server (ASs) – see Section 12.3.8 – base the identification and even the authentication of the user on this header as well.

Tobias's S-CSCF may add an additional URI to the P-Asserted-Identity header. In this example it adds the telephone Universal Resource Locator (tel URL) of Tobias to the header:

```
INVITE sip:theresa@home2.hu SIP/2.0
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
      To: "My beloved Sister" <sip:Theresa@sister.com>
P-Asserted-Identity: <sip:tobias@home1.fr>, <tel:+33123456789>
Privacy: None
```

Before the S-CSCF of Tobias's home network routes the request toward Theresa's home network, it will check whether that network is within its trust domain (see Section 3.21.4.2). If the S-CSCF and the home network of Theresa do not share the same trust domain, the S-CSCF will remove the P-Asserted-Identity header from the request, as long as the Privacy header is set to the value 'id'. For this example we assume that the two networks have a trust relationship that allows the header to be forwarded.

#### 12.2.3.4 P-Asserted-Identity Header at the Terminating End

The P-CSCF of Theresa has to check the value of the Privacy header of the request. As it is not set to the value 'id' it can send the P-Asserted-Identity header to Theresa's UE.

So, finally, the UE of Theresa receives the P-Asserted-Identity header. It can facilitate the information in the header by, say, displaying the 'real name' of Theresa's caller.

#### 12.2.4 Identification of the Called User

##### 12.2.4.1 The Request URI

Let us look again at the INVITE message that Tobias sends. Its first line, the request URI, looks like:

```
INVITE sip:theresa@home2.hu SIP/2.0
```

The request URI is set to the final destination of the request (i.e., to Theresa's SIP URI). Section 0 explains how this URI is used for SIP and IMS routing procedures. But, this URI also identifies Theresa as the called user within her home network. This means that Theresa's S-CSCF will check whether this public user identity is currently registered and authenticated. If Theresa is currently not registered with this public user identity, the S-CSCF will return, say, a 404 (Not Found) response to the INVITE request and the call will fail or, based on the filter criteria for an unregistered user, will forward the INVITE to Theresa's voice mail box.

For our example we assume that Theresa has registered the public user identity that Tobias's UE put into the request URI.

#### 12.2.4.2 The Request URI and P-Called-Party-ID Header

Another problem arises when this request is sent by Theresa's S-CSCF toward the terminating P-CSCF. The S-CSCF, which also acts as Theresa's SIP registrar, will re-write the request URI with the registered contact address of Theresa, in order to route the request to the UE at which Theresa is currently registered. Therefore, the public user identity in the request URI will be lost.

However, Theresa might have several public user identities and might want to know under which of them she receives a call: for example, she might have work-related user identities and others that relate to her private life. Maybe her UE even provides different ring tones for each of her user identities.

We already saw in Section 12.2.2 that Theresa cannot trust the To header in the request, as the originator can set it to any value – one that might be completely different from the public user identity in the request URI.

In order not to lose information about the public user identity that is used by Tobias to call his sister, the S-CSCF, when re-writing the request URI with the registered contact address, will add a P-Called-Party-ID header to the INVITE request. This P-Called-Party-ID header includes the public user identity that was received in the request URI:

```
INVITE sip:[5555::5:6:7:8]:1006 SIP/2.0
P-Called-Party-ID: sip:theresa@home2.hu
```

#### 12.2.4.3 P-Asserted-Identity Header

After receiving the INVITE request, Theresa's UE will send back a P-Preferred-Identity header in the first response to the INVITE request – the 183 (Session in Progress) response – which will include one of Theresa's public user identities:

```
SIP/2.0 183 Session in Progress
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister" <sip:Theresa@sister.com>;tag=schwester
P-Preferred-Identity: <sip:theresa@home2.hu>
Privacy: None
```

The P-CSCF of Theresa will perform the same checks as described before for Tobias's P-CSCF (see Section 12.3.3.2) and will replace it by a P-Asserted-Identity header:

```
SIP/2.0 183 Session in Progress
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister" <sip:Theresa@sister.com>;tag=schwester
P-Asserted-Identity: <sip:theresa@home2.hu>
Privacy: None
```

### 12.2.5 Related Standards

Specifications relevant to Section 12.2 are:

- RFC3323 A Privacy Mechanism for the Session Initiation Protocol (SIP).
- RFC3325 Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks.
- RFC3455 Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the 3rd-Generation Partnership Project (3GPP).

## 12.3 Routing

### 12.3.1 Overview

One of the most complex issues within IMS is the routing of requests, especially the routing of initial requests. In our example, Tobias is sending the initial INVITE request to Theresa. Consequently, a SIP dialog is created within which several subsequent requests – such as ACK, PRACK, UPDATE and BYE – are sent.

Tobias's UE is unaware at the time of sending the INVITE request how Theresa's UE can be reached. All it can provide is:

- The final destination of the INVITE request – which is the SIP URI of Theresa (one of her public user identities) that Tobias had to indicate (e.g., by selecting it from his phone book);
- The address of the P-CSCF – which is the outbound proxy of Tobias's UE and will be the first hop to route to. This address is obtained before SIP registration during the P-CSCF discovery procedures (see Section 11.4);
- The address of the S-CSCF – which was discovered during registration procedures by means of the Service-Route header (see Section 11.5.8).

Armed with this partial route information the INVITE request is sent on its way. It first traverses the P-CSCF and then the S-CSCF that have been selected for Tobias.

Tobias's S-CSCF now has no further routing information available for the request other than the final destination (i.e., the public user identity of Theresa, `sip:theresa@home2.hu`). As Tobias's S-CSCF does not act as a registrar for Theresa, it can only resolve the host part of the address: `home2.hu`. This domain name is sent to the Domain Name System (DNS) server and the S-CSCF will receive back one or more Interrogating-CSCF (I-CSCF) addresses of Theresa's home network, will select one of them and will send the INVITE request to it. Detailed examples on how to resolve addresses via DNS are given in Section 11.4 and 13.3.6.2.

The I-CSCF just acts as the entry point to Theresa's home network. It asks the local HSS for the address of the S-CSCF that was selected for Theresa and sends the INVITE further on to the returned address.

Theresa's S-CSCF now acts as the registrar and replaces her SIP URI with the contact address that she has registered. It does not send the request directly to Theresa's UE, because it has not established an SA with it (see Section 11.7). The INVITE request, therefore, is first sent to Theresa's P-CSCF. The S-CSCF knows the address of the P-CSCF, as that was received within the Path header during Theresa's registration (see Section 11.5.9). The P-CSCF finally forwards the INVITE request to Theresa's UE over the IPsec SA.

This shows that for the initial request the route from Tobias to Theresa is put together piece by piece, as the originating UE and the CSCFs have only information about the next one or two hops that have to be traversed. In order to make further routing within the dialog easier, SIP routing mechanisms will be used:

- All CSCFs put their addresses on top of the Via header – this allows all responses to the INVITE request to be sent back over exactly the same route as the request;
- All CSCFs, other than Theresa's I-CSCF, put their addresses on top of the Record-Route header – this allows all subsequent requests in the dialog to be sent over the CSCFs that put themselves in the Record-Route header. The I-CSCF in Theresa's home network fulfilled its routing task when it discovered the addresses of Theresa's S-CSCF; so, it is no longer needed on the route.

When sending out subsequent requests the UEs will include a list of Route entries, which will force the request to follow the recorded route (Figure 12.2). Routing issues related to the provision of services are explained in Section 12.3.8.

### *12.3.2 Session, Dialog, Transactions and Branch*

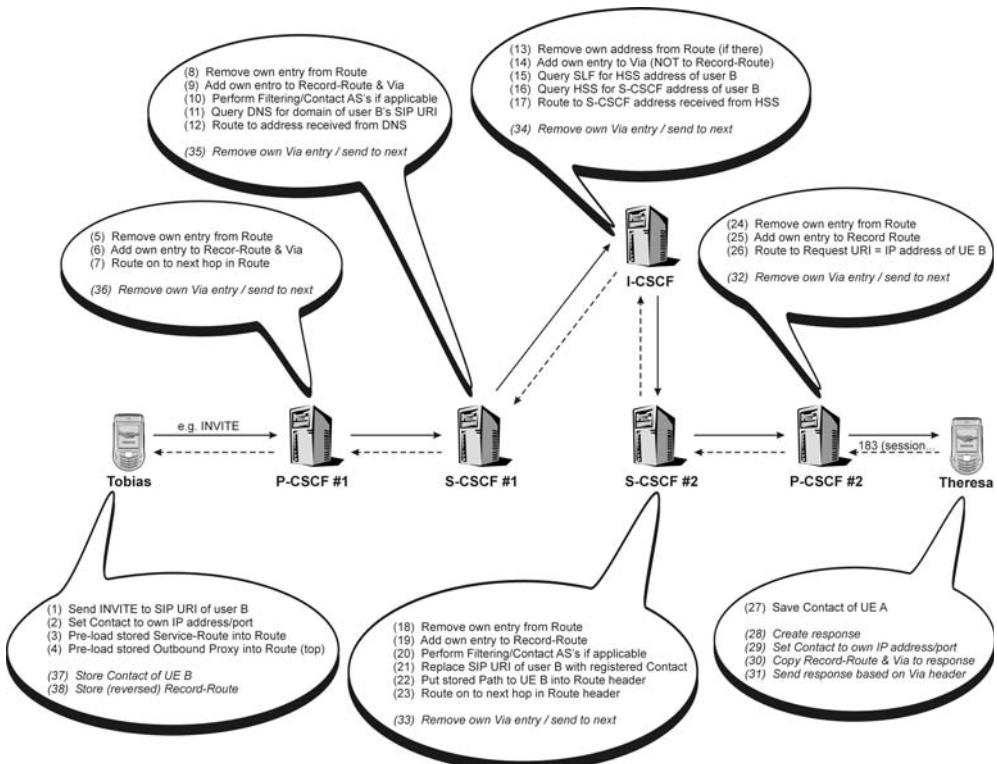
During session establishment and while the session is active, different types of signalling messages are exchanged and different kinds of relations between the two UEs are established.

The term ‘session’ describes the media connections between the two users. Tobias wants to exchange audio and video media streams with his sister. This exchange of media is done on the so-called ‘bearer level’: this means that Real-time Transport Protocol (RTP) packets are sent from the two items of UE to their Gateway GPRS Support Nodes (GGSNs) and the GGSNs exchange these packets between each other directly over the backbone. This session is established on the basis of SIP and SDP signalling that are exchanged via the ‘control plane’.

A SIP dialog is the signalling relation between the two UEs which is needed to establish, modify and release the multimedia session. The dialog will be first established (with the INVITE request) and will exist as long as the related session is active. Every SIP dialog is identified by the value of the Call-ID header and by the tags in the To and the From headers of the SIP requests, which in our example look like:

```
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister" <sip:theresa@home2.hu>;tag=schwester
Call-ID: apb03a0s09dkjdfglkj49555
```

The SIP dialog for the multimedia session between Tobias and Theresa starts with the INVITE request and ends with the 200 (OK) response for the BYE request.



**Figure 12.2** Routing an initial INVITE request and its responses

A SIP transaction comprises a single SIP request and all the responses related to it. In order to establish the session, Tobias's UE sends an INVITE request to Theresa's UE. At the very outset it receives a 100 (Trying) response from the P-CSCF in response to the request. Afterwards, Theresa's UE responds with a 183 (Session in Progress), a 180 (Ringing) and, finally, a 200 (OK) response. All these five messages belong to the same dialog and have the same CSeq number:

```

From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister" <sip:theresa@home2.hu>;tag=schwester
Call-ID: apb03a0s09dkjdfglkj49555
CSeq: 1112 INVITE

```

Every subsequent request sent from the same end (in this case from Tobias's UE) will have a higher CSeq number than the preceding request: this means that, for example, the first PRACK request includes CSeq 1113, the following UPDATE request CSeq 1114 and so forth.

Every entity – either UE or CSCF – will correlate the responses that are received for a sent request on the basis of the branch parameter that it added as a parameter to its Via

header entry: for example, the P-CSCF of Tobias adds the following Via header to the INVITE request:

```
INVITE sip:theresa@home2.hu SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=9pctb
```

The branch parameter identifies the INVITE transaction (i.e., the INVITE request and the responses to it) at the P-CSCF. It is constructed from the tags in the To and From headers, the Call-ID, the CSeq number and the information in the topmost Via header of the request.

### 12.3.3 Routing of the INVITE Request

#### 12.3.3.1 From Tobias's UE to the P-CSCF

Tobias's UE will include the following routing-related headers in the initial INVITE request:

```
INVITE sip:theresa@home2.hu SIP/2.0
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=8uetb
Route: <sip:[5555::a:b:c:d]:7531;lr>
Route: <sip:orig@scscf1.home1.fr;lr>
Contact: <sip:[5555::1:2:3:4]:1357>
```

The destination of the request is Theresa's SIP URI, which is indicated in the request URI.

During registration the route between Tobias's UE and its S-CSCF in the home network was discovered by the Service-Route header (see Section 11.5.8). The UE pre-loads this first part of the route into the Route header and places the P-CSCF above it, because it always needs to contact its outbound proxy first.

Tobias's UE puts its IP address in the Contact header of the request, so that the remote UE B can directly reach it. It also adds its IP address to the Via header in order to receive responses to that request. Note that this Contact header can include a variety of additional parameters, such as:

- callee capabilities that are supported by the UE, e.g. video and audio (see Section 11.9);
- IMS Communication Service Identifiers (ICSIIs) and IMS Application Reference Identifiers (IARIs) that are supported by the UE, e.g. the MMtel-ICSI (see Section 11.9);
- a public or temporary GRUU assigned to the phone (see Section 11.13.5);
- support for compression (see Section 12.4).

Here is an example of how the Contact header could look like in the INVITE request:

```
Contact: <sip:[5555::1:2:3:4]:1357;comp=sigcomp>;
          audio;video;mobility
          ;methods=
          "INVITE,BYE,ACK,OPTIONS,CANCEL,NOTIFY,MESSAGE,PRACK,UPDATE"
          ;g.3gpp.icsi_ref="urn%3Aurn-xxx%3A3gpp-service-ims".
```

```
icis.mmtel"
;pub-gruu="sip:tobias@home1.fr;gr=urn:uuid:jk1hzzqw7as9asfd
```

For the sake of readability, these additional parameters within the Contact header will not be indicated further on in this example. They will be treated in the related sections as indicated above.

As the request is sent over established IPsec SAs (see Section 11.7), Tobias's UE puts:

- The protected server port of the UE (1357) as the port value in the Contact header, because it wants to receive all subsequent requests within this dialog via the established IPsec SA;
- The protected server port of the UE (1357) as the port value in the Via header, because it wants to receive all responses to the INVITE request via the established IPsec SA;
- The protected server port of the P-CSCF (7531) as the port value of the address of the P-CSCF in the Route header, because the P-CSCF must receive all requests from the UE via an established IPsec SA. The UE became aware of the P-CSCF's protected server port during SIP Security Mechanism Agreement procedures (see Sections 11.7.5 and 11.8).

The To and From headers are never used for routing purposes (see Section 12.2.2).

The INVITE request is now sent to the topmost entry in the Route header, which is the P-CSCF that serves Tobias.

### 12.3.3.2 From Tobias's P-CSCF to the S-CSCF

When receiving this request the P-CSCF:

- removes its own entry from the topmost Route header;
- checks that the request includes further routing information in accordance with the routing information it saved during registration (i.e., that the UE does not try to deviate from the Service-Route);
- puts its address at the top of the Via header, as it needs to receive all responses to the requests;
- adds the first Record-Route header and puts its own address there – this guarantees that all subsequent requests within this dialog will traverse the P-CSCF;
- does not include the protected server port number in both the Via and the Record-Route entry – the protected server port number identifies only the port over which the P-CSCF wants to receive SIP messages that are sent from the UE over the set of established IPsec SAs.

On completion of this, the P-CSCF again routes toward the topmost entry of the Route header, which in this case is the S-CSCF that serves Tobias:

```
INVITE sip:theresa@home2.hu SIP/2.0
Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=9pctb
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=8uetb
```

```
Record-Route: <sip:pcscf1.visited1.fi;lr>
Route: <sip:orig@scscf1.home1.fr;lr>
Contact: <sip:[5555::1:2:3:4]:1357>
```

### 12.3.3.3 From Tobias's S-CSCF to Theresa's Home Network (I-CSCF)

Tobias's S-CSCF removes its entry from the topmost Route header, which afterwards is empty and can be removed. It then adds its address at the top of the Record-Route and Via headers.

Afterwards, the S-CSCF will perform the procedures for service provisioning that are described in Section 12.3.8.

Having done this, it needs to route the request further. But, now there is a problem: there is no Route header left to point to the next hop. All the S-CSCF can do now is take the host part of the address of Theresa's public user identity that is indicated in the request URI (i.e., home2.hu) and resolve a SIP server in that domain from the DNS (see Section 15.1). In return, it receives one or more addresses of I-CSCFs that are located in the home network of Theresa. It takes one of them and sends the request there.

Note that the S-CSCF can only put the address of the I-CSCF into a Route header when it is aware that this I-CSCF is able to act as a loose router. In the example case the S-CSCF and the I-CSCF are in different networks and it is not assumed that the S-CSCF knows about the routing capabilities of the I-CSCF. Therefore, it sends the UDP packet that transports the initial INVITE to the I-CSCF address:

```
INVITE sip:theresa@home2.hu SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.fr;branch=asctb
Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=9pctb
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=8uetb
Record-Route: <sip:scscf1.home1.fr;lr>
Record-Route: <sip:pcscf1.visited1.fi;lr>
Contact: <sip:[5555::1:2:3:4]:1357>
```

### 12.3.3.4 From the I-CSCF to Theresa's S-CSCF

The I-CSCF in Theresa's home network now needs to discover the address of the S-CSCF that is allocated to her. Even if Theresa were not currently registered, the I-CSCF may well be able to discover the address of a default S-CSCF as long as Theresa is subscribed to some services as an unregistered user. Information about the S-CSCF currently allocated to a user is stored in the Home Subscriber Server (HSS). There could be several HSSs within the network. If so, the I-CSCF would need to first have to query the Subscription Locator Function (SLF) to discover which HSS holds the data for Theresa (an example on how this is done can be found in Section 13.3.4). In this example we assume, for simplification, that there is only one HSS available in the network and that the address of that HSS is configured at the I-CSCF.

The I-CSCF performs a User Location Query by sending a Diameter Location Info Request (LIR) via the Cx interface to the HSS, containing:

- the 'R' command flag set to '1', indicating that this is a Diameter request;

- the Command-Code set to ‘302’, indicating the Diameter ‘Location Info’ command;
- the ‘Public-Identity’ AVP (601) set to the SIP URL indicated in the SIP INVITE request URI, i.e. to `sip:theresa@home2.hu`, which is Theresas public user identity;
- the ‘Origin-Host’ AVP (264) set the address of the querying I-CSCF, i.e. ‘`icscf1.home2.hu`’;
- the ‘Origin-Realm’ AVP (296) set to the domain name of the operator network in which the I-CSCF is located, i.e. ‘`home2.hu`’;
- the ‘Destination-Realm’ AVP (283) set to the home domain of the SLF/HSS, i.e. ‘`home2.hu`’, as this is the domain within which the user location information is queried;
- the ‘Destination-Host’ AVP (293) set to the address of the HSS, which needs to be locally configured at the I-CSCF, as no SLF query has taken place;
- the ‘Originating-Request’ AVP (633) in order to indicate that the request is an originating SIP request.

The HSS determines, that the SIP URI `sip:theresa@home2.hu` belongs to Theresa and the S-CSCF address stored for Theresa’s registrations is `scscf2.home2.hu`, i.e. the S-CSCF at which Theresa is currently registered. The HSS returns a Diameter Location Info Answer (LIA) to the I-CSCF, containing:

- the ‘R’ command flag set to ‘0’, indicating that this is a Diameter answer;
- the Command-Code set to ‘302’, indicating the Diameter ‘Location Info’ command;
- the ‘Result-Code’ AVP (268) set to ‘DIAMETER\_SUCCESS’ (2001), indicating that the query was successful;
- the ‘Server-Name’ AVP (602) set to the address of Theresas S-CSCF, i.e. `scscf2.home2.hu`;

The I-CSCF now adds a Route entry at the top of the Route list and indicating the S-CSCF address, as received in the ‘Server-Name’ AVP in the LIA. Furthermore, the I-CSCF:

- removes its entry from the topmost Route header, if one is present (in this example this is not the case);
- puts its address at the top of the Via list, in order to receive all responses for the INVITE request;
- does not put its address in the Record-Route, because it does not need to receive any subsequent requests in this dialog – the task of the I-CSCF is to find the S-CSCF of the called user, and, as this is done during initial request processing, there is no need for it to stay in the Route header.

The request once again goes toward the topmost entry in the Route header, which this time is set to Theresa’s S-CSCF:

```
INVITE sip:theresa@home2.hu SIP/2.0
Via: SIP/2.0/UDP icscf1.home2.hu;branch=bictb
Via: SIP/2.0/UDP scscf1.home1.fr;branch=asctb
Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=9pctb
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=8uetb
```

```

Route: <sip:scscf2.home2.hu;lr>
Record-Route: <sip:scscf1.home1.fr;lr>
Record-Route: <sip:pcscf1.visited1.fi;lr>
Contact: <sip:[5555::1:2:3:4]:1357>

```

### 12.3.3.5 From Theresa's S-CSCF to the P-CSCF

Now Theresa's S-CSCF – her registrar – receives the INVITE request. Once again, it removes its entry from the Route header and puts itself in the Via and the Record-Route lists. Afterwards, it provides the services for Theresa as described in Section 12.3.8.

Having done so, the S-CSCF performs the actions of a registrar (i.e., it replaces the request URI, which is still set to Theresa's SIP URI, by her registered contact address). The registered contact address also includes the protected server port (1006) that is used to send requests from the P-CSCF to Theresa's UE via the established IPsec SA.

During Theresa's registration the S-CSCF received the Path header from the P-CSCF. It must now put the entries of the Path header in the Route header of the INVITE request. Were this not done, the request would immediately be sent to Theresa's UE, which could not accept the request as it had not established an IPsec SA with the S-CSCF.

The S-CSCF adds a new Route header, puts the P-CSCF address in it and, as this is now the topmost entry, sends the request to this address immediately:

```

INVITE sip:[5555::5:6:7:8]:1006 SIP/2.0
Via: SIP/2.0/UDP scscf2.home2.hu;branch=cscth
Via: SIP/2.0/UDP icscf1.home2.hu;branch=bicth
Via: SIP/2.0/UDP scscf1.home1.fr;branch=asctb
Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=9pctb
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=8uetb
Route: <sip:pcscf2.home2.hu;lr>
Record-Route: <sip:scscf2.home2.hu;lr>
Record-Route: <sip:scscf1.home1.fr;lr>
Record-Route: <sip:pcscf1.visited1.fi;lr>
Contact: <sip:[5555::1:2:3:4]:1357>

```

### 12.3.3.6 From the P-CSCF to Theresa's UE

The P-CSCF receives the request and does the usual: it removes the whole Route header, adds itself to the Record-Route and Via headers and then sends the request to the final destination indicated in the request URI – Theresa's UE:

```

INVITE sip:[5555::5:6:7:8]:1006 SIP/2.0
Via: SIP/2.0/UDP pcscf2.home2.hu:1511;branch=dpcth
Via: SIP/2.0/UDP scscf2.home2.hu;branch=cscth
Via: SIP/2.0/UDP icscf1.home2.hu;branch=bicth
Via: SIP/2.0/UDP scscf1.home1.fr;branch=asctb
Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=9pctb

```

```
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=8uetb
Route: <sip:pcscf2.home2.hu:1511;lr>
Record-Route: <sip:scscf2.home2.hu;lr>
Record-Route: <sip:scscf1.home1.fr;lr>
Record-Route: <sip:pcscf1.visited1.fi;lr>
Contact: <sip:[5555::1:2:3:4]:1357>
```

The entry of the P-CSCF in the Via header also includes the port number of the protected server port (1511), which was negotiated with Theresa's UE during the registration procedure in the same way as described for Tobias's registration in Section 11.7.5. This entry forces Theresa's UE to send all responses to this request over the established IPsec SA.

The selfsame protected server port value (1511) is put in the Record-Route header entry of the P-CSCF, where it expects to receive all subsequent requests from Theresa's UE that are sent in this dialog.

After Theresa's UE has received the INVITE request, it stores the received Contact value and the Record-Route header list, as it will route subsequent requests in the dialog based on them.

#### 12.3.4 Routing of the First Response

##### 12.3.4.1 From Theresa's UE to the P-CSCF

Theresa's UE now creates a response to the received INVITE request, which is, due to the usage of preconditions (see Section 12.6.4), a 183 (Session in Progress) response.

The UE puts its own IP address in the Contact header to indicate the address it wants to use to receive subsequent requests in this dialog. The contact address also includes the protected server port of Theresa's UE (1006), which guarantees that all subsequent requests will be received via the established IPsec SA as well.

The Record-Route and Via headers of the INVITE request also go into the response. After doing so, Theresa's UE sends the response to the address and port number of the topmost entry in the Via header, which is the protected server port of the P-CSCF:

```
SIP/2.0 183 Session in Progress
Via: SIP/2.0/UDP pcscf2.home2.hu:1511;branch=dpcth
Via: SIP/2.0/UDP scscf2.home2.hu;branch=cscth
Via: SIP/2.0/UDP icscf1.home2.hu;branch=bicth
Via: SIP/2.0/UDP scscf1.home1.fr;branch=asctb
Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=9pctb
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=8uetb
Record-Route: <sip:pcscf2.home2.hu:1511;lr>
Record-Route: <sip:scscf2.home2.hu;lr>
Record-Route: <sip:scscf1.home1.fr;lr>
Record-Route: <sip:pcscf1.visited1.fi;lr>
Contact: <sip:[5555::5:6:7:8]:1006>
```

All other responses that are sent from Theresa's UE to this INVITE request will include the same Via header entries as the 183 (Session in Progress) response.

### 12.3.4.2 From Theresa's P-CSCF Onward to Tobias's P-CSCF

The P-CSCF identifies the INVITE transaction the response belongs to by the branch parameter that it set in its own entry in the Via header. It then manipulates the routing information in the 183 (Session in Progress) response in the following way:

- it removes its own address from the the Via header;
- it re-writes its own Record-Route entry;
- it sends the request to the topmost entry in the Via header, which is the S-CSCF in Theresa's home network.

Why does the P-CSCF re-write its own Record-Route entry? Well, it does this to ensure that no other entity than Theresa's UE sends messages to the P-CSCF's protected server port that is used for the IPsec SA with the UE. If Theresa's S-CSCF were to send the next request (the PRACK) to the P-CSCF's protected server port (1511), the request would be dropped by the IPsec layer in the P-CSCF's protocol stack, as it had not been sent integrity-protected via the IPsec SA:

```
SIP/2.0 183 Session in Progress
Via: SIP/2.0/UDP scscf2.home2.hu;branch=cscth
Via: SIP/2.0/UDP icscf1.home2.hu;branch=bicth
Via: SIP/2.0/UDP scscf1.home1.fr;branch=asctb
Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=9pctb
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=8uetb
Record-Route: <sip:pcscf2.home2.hu;lr>
Record-Route: <sip:scscf2.home2.hu;lr>
Record-Route: <sip:scscf1.home1.fr;lr>
Record-Route: <sip:pcscf1.visited1.fi;lr>
Contact: <sip:[5555::5:6:7:8]:1006>
```

From then on, nothing of consequence happens to the response until it reaches Tobias's P-CSCF – every hop simply removes its own Via entry and sends the message toward the next entry in the Via. The Record-Route stays untouched. Note that other servers on the way back are permitted to re-write their Record-Route entries in order to distinguish requests received from different directions; however, this is not shown in this example, as it is an implementation option for a CSCF to carry out.

As said already in Section 12.3.3.1, the Contact header indicated here can include an additional set of parameters, which are not further treated in this section.

### 12.3.4.3 From Tobias's P-CSCF to his UE

When receiving the 183 (Session in Progress) response, Tobias's P-CSCF performs similar actions to Theresa's P-CSCF. It also re-writes its entry in the Record-Route header; but, instead of removing the protected server port value in its entry (as Theresa's P-CSCF did during the handling of the same response), it adds its own protected server port value (7531). Consequently, it forces Tobias's UE to send all subsequent requests via the established IPsec SA.

As the P-CSCF routes the response on the basis of the Via header, it will send it to the protected server port of Tobias's UE (1357) – i.e., via the IPsec SA:

```
SIP/2.0 183 Session in Progress
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=8uetb
Record-Route: <sip:pcscf2.home2.hu;lr>
Record-Route: <sip:scscf2.home2.hu;lr>
Record-Route: <sip:scscf1.home1.fr;lr>
Record-Route: <sip:pcscf1.visited1.fi:7531;lr>
Contact: <sip:[5555::5:6:7:8]:1006>
```

After receiving the response, Tobias's UE:

- stores the IP address of Theresa's UE, as received in the Contact header; and
- stores the Record-Route list after reversing the order of all entries in it.

### *12.3.5 Re-transmission of the INVITE Request and the 100 (Trying) Response*

After having sent out the INVITE request, Tobias's UE waits for responses from Theresa's UE. It will wait until its timer T1 – in IMS this is set to the value of two seconds – expires. Afterwards, it will re-transmit the INVITE request repeatedly until either a response to the request is received or until 128 (= 64 \* T1) seconds have elapsed; it will then indicate to Tobias that establishment of the session has failed.

As the INVITE request has to pass through several CSCFs all over Europe, it might take longer than two seconds for it to reach Theresa's UE, which has to construct the 183 (Session in Progress) response before once again travelling back to Finland.

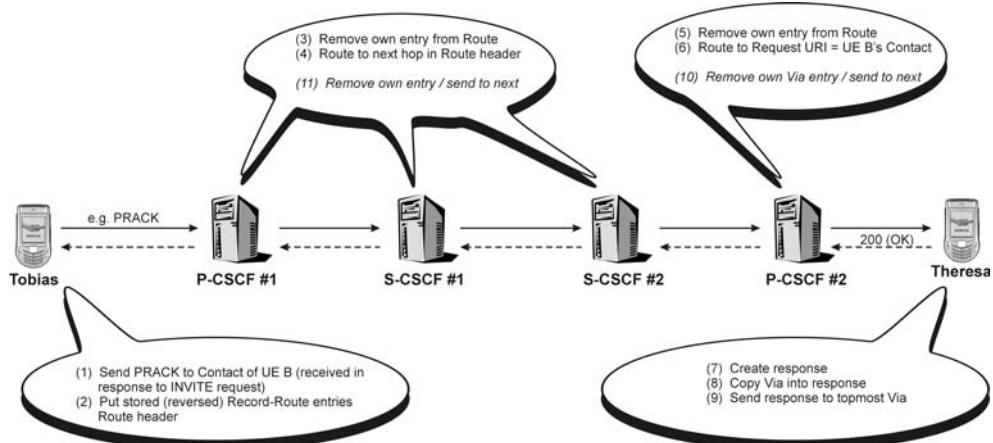
To avoid frequent re-transmissions of the INVITE request from Tobias's UE, the P-CSCF sends back a 100 (Trying) response after it has received the INVITE request. This indicates that from now on the P-CSCF will take care of such re-transmissions.

The same is done by all other call-statefull SIP proxies on the route (see Figure 12.1). The 100 (Trying) is always stopped at the SIP proxy that was the latest to take over responsibility for re-transmission. For example, the S-CSCF of Theresa's home network sends back the 100 (Trying) response, which first reaches the I-CSCF. As the I-CSCF is not a call-statefull SIP proxy it just sends it on (based on the Via header). Next it reaches the S-CSCF of Tobias's home network. Tobias's S-CSCF has sent the 100 (Trying) response to the P-CSCF; consequently, it took over responsibility for the re-transmission of the INVITE request. Now the receipt of the 100 (Trying) response indicates that it no longer needs to re-transmit the INVITE request, as this responsibility is taken over by Theresa's S-CSCF.

### *12.3.6 Routing of Subsequent Requests in a Dialog*

When one of the two UEs needs to send a subsequent request within a dialog, it copies the stored Record-Route entries into the Route header of the new requests and the remote UE's IP address into the request URI.

The request then is routed toward the remote UE by strictly following the entries in the Route header (Figure 12.3). Every CSCF that is traversed puts itself in the Via header, in order to get all the responses to this request.



**Figure 12.3** Routing of subsequent requests and their responses

As the I-CSCF did not record any route in the beginning, it does not receive any subsequent request. For example, Tobias's UE has to send back a PRACK request to acknowledge the received 183 (Session in Progress) response (see Section 12.5.2). This PRACK request would include the following routing-related information:

```
PRACK sip:[5555::5:6:7:8]:1006 SIP/2.0
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=82uetb
Route: <sip:pcscf1.visited1.fi:7531;lr>
Route: <sip:scscf1.home1.fr;lr>
Route: <sip:scscf2.home2.hu;lr>
Route: <sip:pcscf2.home2.hu;lr>
```

The PRACK request, therefore, will be routed:

- on the basis of the Route headers to Tobias's P-CSCF and his S-CSCF as well as Theresa's S-CSCF and her P-CSCF; and
- from Theresa's P-CSCF based on the address in the request URI, which Tobias's UE took from the received Contact header that was received in the 183 (Session in Progress) response to Theresa's UE over the IPsec SA.

A subsequent request within a dialog does not include a Contact header, as the addresses of the two UEs were already exchanged during the sending and receiving of the initial request and its first response. Furthermore, the CSCFs will not put any Record-Route headers in the request, because the route was already recorded during the initial request.

Theresa's UE will send back a 200 (OK) response to this PRACK request and will include the following routing information:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf2.home2.hu;branch=c2scth
```

```
Via: SIP/2.0/UDP scscf1.home1.fr;branch=a2sctb
Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=92pctb
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=82uetb
```

This response will be routed back on the basis of the Via header entries. Record-Route headers are no longer returned.

### 12.3.7 Standalone Transactions from One UE to Another

For standalone transactions, such as MESSAGE or OPTIONS, the same routing procedures as those used for an initial request are performed, although record routing does not need to be done, because a standalone transaction does not create a dialog.

### 12.3.8 Routing to and from ASs

#### 12.3.8.1 Filter Criteria Evaluation in the S-CSCF

Service provisioning in the IMS is achieved by ASs, which are contacted on the basis of initial filter criteria. When Tobias's or Theresa's S-CSCF receives an initial request, they will go through these filter criteria one by one and, if one or more of them matches, they will send the request toward the indicated AS. Filter criteria are downloaded by the S-CSCF from the HSS during registration and are part of Tobias's and Theresa's service profile; this is further described in Section 3.12.

In this example we assume that there are three ASs that have set filter criteria for requests that originate from Tobias (see Table 12.1). Tobias's S-CSCF will check these filter criteria one by one against the information received in the INVITE request:

```
INVITE sip:theresa@home2.hu SIP/2.0
Via: SIP/2.0/UDP scscf1.home1.fr;branch=asctb
Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=9pctb
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=8uetb
Route: <sip:orig@scscf1.home1.fr;lr>
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister" <sip:Theresa@sister.com>
P-Asserted-Identity: <sip:tobias@home1.fr>
Privacy: None
```

**Table 12.1** Filter criteria in Tobias's S-CSCF

Element of filter criteria	Filter criteria 1	Filter criteria 2	Filter criteria 3
SPT: session case	Originating	Originating	Terminating
SPT: public user identity	<i>tel:+44-123-456-789</i>	<i>sip:tobias@home1.fr</i> <i>tel:+44-123-456-789</i>	<i>sip:tobias@home1.fr</i>
SPT: SIP method	*	INVITE	SUBSCRIBE
Further SPT	-	-	SIP header: event: pres
Application server	<i>sip:as1.home1.fr;lr</i>	<i>sip:tas.home1.fr;lr</i>	<i>sip:as3.home1.fr;lr</i>

The asterisk signifies that any value is going to match.

Filter criterion #1 does not match, because the P-Asserted-Identity header, which is checked against the Service Point Trigger (SPT) for the public user identity, does not include Tobias's tel URL.

Filter criterion #2 does match, because:

- the INVITE request is received from the originating user – the S-CSCF knows this from the user part it set in its Service-Route header entry (see Section 11.5.8) and which is now returned in the Route header;
- the P-Asserted-Identity is set to one of the public user identities that are filtered (`sip:tobias@home1.fr`);
- the SIP method is INVITE.

### 12.3.8.2 From the S-CSCF to the Telephony Application Server (TAS)

Consequently, the S-CSCF now has to send the INVITE request to the AS (Figure 12.4), which in this case is the telephony application server (TAS) that is indicated in filter criterion #2. It also needs to take care that it receives the request again after the TAS has fulfilled its actions, because the S-CSCF needs to evaluate filter criterion #3 and to send the request toward the home network of Theresa. To achieve this, the S-CSCF adds a set of routing-related headers by putting:

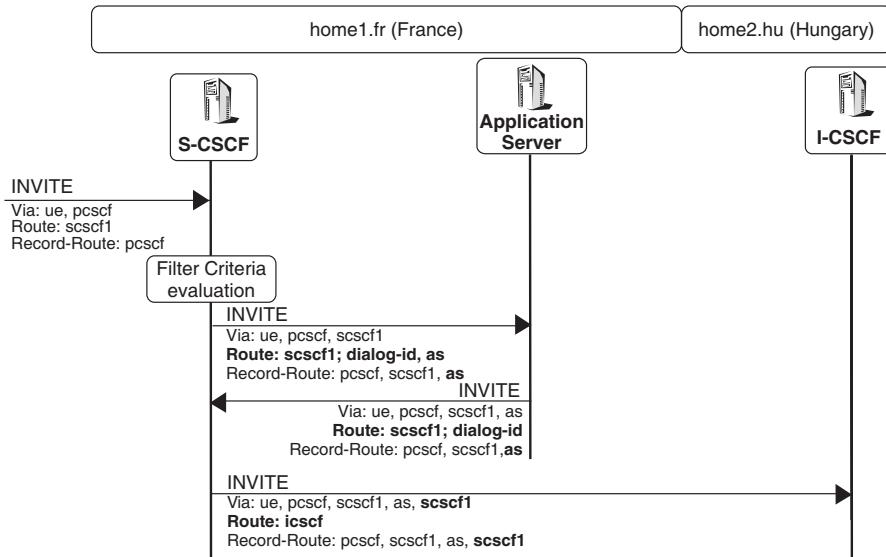
- its own address at the top of the Route headers, in order to receive the INVITE request back from the AS;
- the address of the AS at the top of the Route headers, in order to route the INVITE request to the AS as the next hop;.
- its own address on top of the Record-Route headers, so that it stays on the route for subsequent requests as well;
- its own address on top of the Via headers, so that it receives all responses to the request.

In addition to this, the S-CSCF will add an implementation-specific dialog identifier to its own Route header entry, which it has just added. It sets this dialog identifier to a value that allows it to identify the dialog that is created with this INVITE. What is the purpose of this?

The AS could decide to act as a Back-to-Back User Agent (B2BUA) and terminate the INVITE request locally. It would then send a new INVITE request with a new Call-ID toward the S-CSCF. As this AS would use the URI that is included in the Route header for routing to the next hop, the S-CSCF would also get back the dialog identifier. Consequently, it recognizes that the new Call-ID is in fact related to the previously received INVITE request. The S-CSCF would then return to the point where it stopped after sending out the INVITE request to the AS (i.e., it will evaluate the third Filter Criteria).

We will not further consider the scenario of an AS acting as a B2BUA in this example:

```
INVITE sip:theresa@home2.hu SIP/2.0
Via: SIP/2.0/UDP sip:scscf1.home1.fr;branch=9sc2as2tb
Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=9pctb
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=8uetb
```



**Figure 12.4** Routing to an application server

```

Route: <sip:as2.home1.fr;lr>
Route: <sip:scscf1.home1.fr;lr>;dia-id=6574839201
Record-Route: <sip:scscf1.home1.fr;lr>
Record-Route: <sip:pcscf1.visited1.fi;lr>

```

### 12.3.8.3 From the AS Back to the S-CSCF

When receiving the INVITE request, the AS:

- will remove the topmost entry in the Route header that is pointing to the AS;
- provide the service based on the information in the request;
- may modify the request in compliance with [RFC3261] (e.g., add another header);
- put its own address at the top of the Via list.
- decide whether it wants to receive subsequent requests within this dialog – if it wants to then it puts its own address at the top of the Record-Route list (in this example we assume that the AS wants to stay in the Route header);
- route the INVITE request based on the topmost Route header back to the S-CSCF.

Our INVITE request now looks like:

```

INVITE sip:theresa@home2.hu SIP/2.0
Via: SIP/2.0/UDP sip:as2.home1.fr;branch=vas2tb
Via: SIP/2.0/UDP sip:scscf1.home1.fr;branch=9sc2as2tb
Via: SIP/2.0/UDP pcscf1.visited1.fi;branch=9pctb
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357;branch=8uetb

```

---

```
Route: <sip:scscf1.home1.fr;lr>;dia-id=6574839201
Record-Route: <sip:as2.home1.fr;lr>
Record-Route: <sip:scscf1.home1.fr;lr>
Record-Route: <sip:pcscf1.visited1.fi;lr>
```

### 12.3.8.4 Evaluation of Further Filter Criteria at the S-CSCF

When it receives the INVITE request again, the S-CSCF will then evaluate filter criterion #3; this does not match, because the SIP method is not SUBSCRIBE (as indicated in the SPT). Consequently, the S-CSCF will continue with its normal routing procedures, as described in Section 12.3.3.3 (i.e., it will send the INVITE request to the I-CSCF of Theresa's home network).

Because service provisioning further complicates the routing, no further attention is paid to it throughout this example; the Via, Route and Record-Route headers added here will likewise not be shown in the rest of this example.

### 12.3.9 IMS Communication Service Identification

#### 12.3.9.1 Overview

Tobias, when calling his sister, makes use of the Multimedia Telephony application on his phone. It is assumed, that on an IMS-enabled mobile phone, this application will be the default application that is used to set up multimedia calls over the IMS. This also means that Tobias

- wants the call to be treated in accordance with the IMS Multimedia Telephony specifications (see Chapter 9) within the IMS networks; and
- has a preference that the call reaches one of Theresa's terminals that also supports the IMS Multimedia Telephony Communication Service.

In order to achieve this, the IMS uses two independent SIP protocol extensions that serve different purposes for IMS Communication Service Identification:

- the SIP service identification framework as defined in [draft-drage-sipping-service-identification], which defines the P-Preferred-Service and the P-Asserted-Service headers – this is used to identify a specific communication service within the IMS network and described in Section 12.3.9.2;
- the caller preferences as defined and [RFC 3841], which allow the calling terminal to select the called users terminals based on whether the called terminal supports the communication service or not – this is described in Section 12.3.9.3.

An IMS Communication Service is identified by a service URN that is generally called IMS Communication Service Identification (ICSI) – see Section 11.9.3. When an initial request for a specific service is sent, the ICSI is included in the P-Preferred-Service headers and in addition to that as a caller preference. An IMS Application Reference Identification (IARI) can be additionally used as a caller preference and included in the call.

Tobias's S-CSCF checks whether Tobias has subscribed to the related communication service and replaces the P-Preferred-Service header with the P-Asserted-Service header. The ICSI included in the P-Asserted-Service header will be used to select the Application Servers related to the indicated communication service, in our example this will be the Telephony Application Server (TAS), which provides the Multimedia Telephony specific Supplementary Services (see Chapter 9).

Theresa's S-CSCF performs similar actions, i.e. it sends the SIP INVITE request to Theresa's TAS based on the presence of the ICSI present in the P-Asserted-Service header. Once it has worked its way through all filter criteria, it has to decide to which of Theresa's phones it should send the SIP INVITE request. In the examples given here, we assume that Theresa is registered for three different phones:

- a mobile phone that supports IMS Multimedia Telephony Communication Service (MMtel);
- a fixed office phone that supports MMtel;
- a fixed phone that is located at her home, that does not support MMtel.

By normal SIP routing procedures the INVITE request should be routed to all three phones. But as Tobias has indicated caller preferences, Theresa's S-CSCF has to look at those first in order to find out if one of the three phones should be preferred, based on its registered capabilities.

### 12.3.9.2 IMS Communication Service Identification and Service Provisioning

Upon sending the initial SIP INVITE request, Tobias's UE identifies the IMS Multimedia Telephony Communication Service by indicating the ICSI as a service URN in the P-Preferred-Service header.

```
INVITE sip:theresa@home2.hu SIP/2.0
P-Preferred-Service: urn:urn-xxx:3gpp-service-ims.icis.mmTEL
```

Note that the 'xxx' values within these URNs will be replaced by numeric values, once these URNs have been registered with the Internet Assigned Number Authority (IANA).

Once the SIP INVITE request reaches Tobias's S-CSCF, the S-CSCF checks whether Tobias has subscribed to the related service. In this case the check is successful, i.e. Tobias's user profile, that the S-CSCF downloaded during Tobias's registration from the HSS (see Section 11.5.6), includes an indication, that Tobias has subscribed to the Multimedia Telephony Communication Service. The S-CSCF then asserts the indicated ICSI by replacing the P-Preferred-Service header with the P-Asserted-Service header.

```
INVITE sip:theresa@home2.hu SIP/2.0
P-Asserted-Service: urn:urn-xxx:3gpp-service-ims.icis.mmTEL
```

After this, the S-CSCF starts to apply the Filter Criteria of Tobias's users profile (see Section 11.3.8). One of the Service Trigger Points (SPT) in one of the Filter Criteria triggers on the presence of the value 'urn:urn-xxx:3gpp-service-ims.icis.mmTEL' within the P-Asserted-Service header. As the SIP INVITE request matches this SPT, the request is

sent towards the related Application Server, which in this case is the Telephony Application Server (TAS). The procedures on how to route to and from Application Servers are described in Section 11.3.8.1. The TAS now will perform all the supplementary services that apply for the call (see Chapter 9) and later on send the request back to the S-CSCF.

After finishing the Filter Criteria evaluation, Tobias's S-CSCF will forward the SIP INVITE request towards Theresa's IMS network, where also Theresa's S-CSCF triggers on the presence of the Multimedia Telephony Service-URN within the P-Asserted-Service header, routes the request to Theresa's TAS, which then will perform the Supplementary Services which Theresa has configured and activated.

The P-Asserted-Service header is removed at the boundary of the trust domain (see Section 3.21.4.3) which in this example means that it is removed by Theresa's P-CSCF.

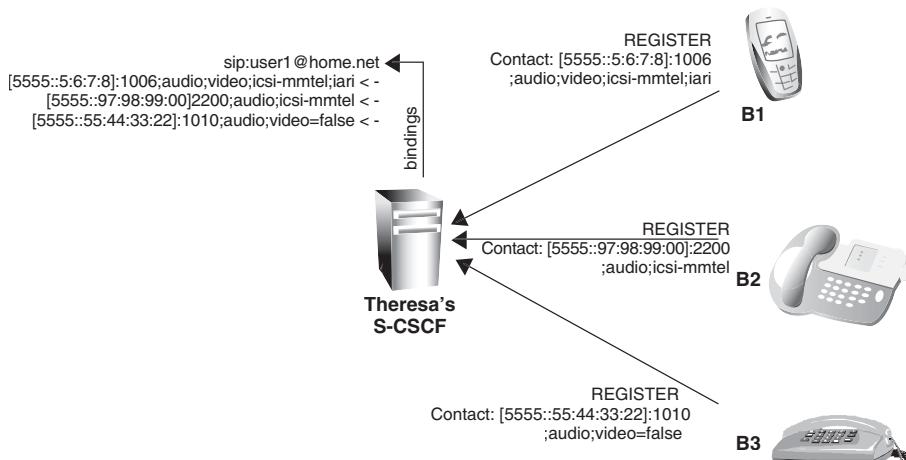
### 12.3.9.3 Routing Based on Caller Preferences

#### 12.3.9.3.1 Basic Caller Preferences Handling

In Section 11.9 it was already shown, how a UE can register feature tags and specifically ICSI and IARI values during the IMS registration procedures. These feature tags are then stored within the S-CSCF, together with the contact address (device address) from which they were registered.

As said above, we assume that Theresa has registered from three different phones, that all support different capabilities. This means that Theresa's S-CSCF currently holds three bindings for the public user identity `sip:theresa@home2.hu` (see Figure 12.5):

- Phone B1, mobile phone:
  - Contact address: [5555::5:6:7:8]:1006
  - Feature Tags (Callee Capabilities):
    - ;audio
    - ;video



**Figure 12.5** Registration of feature tags

- :g.3gpp.icsi\_ref="urn%3Aurn-xxx%3A3gpp-service-ims.icis.mmTEL"
- :g.3gpp.iari\_ref="urn%3Aurn-xxx%3Asome-app-ims.iari.proprietary-vs"
- Phone B2, office phone:
  - Contact address: [5555::97:98:99:00]:2200
  - Feature Tags (Callee Capabilities):
    - ;audio
    - :g.3gpp.icsi\_ref="urn%3Aurn-xxx%3A3gpp-service-ims.icis.mmTEL"
- Phone B3, home phone:
  - Contact address: [5555::55:44:33:22]:1010
  - Feature Tags (Callee Capabilities):
    - ;audio
    - ;video=false

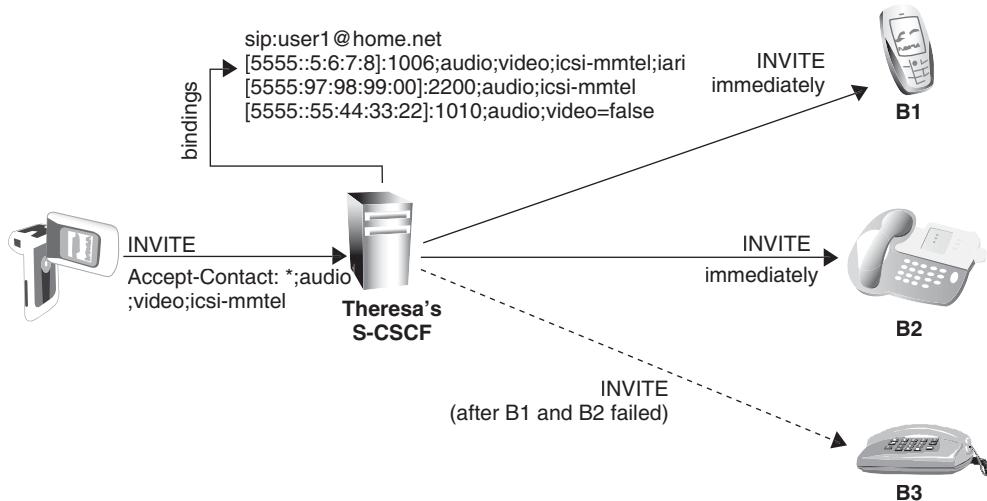
When sending the initial SIP INVITE request, Tobias's phone expresses a preference to reach one of Theresa's phones, that supports audio and video as well as the IMS Multimedia Telephony Communication Service. This preference is expressed within the Accept-Contact header, that is added to the SIP INVITE request:

```
INVITE sip:theresa@home2.hu SIP/2.0
P-Preferred-Service: urn:urn-xxx:3gpp-service-ims.icis.mmTEL
Accept-Contact: *,audio
      ;video;
      g.3gpp.icsi_ref="urn%3Aurn-xxx%3A3gpp-service-ims.
      icis.mmTEL"
Contact: <sip:[5555::1:2:3:4]:1357>
      ;audio
      ;video
      ;mobility
      ;methods=
      "INVITE, BYE, ACK, OPTIONS, CANCEL, NOTIFY, MESSAGE, PRACK,
      UPDATE"
      ;g.3gpp.icsi_ref="urn%3Aurn-xxx%3A3gpp-service-ims.
      icis.mmTEL,
      urn%3Aurn-xxx%3Aother-vendor-service-ims.
      icsi.ongame"
      ;g.3gpp.iari_ref= "urn%3Aurn-xxx%3Aother-app-ims.
      iari.firefighter"
```

Here we see that the originating UE additionally also adds all its own supported feature tags within the Contact header, in the same way as described in Section 11.9.3 for the REGISTER request.

The Accept-Contact header expresses a preference of the caller to reach a terminal of the called party (Theresa), which supports the indicated capabilities. There are further headers that allow a calling user to express specific preferences:

- Request-Disposition header, indicates whether the calling user wants the request to be forked in parallel (at the same time to all relevant UEs) or sequentially (UEs are tried one by one, if one does not answer or fails, the next UE is tried);



**Figure 12.6** Routing based on caller preferences

- Reject-Contact header, which also indicates feature tags, but only such which shall not apply to the selected terminals (e.g. the Reject-Contact header could be used to not reach a fixed phone by setting it to ;mobility="fixed").

When Theresa's S-CSCF receives the SIP INVITE request, it has to verify towards which of the terminals it has to send the SIP INVITE request, which in a simplified way works as follows (see Figure 12.6):

- phone B1 will immediately receive the SIP INVITE request, as it is registered with all three feature tags indicated in the Accept-Contact header;
- phone B2 will immediately receive the SIP INVITE request – it only has registered two of the feature tags indicated in the Accept-Contact header and did not explicitly register the ‘video’ feature tag, but as phone B2 did not explicitly indicate, that it does **not** support video, it is assumed this functionality;
- phone B3 will only receive the SIP INVITE request if both phone B1 and phone B2 do not accept the calls or fail due to other reasons – although phone B3 indicated ‘video=false’ it will receive the call, as the feature tags indicated in the Accept-Contact header are only preferences, i.e. they are in this example not a list of mandatorily required functionalities of the called terminal.

In this given example, the caller preferences in the Accept-Contact header do not exclude any of the phones to be contacted – they only cause that the two phones, which seem to support the preferred capabilities, are contacted first. This is done in order to deliver the call to Theresa, even if the indicated feature tags are not all matched.

### **12.3.9.3.2 Examples on How to Use ;Require and ;Rxplicit Parameters with Caller Preferences**

As shown so far, the feature tags in the Accept-Contact header are preferences, that influence the routing of the SIP INVITE request (see Section 12.3.9.3.1). In some cases it is desirable that the originating user could express stronger preferences or even strict requirements about the capabilities the called terminal shall support.

For this purpose two more parameters have been defined in RFC 3841:

- ‘;require’ parameter, which causes that a registered contact address will not receive the request, if its capabilities either do not match the expressed caller preferences or if the contact did not include the expressed caller preference. In other words: Tobias’s phone would only set this parameter if it has a preference to reach a phone that supports the capability and does not want to reach a phone that indicated that it does not support the capability;
- ‘;explicit’ parameter, which causes that the request is sent first to those contacts that explicitly indicated the preferred capability, whilst those, which did not indicate them, will receive the request only in the case of the other contacts failing to accept the call. In other words: this parameter does not require the specific capability from the target phone, it only tries those phones first, that explicitly stated the support for the capability;
- additionally, the combination of these two parameters ‘;explicit ;require’ means that only those contacts will receive the request, which explicitly indicated support for the capability.

Let’s look at the different possible scenarios by using the above mentioned three devices, from which Theresa has registered. For simplicity, we assume that Tobias’s phone only indicates a preference for support of video within the Accept-Contact header.

Please note that for simplicity, we will not look at the influence given by caller preference specific q-values in these examples. The calculation of those is rather complex and would go beyond the scope of this example. Also the handling of multiple Accept-Contact headers as well as the presence of the Reject-Contact or the Request-Disposition headers are not further elaborated here. Guidance and examples on how to handle these can be found in RFC 3841 as well as RFC 4596.

In the first example we assume, that the Accept-Contact header in the SIP INVITE request includes the video feature tag (caller preference) plus the ‘;require’ parameter, i.e.:

```
INVITE sip:theresa@home2.hu SIP/2.0
Accept-Contact: *;video;require
```

This means that support of the video capability is required. From Theresa’s registrations we know that:

- phone B1 supports video, this is explicitly stated;
- phone B2 did not indicate the video callee capability during registration – it might support it;
- phone B3 explicitly stated that it does not support video.

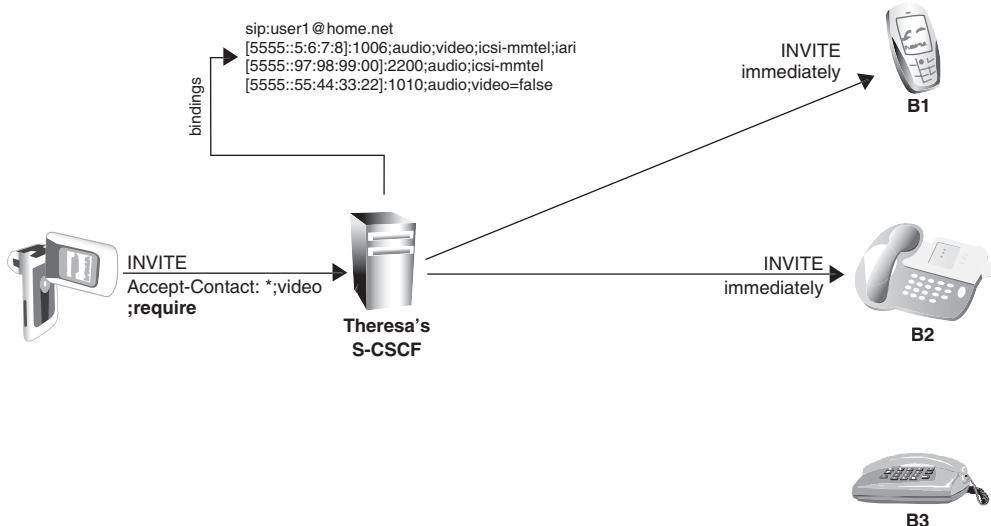


Figure 12.7 Routing based on caller preferences: require

Therefore (see Figure 12.7) the SIP INVITE request will be forked immediately to phones B1 and B2, but will never be delivered to phone B3, as it explicitly indicated that it does not support the required capability. It might be surprising that phone B2 receives the request immediately. This is due to the definition of the ‘;require’ parameter, which states that support of a specific capability is assumed, even if the registered contact did not include the related feature tag.

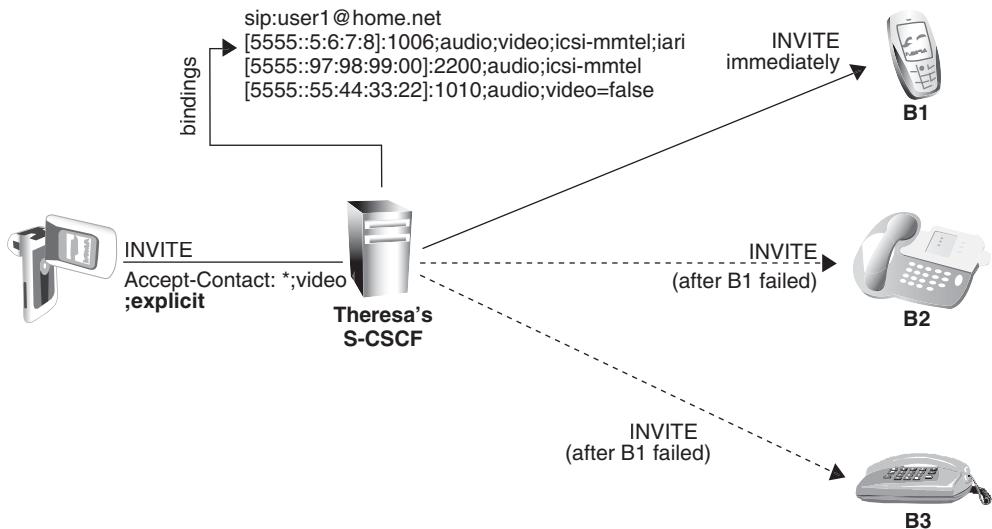
In the second example we assume that the Accept-Contact header in the SIP INVITE request includes the video feature tag plus the ‘;explicit’ parameter, i.e.:

```
INVITE sip:theresa@home2.hu SIP/2.0
Accept-Contact: *;video;explicit
```

In this case (see Figure 12.8), the SIP INVITE request will be delivered first to phone B1, as it made an explicit statement that it supports the capability. If phone B1 does not respond to the SIP INVITE request or sends a negative response, the SIP INVITE will then be delivered afterwards to phone B2 and phone B3. Both phones B2 and B3 are tried, as Tobias’s phone did not indicate that it requires the support for the capability, it only has a preference to reach a phone that supports it.

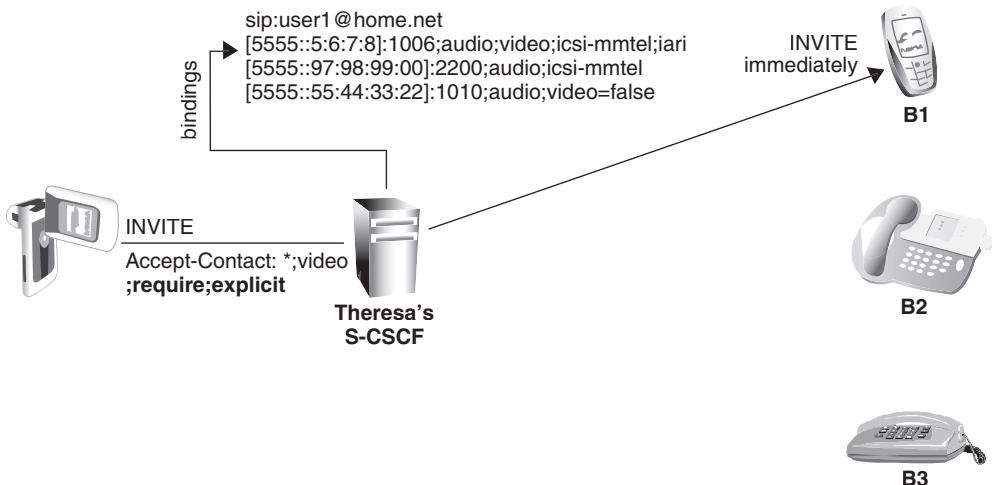
In the last example we assume that the Accept-Contact header in the SIP INVITE request includes the video feature tag and both the ‘;explicit’ as well as the ‘;require’ parameter, i.e.:

```
INVITE sip:theresa@home2.hu SIP/2.0
Accept-Contact: *;video;explicit;require
```



**Figure 12.8** Routing based on caller preferences: explicit

In this example (see Figure 12.9) Tobias's phone requires that the related capability is supported at the remote phone and that this phone has registered this capability explicitly. Therefore the INVITE request will only be delivered to Theresa's mobile phone (B1). Phone B2 did not make an explicit statement about the support of video, therefore the request will not be sent there. Phone B3 explicitly indicated that it does not support the required capability, which means that it also does not receive the request.



**Figure 12.9** Routing based on caller preferences: require; explicit

### 12.3.10 Related Standards

Standards and Links related to this section:

3GPP TS 23.218	IP Multimedia (IM) session handling; IM call model; Stage 2.
RFC 3840	Indicating User Agent Capabilities in the Session Initiation Protocol (SIP)
RFC 3841	Caller Preferences for the Session Initiation Protocol (SIP)
RFC 4596	Guidelines for Usage of the Session Initiation Protocol (SIP) Caller Preferences Extension
draft-drage-sipping-service-identification	A Session Initiation Protocol (SIP) Extension for the Identification of Services
draft-monrad-sipping-3gpp-urn-namespace	A Uniform Resource Name (URN) Namespace for the 3rd Generation Partnership Project (3GPP)
<a href="http://www.3gpp.org/tb/Other/URN/URN.htm">http://www.3gpp.org/tb/Other/URN/URN.htm</a>	URN values maintained by 3GPP

## 12.4 Compression Negotiation

### 12.4.1 Overview

The basic compression capabilities of the UE and the P-CSCF have already been negotiated during the registration procedures (see Section 11.10). Consequently, all requests and responses that are sent between the two sets of UE and their P-CSCFs will be compressed.

In this example we only show how compression parameters are basically set during session establishment and concentrate only on the compression between Theresa's UE and her P-CSCF. The procedures for Tobias's end are identical.

### 12.4.2 Compression of the Initial Request

We assume that Theresa has registered a contact address that included the `comp=SigComp` parameter at her S-CSCF. Therefore, Theresa's S-CSCF will include this parameter when it acts as a SIP registrar and re-writes the request URI of the INVITE request (see Section 12.3.3.5).

```
INVITE sip:[5555::5:6:7:8]:1006;comp=SigComp SIP/2.0
```

When the P-CSCF receives this request, it will route it toward Theresa's UE based on the request URI and, as the `comp=SigComp` parameter is included, it will send it compressed. Furthermore, the P-CSCF will:

- add the `comp=SigComp` parameter to its entry in the `Via` header, so that Theresa will send all responses to the INVITE request compressed;

- add the comp=SigComp parameter to its entry in the Record-Route header, so that Theresa will send all subsequent requests in this dialog compressed.

Our INVITE request now looks like:

```
INVITE sip:[5555::5:6:7:8]:1006;comp=SigComp SIP/2.0
Via: SIP/2.0/UDP pcscf2.home2.hu:1511;comp=SigComp
Via: SIP/2.0/UDP scscf2.home2.hu
Via: SIP/2.0/UDP icscf1.home2.hu
Via: SIP/2.0/UDP scscf1.home1.fr
Via: SIP/2.0/UDP pcscf1.visited1.fi
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357
Record-Route: <sip:pcscf2.home2.hu:1511;comp=SigComp;lr>
Record-Route: <sip:scscf2.home2.hu;lr>
Record-Route: <sip:scscf1.home1.fr;lr>
Record-Route: <sip:pcscf1.visited1.fi;lr>
Contact: <sip:[5555::1:2:3:4]:1357;comp=SigComp>
```

#### 12.4.3 Compression of Responses

When Theresa's UE constructs the 183 (Session in Progress) response to the INVITE request, it will add its IP address in the Contact header and will also include the comp=SigComp parameter there. Based on this entry all subsequent requests will be routed from Theresa's P-CSCF to her UE.

The Record-Route headers are stored by Theresa's UE: whenever the UE sends a subsequent request (e.g., a PRACK or BYE request) it will send it compressed due to the compression parameter being set in the topmost entry.

Theresa's UE will send the 183 (Session in Progress) response to the P-CSCF and, as that shows the comp=SigComp parameter in the Via header, it will also send this response compressed:

```
SIP/2.0 183 Session in Progress
Via: SIP/2.0/UDP pcscf2.home2.hu:1511;comp=SigComp;lr
Via: SIP/2.0/UDP scscf2.home2.hu
Via: SIP/2.0/UDP icscf1.home2.hu
Via: SIP/2.0/UDP scscf1.home1.fr,
Via: SIP/2.0/UDP pcscf1.visited1.fi
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357
Record-Route: <sip:pcscf2.home2.hu:1511;comp=SigComp;lr>
Record-Route: <sip:scscf2.home2.hu;lr>
Record-Route: <sip:scscf1.home1.fr;lr>
Record-Route: <sip:pcscf1.visited1.fi;lr>
Contact: <sip:[5555::5:6:7:8]:1006;comp=SigComp>
```

We saw in Section 12.3.4.2 that the P-CSCF re-writes its entry in the Record-Route header to remove its protected server port number from it. When doing so, it also removes the compression parameter from it, because it wants to receive compressed requests from the UE, and not from the S-CSCF:

```
SIP/2.0 183 Session in Progress
Via: SIP/2.0/UDP scscf2.home2.hu
Via: SIP/2.0/UDP icscf1.home2.hu
Via: SIP/2.0/UDP scscf1.home1.fr,
Via: SIP/2.0/UDP pcscf1.visited1.fi
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357
Record-Route: <sip:pcscf2.home2.hu;lr>
Record-Route: <sip:scscf2.home2.hu;lr>
Record-Route: <sip:scscf1.home1.fr;lr>
Record-Route: <sip:pcscf1.visited1.fi;lr>
Contact: <sip:[5555::5:6:7:8]:1006;comp=SigComp>
```

#### *12.4.4 Compression of Subsequent Requests*

After the 183 (Session in Progress) request has reached Tobias's UE, it will send a PRACK request in the same dialog. The request URI of this PRACK request will be sent to the address received in the Contact header of the 183 (Session Progress) response (see Section 12.3.4), which includes the compression parameter:

```
PRACK sip:[5555::5:6:7:8]:1006;comp=SigComp SIP/2.0
```

When this PRACK request is received at Theresa's P-CSCF, it will again be routed to Theresa's UE based on the request URI and can be sent compressed as it includes the compression parameter. Once again, the P-CSCF will add the comp=SigComp parameter to the Via header of the PRACK, so that Theresa can send the 200 (OK) response to it compressed.

Following these procedures, all requests and responses within the dialog will be sent compressed between the UE and their P-CSCFs.

#### *12.4.5 Related Standards*

The comp parameter is defined in:

[RFC3486] Compressing the Session Initiation Protocol (SIP).

## **12.5 Media Negotiation**

### *12.5.1 Overview*

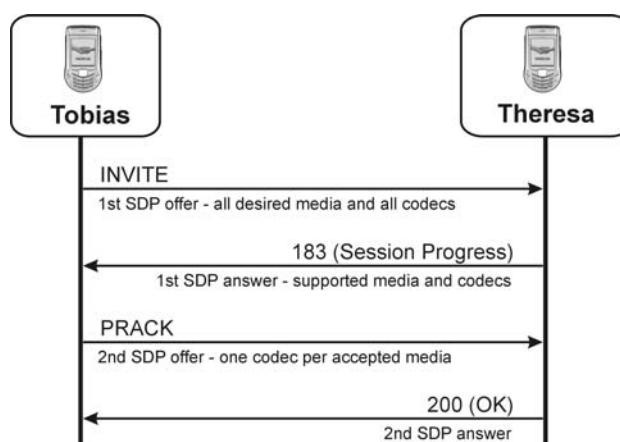
Media negotiation and the handling of preconditions, which were described in Section 11.6.4, are closely related concepts in IMS. Both are more concerned with the description of the session parameters in SDP. Nevertheless, they have a major influence on SIP signalling. In this section a normal session establishment between two phones that are connected to the IMS via GPRS access technology is shown. But that is only one of the possible scenarios and in Section 12.9 we will see different ways of establishing sessions, based on the characteristics of the access network, the resource reservation situation and the supported functionalities of the involved phones.

During media negotiation the two items of UE agree on the set of media they want to use for the session and the codecs which will be used for the different media types. Therefore, the SDP offer/answer mechanism is used, which – in IMS – basically works in the following way (Figure 12.10):

1. The calling UE sends a first SDP offer in the INVITE request to the called UE. This SDP lists all media types (e.g., audio, video or certain applications like whiteboard or chat) the caller wants to use for this session and lists the different codecs that the caller supports for encoding these different media types.
2. The called UE responds with a first SDP answer, in which it may reject some of the proposed media types. It also selects for every media type a single codec (so-called “final codec selection”) by dropping those that it does not want to use. This is done because both lots of UE must be prepared to receive any of the selected codecs and, therefore, would need to reserve resources on the air interface for the codec with the higher bandwidth, despite maybe using the codec with the lower bandwidth throughout the session.

Due to resource reservation, which is explained in Section 12.6, the two offer/answer exchanges must take place before the 200 (OK) for the INVITE is received. Consequently, the called UE needs to put the first answer in a 100-class response. We will also see in Section 12.6 that the first response is a 183 (Session in Progress) response. If this happens, two problems arise:

- the 183 (Session in Progress) response is – like all 100-class responses – a provisional response and, therefore, is not sent reliably, which means Theresa’s UE cannot be sure that it will ever be received by the calling user;
- the calling end is no longer able to send a second offer back, as during a normal INVITE transaction there is no possibility for the calling UE to send any further SIP



**Figure 12.10** SDP offer/answer in IMS

requests to the called UE besides the initial INVITE and the ACK at the very end of session establishment.

[RFC3262] solves both these problems by making the provisional 100-class responses reliable: this means that, when sending a provisional response back to Tobias's UE, Theresa's UE can indicate that it wants to send this response in a reliable way. Tobias's UE must then send back an acknowledgment (ACK) for the received provisional (PR) response: the PRACK request. As every request in SIP (besides the ACK) must be answered by a final response, Theresa's UE will send a 200 (OK) response back, after receiving the PRACK request.

With this addition to SIP, the first SDP answer in the 183 (Session in Progress) response can be sent reliably and the second SDP offer/answer exchange can be done in the PRACK request and in its 200 (OK) response.

### 12.5.2 Reliability of Provisional Responses

The 100-class responses in SIP are provisional: that is, the terminal that sends them out does not get any indication back whether these responses were ever received by the other end. As shown above, there are some cases that require provisional responses to be sent reliably: that is, that the UE that receives the response can explicitly acknowledge it. One of these cases in IMS is that the provisional response carries an SDP answer, which is obliged to be reliably delivered to the remote end.

The mechanism for sending provisional responses reliably is called, in short, “100rel” and its support is mandated for every UE that connects to IMS.

In order to indicate that it supports the 100rel mechanism, Tobias's UE includes a Supported header in the INVITE request, indicating the “100rel” option tag:

```
INVITE sip:theresa@home2.hu SIP/2.0
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister" <sip:theresa@home2.hu>
Supported: 100rel
CSeq: 1112 INVITE
Call-ID: apb03a0s09dkjdfglkj49555
```

After receiving this, Theresa's UE can start sending provisional responses reliably, as it knows that Tobias's terminal is going to acknowledge them. So, when Theresa's UE sends the 183 (Session in Progress) response, it inserts two additional headers:

```
SIP/2.0 183 Session in Progress
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister"
<sip:theresa@home2.hu>;tag=schwester
Require: 100rel
RSeq: 1971
CSeq: 1112 INVITE
Call-ID: apb03a0s09dkjdfglkj49555
```

The Require header indicates that the terminal that receives the provisional response must send a PRACK request back, in order to distinguish between multiple provisional responses, the RSeq header is included.

Tobias's UE is now requested to send a PRACK request back, in order to acknowledge the provisional 183 (Session in Progress) response:

```
PRACK <sip:[5555::5:6:7:8]:1006 SIP/2.0
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister"
<sip:theresa@home2.hu>;tag=schwester
RAck: 1971 1112 INVITE
CSeq: 1113 PRACK
Call-ID: apb03a0s09dkjdfglkj49555
```

This request is now sent to the IP address of Theresa's terminal, which was returned in the Contact header of the 183 (Session in Progress) response. In addition, the CSeq number is incremented by 1, as the PRACK is a subsequent request in the dialog that was created by the INVITE/183 (Session in Progress) exchange.

The provisional response that is explicitly acknowledged is identified in the RAck header, which includes the values of the RSeq and CSeq headers that were included in the received response. Both the RSeq and CSeq values are included in the RAck header in order to uniquely identify the 183 (Session in Progress) response.

As the PRACK is a normal SIP request, Theresa's terminal needs to return a final response to it:

```
SIP/2.0 200 OK
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister"
<sip:theresa@home2.hu>;tag=schwester
CSeq: 1113 PRACK
Call-ID: apb03a0s09dkjdfglkj49555
```

The CSeq header in this 200 (OK) response indicates that this is a final response to the PRACK request (CSeq value '1113 PRACK') and not to the INVITE request (CSeq value '1112 INVITE'). Tobias's terminal is still waiting for the final response to the INVITE request and, as we are looking at an IMS case here, this will still take some time to be sent.

### *12.5.3 SDP Offer/Answer in IMS*

The SDP offer/answer mechanism allows two users to agree on the media types and the codecs that they want to use for a specific session.

We assume that Tobias wants to use the following media types for his multimedia session with his sister:

- an audio stream – so that they can talk to each other;
- a first video stream – which is filmed by the camera built into Tobias's UE, so that Theresa can see him and, if Theresa has a similar camera, he can see her;

- a second video stream – which is recorded by an external camera that is connected to the UE; this currently pictures a wooden house in Oulu in Finland.

Therefore, the INVITE will include an SDP body that will look like this (note that not all SDP parameters are shown here):

```
v=0
o=- 2987933615 2987933615 IN IP6 5555::1:2:3:4
s=- 
c=IN IP6 5555::1:2:3:4
t=907165275 0
m=audio 3458 RTP/AVP 0 96 97 98
a=rtpmap:0 PCMU
a=rtpmap:96 G726-32/8000
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
m=video 3400 RTP/AVP 98 99
a=rtpmap:98 MPV
a=rtpmap:99 H.261
m=video 3456 RTP/AVP 98 99
a=sendonly
a=rtpmap:98 H.261
a=rtpmap:99 MPV
```

### 12.5.3.1 General SDP Parameters

Let us have a closer look at this SDP information: first, there is the heading with five lines:

```
v=0
o=- 2987933615 2987933615 IN IP6 5555::1:2:3:4
s=- 
c=IN IP6 5555::1:2:3:4
t=907165275 0
```

The v-line indicates the protocol version and is always set to 0.

The o-line holds parameters related to the owner of the session, who is in this case Tobias:

- the first parameter should include the name that Tobias wants indicated to the receiver of SDP; however, as he is already identified by the SIP From header in the INVITE request, this can be left out;
- the second parameter is Tobias's session identifier, which is a number that allows his UE to make a link between the session description and the media he wants to set up; the third parameter is the version of the session information that Tobias sends – in this case it is initialized with the same value as the session identifier, but it could have any other value;
- the subsequent parameters tell us that Tobias's UE uses the Internet (IN) and IPv6 addressing and that he is located at a terminal that has a certain address.

The s-line may include a subject for the session; but, again, this is already handled by SIP.

The c-line contains information about the connection that has to be established for the multimedia session: that is, this indicates the addresses used for the real media streams and not those for signalling. In this case Tobias's UE just indicates that he wants to receive all media for this session on the IP address that he is also using for SIP signalling: that is, the address that was assigned to it during activation of the signalling PDP context (see Section 11.3). Tobias's UE will also establish all secondary PDP contexts for media with the same IP address toward this access point.

Finally, there is the t-line, which indicates when the session was created and how long it is intended to last. There need be no time limitation to the session set in SDP, as SIP users already end a session by manually sending a BYE request. So, the second parameter of the t-line can safely be given as 0.

The number given as the time at which the session starts is based on definitions from the Network Time Protocol (NTP): this is a 64-bit, unsigned, fixed-point number representation of the seconds that have elapsed since 00 : 00 o'clock on 1 January 1900.

### 12.5.3.2 Media Lines

Individual media lines, or m-lines, represent the three different media streams that Tobias wants to send:

```
m=audio 3458 RTP/AVP 0 96 97 98  
m=video 3400 RTP/AVP 98 99  
m=video 3456 RTP/AVP 98 99
```

The first line indicates that Tobias intends to use audio for this session and that his UE will send this on the local port 3458. RTP/AVP (Audio Video Profile) will be used as the transport protocol for audio-related media and the terminal seems to be capable of coding the audio in four different ways, because the m-line gives four different formats (0, 96, 97 and 98). Later on, we will see that only three formats are supported, the last one of which (98) points to Dual-Tone Multi-Frequency (DTMF).

The last two m-lines represent two different video streams, which are intended to be sent in parallel: one shows Tobias and the other the wooden house. One is sent over the local port 3400 of Tobias's terminal and the other over port 3456. Both will be transported by RTP/AVP. The terminal holds two formats for each of them.

### 12.5.3.3 Audio and Video Formats

The m-lines in SDP include one or more formats that indicate the codecs in which the media streams are encoded. [RFC3551] includes up to 35 formats or codecs that have statically assigned RTP/AVP payload-type numbers (0 to 35). Since that assignment, many more codecs have been defined, which can also be transported via RTP/AVP. These newer codecs can be dynamically assigned to a payload-type number between 96 and 127.

In our example the first media line includes four RTP/AVP payload types, which are further explained in the subsequent lines of the SDP information. The lines between two

SDP m-lines form a block that describes in more detail the media of the m-line under which they are placed:

```
m=audio 3458 RTP/AVP 0 96 97 98
a=rtpmap:0 PCMU
a=rtpmap:96 G726-32/8000
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
```

The four format numbers that are indicated in the m-line for audio are now mapped to individual payload types or codecs.

Payload type 0 is statically assigned in [RFC3551] to the Pulse Code Modulation m-Law (PCMU; ITU-T G.711) codec. The first a-line (attribute line) shows this relation again.

The following two attribute lines map (rtpmap) the next two dynamic RTP/AVP format numbers that are indicated in the media line (96 and 97) to specific codecs. In this case:

- payload type 96 gets mapped to G.726; and
- payload type 97 gets mapped to Adaptive Multi-Rate Wideband (AMR-WB).

The last payload type gets mapped to the telephone event representation of DTMF tones, as described in [RFC2833]. This means that the terminal is able to generate standardized tones whenever the user presses a specific key. These tones are well known from normal (Circuit-Switched, or CS) phones when, say, an automatic voice response unit asks questions that the user has to answer with certain keys (e.g., ‘if you want information in English, press 1, if you want information in German, press 2’). The telephone event defines a text-based representation of these DTMF tones and other telephone-related tones that can be transported over RTP.

The first three payload types (PCMU, G.726 and AMR-WB) represent alternatives – each of which can be used to encode the media indicated in the m-line above them. The telephone event payload type cannot be seen as an alternative: it will always be possible for the UE to send this information whenever a tone needs to be generated.

Consequently, it should now be easy to interpret the last two media blocks in the SDP description:

```
m=video 3400 RTP/AVP 98 99
a=rtpmap:98 MPV
a=rtpmap:99 H.261
m=video 3456 RTP/AVP 98 99
a=sendonly
a=rtpmap:98 H.261
a=rtpmap:99 MPV
```

The terminal is able to encode the two video streams using either MPV (dynamically assigned RTP/AVP payload type 98) or H.261 (dynamically assigned payload type 99).

The first video stream is sent from the camera that pictures Tobias; as nothing further is indicated this is deemed a send-and-receive stream, which means that Theresa can send video over the same stream toward Tobias. Video of the wooden house can be watched

via the second video stream and, as it is assumed that Theresa will not be sending any video back, it is set to ‘sendonly’.

#### 12.5.3.4 Additional Bandwidth Modifiers

The bandwidth used on the session level is dependant on different factors. First of all the media type (e.g. audio, video) and the codec used to encode the media (e.g. AMR-WB for audio) different quality of service resources in order to be transported from one phone to the other. In addition to the media, which is transported by RTP, the RTP Control Protocol (RTCP) is used in order to exchange additional information about the media connection between the two involved phones. Information exchanged with RTCP relates e.g. to the amount of RTP packets lost (loss-rate) or whether a media stream is put on hold or activated again.

In order to make both UEs and the IMS network elements aware of the required resources for a specific media stream, the SDP offer includes three bandwidth modifiers:

- the application specific bandwidth modifier (AS), indicating the maximum resources in kilobits per second (kbps) that are needed for the media stream. This value includes the bandwidth required for RTP as well as for RTCP packets;
- the RTCP sender bandwidth modifier (RS), indicating the allocated resources in bits per second (bps) for RTCP messages sent from the UE that are related to the media sent by the UE;
- the RTCP receiver bandwidth modifier (RR), indicating the allocated resources in bps for RTCP messages sent from UE that are related to the media received by the UE. This value is not sent in a two-party call, i.e. in a call between only two endpoints, as both phones are actively sending media.

The total of the normal RTCP bandwidth, i.e. for sending and receiving together, is about 5% of the application specific bandwidth. Note that the AS is expressed in kilobits per second whilst RR and RS are expressed in only bits per second.

The SDP offer shows typical bandwidth modifier values that are used by Tobias’s phone:

```
m=audio 3458 RTP/AVP 0 96 97 98
a=rtpmap:0 PCMU
a=rtpmap:96 G726-32/8000
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
b=AS:75
b=RS:937
m=video 3400 RTP/AVP 98 99
a=rtpmap:98 MPV
a=rtpmap:99 H.261
b=AS:92
b=RS:1150
m=video 3456 RTP/AVP 98 99
a=sendonly
```

```
a=rtpmap:98 H.261
a=rtpmap:99 MPV
b=AS:92
b=RS:1150
```

As the exchange of bandwidth parameters is straight forward, these SDP parameters will not be shown further in this example.

### 12.5.3.5 Usage of AVPF and SDP Capability Negotiation

For the IMS Multimedia Telephony communication service, an extension to the RTP audio-visual profile (AVP) is used, the Extended RTP Profile for Real-Time Transport Control Protocol (RTCP)-Based Feedback, in short RTP/AVPF or just AVPF. AVPF enables the two clients involved in the multimedia session to adapt their media characteristics more efficiently during the transmission of the media streams by making a more effective usage of the RTCP feedback-mechanism. The details of AVPF will not be further discussed here, as the profile is related to the media transmission only.

As IMS Multimedia Telephony mandates AVPF as an alternative to AVP, AVPF should be indicated within the SDP m-lines. But it might be that the called phone does not support AVPF and therefore would not understand the SDP m-line and reject it. In order to avoid such behaviour and allow a fallback mechanism from AVPF to AVP in case the called phone does not support AVPF, the SDP Capability Negotiation mechanism is applied for IMS Multimedia Telephony.

SDP Capability Negotiation offers the possibility of negotiating several SDP capabilities between the phones involved in an offer/answer exchange. In this example we will only concentrate on the capability negotiation for AVPF.

When sending the first SDP Offer, Tobias's phone indicates normal AVP support in the m-line. This will guarantee that the m-line is understood by Theresa's phone even if it only supports AVP. In addition to that, Tobias adds two more attribute lines to the every media line (in this example we only show one media line):

```
m=audio 3458 RTP/AVP 0 96 97 98
a=tcap:1 RTP/AVP RTP/AVPF
a=pcfg:1 t=1|2
```

The first attribute line (a=tcap) indicates the two alternative transport protocol (tcap) capabilities that Tobias's UE supports: RTP/AVP (number 1) and RTP/AVPF (number 2). The two protocols are numbered automatically – the first protocol takes the number that is indicated after 'a=tcap:', which in this case is 1, this number is then incremented by one for all following protocol indications (in this case only RTP/AVPF, which takes the number 2).

The second attribute line (a=pcfg) is the potential configuration attribute line. As there can be several configuration options that are negotiated within a single SDP, the a=pcfg enumerates all possible configuration decisions, i.e. this is the first (and only) configuration option, therefore it takes the number 1: 'a=pcfg:1'. This potential configuration option indicates that for the transport protocols (t=) Tobias's phone supports both options number 1 (RTP/AVP) and number 2 (RTP/AVPF) as numbered in the line above.

If Theresa's phone does not support AVPF and the SDP Capability Negotiation Mechanism, it will simply disregard the additional attribute lines and proceed with the SDP Offer/Answer exchange and using RTP/AVP, as this is indicated in the media line.

In this example we assume that Theresa's phone supports AVPF and the SDP Capability Negotiation Mechanism. It therefore sees the two options indicated in the SDP Offer and chooses to use AVPF for the media session. Theresa's phone sends back an SDP Answer with the following information:

```
m=audio 3458 RTP/AVPF 0 96 97 98
a=acfg:1 t=2
```

The transport protocol in the media line has changed to RTP/AVPF. The additional attribute line is the actual configuration attribute (a=acfg), which references the only potential configuration option that was received in the SDP Offer and was numbered 1: 'a=acfg:1'. The 't=2' parameter indicates that the second configuration option was chosen, which was AVPF.

After this, no further exchange of SDP Capability Negotiation is needed between the terminals for the duration of the related media stream. This means that also subsequent SDP Offer/Answer exchanges will not include any of the here shown additional attribute lines related to this media line, for which the usage of RTP/AVPF was already successfully negotiated.

#### 11.5.3.4 The first SDP offer and answer exchange.

The SDP information as shown above is the initial offer, and Tobias's UE sends it within the body of the INVITE request to Theresa. The SDP offer arrives at Theresa's UE due to SIP routing of the INVITE request. Theresa's UE afterwards generates an SDP answer for the received offer, which looks like this:

```
v=0
o= - 1357924 1357924 IN IP6 5555::5:6:7:8
s=-
c=IN IP6 5555::5:6:7:8
t=907165275 0
m=audio 4011 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=acfg:1 t=2
m=video 4012 RTP/AVPF 99
a=rtpmap:99 H.261
a=acfg:1 t=2
m=video 0 RTP/AVPF 98
a=acfg:1 t=2
```

The SDP header includes information about Theresa's UE – i.e., the IP address in the o-line along with the session identification and the version, as well as the UE's IP address in the c-line.

From the m-lines we see that the terminal is capable of handling the audio stream (on port 4011) and the first video stream (on port 4012), but cannot deliver the second video

stream to the user; therefore, the port number in the third m-line is set to zero and all related a-lines are simply dropped.

The SDP answer must repeat all the m-lines that are included in the SDP offer; therefore, the terminal cannot drop the last m-line from here: the only way to indicate that this video cannot be handled is to set the port number to 0.

Furthermore, the UE chooses to use H.261 as the media for the video stream and therefore drops the MPV codec for video: this RTP/AVP payload-type value was dropped from the first video m-line, and the related a-lines are omitted in the answer.

For audio the UE wants to make use of the AMR-WB codec and therefore drops the PCMU and G.726 payload types. By setting the related values, Theresa's UE takes the final codec decision for all the media streams indicated in the SDP. Additionally, DTMF signals can be represented as telephone events.

For all media streams Theresa's phone will make use of RTP/AVPF (as indicated in the media lines) instead of RTP/AVP. It indicates that it has accepted the second possible configuration option for the transport protocol in the a=acfg line.

### 12.5.3.6 Is a Second SDP Offer and Answer Exchange Necessary?

Due to the sending of the PRACK request, which needs to be answered by a 200 (OK) response, there is the possibility to send a second SDP Offer/Answer exchange immediately after the first one, which is performed in the INVITE request and its 183 (Session Progress) response. This is usually not necessary, as during the first SDP Offer/Answer exchange the media and codecs for the call have been selected and agreed upon between the two UEs.

There is nevertheless an exception for UEs of earlier IMS releases. In 3GPP Release 5 the terminating UE (in this case Theresa's phone) was not mandated to take the final codec selection as described in the section above. In this case, Tobias's UE could receive in Theresa's SDP Answer a list of codecs for one or more media streams, for example, the m-line for the audio stream could look as follows:

```
m=audio 4011 RTP/AVPF 96 97 98
a=rtpmap:96 G726-32/8000
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
```

In this case, Theresa's phone would not have taken the final codec selection and Tobias's UE would need to do so. Therefore the phone would send out another SDP Offer (within the PRACK request), in which the final codec would be selected:

```
m=audio 3458 RTP/AVP 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
```

And Theresa's phone, after receiving this, would need to send an SDP Answer within the 200 (OK) response to the SIP PRACK request.

As said above, this is not the normal procedure in IMS, but it is possible that this procedure is needed for fallback purposes.

#### 12.5.4 Related Standards

Specifications relevant to Section 12.5 are:

3GPP TS 24.173	IMS Multimedia telephony service and supplementary services
3GPP TS 26.114	IMS Multimedia telephony; Media handling and interaction
RFC 4566	SDP: Session Description Protocol.
RFC1305	Network Time Protocol (Version 3) Specification, Implementation and Analysis.
RFC2833	RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals.
RFC3262	Reliability of Provisional Responses in the Session Initiation Protocol (SIP).
RFC3264	An Offer/Answer Model with SDP.
RFC3550	RTP: A Transport Protocol for Real-time Applications.
RFC3551	RTP Profile for Audio and Video Conferences with Minimal Control.
RFC 3556	SDP Bandwidth Modifiers for RTCP Bandwidth
RFC 4585	Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF)
draft-ietf-mmusic-sdp- capability-negotiation	SDP Capability Negotiation

## 12.6 Resource Reservation

### 12.6.1 Overview

The media sessions between Tobias and Theresa are negotiated via SIP and SDP signalling. As in our example both UEs make use of a dedicated signalling PDP context for the transport of SIP signalling, they will have to establish one or more media PDP contexts for the transport of the media streams.

The procedure for establishing media PDP contexts is called ‘resource reservation’. Both UEs perform these procedures completely independently of each other.

The establishment of media PDP contexts may consume some time and may even fail: when, say, insufficient resources are available over the wireless link. This means that, until the resources have been reserved, it cannot be guaranteed that the agreed media sessions can be established at all.

Therefore, Theresa’s UE should not inform her about the incoming session request (INVITE): that is, it should not start to ring until it has confirmation that resource reservation has succeeded locally as well as at the calling user’s end.

In order to achieve this, both UEs exchange preconditions during the SDP offer/answer negotiation, which basically instruct:

- Tobias's UE to send a SIP UPDATE request to Theresa's UE, when resource reservation has succeeded at Tobias's UE; and
- Theresa's UE not to ring until it receives the SIP UPDATE request from the remote end and has also successfully reserved its own/local resources.

Furthermore, the preconditions indicate what should happen with a session if a specific media stream could not be successfully reserved.

Figure 12.11 gives an overview of the relations between SIP, SDP offer/answer, resource reservation, preconditions, the SIP PRACK and the SIP UPDATE method during the establishment of a session.

### 12.6.2 *The 183 (Session in Progress) Response*

As shown at the beginning of this chapter, the IMS session establishment attempt should only be indicated to Theresa (the called user) if resource reservation on the wireless link for both users has succeeded. Only after successful resource reservation will the SIP 180 (Ringing) response for the INVITE message be sent by Theresa's terminal.

On the other hand, Tobias's UE expects an early response from the called terminal, especially as SDP offer/answer negotiation in IMS needs to be done before resource reservation can start. Therefore, after receipt of the INVITE message, Theresa's UE will send back the 183 (Session in Progress) response, indicating that session establishment procedures have been started, but the called user has not been informed yet.

### 12.6.3 *Are Preconditions Mandatorily Supported?*

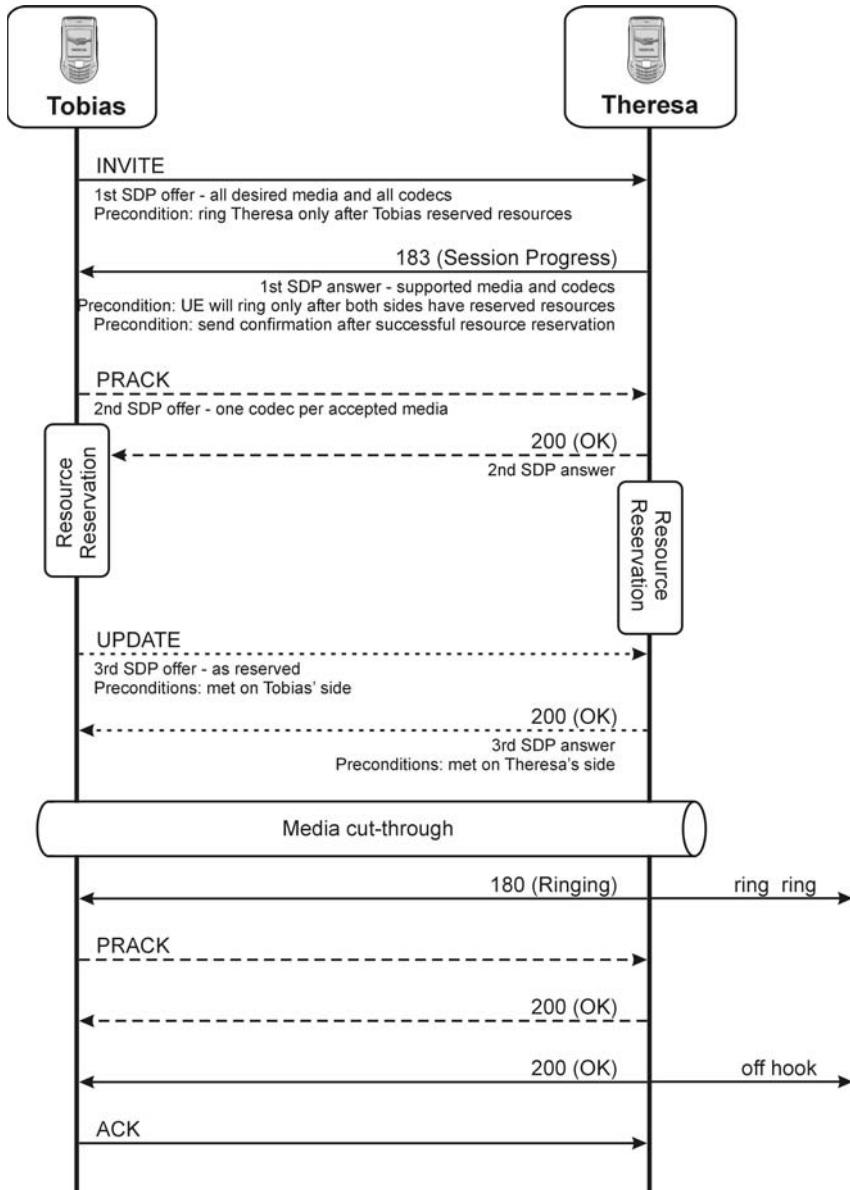
The preconditions mechanism is required to be supported by all phones that want to connect to IMS, with the exception of very specific applications, such as e.g. instant messaging, for which no specific resources need to be reserved. Nevertheless, a phone attached to IMS will not require the remote side to support the preconditions mechanism. Clause 12.9 will show several scenarios, in which preconditions are used within IMS and also shows interworking scenarios with non-IMS SIP phones, which do not support the preconditions mechanism.

In order to illustrate the need for the support of the preconditions mechanism let us assume for a moment that:

- Theresa's terminal does not support the preconditions mechanism but, nevertheless, goes ahead with session establishment based on the first received SDP Offer from Tobias's side (Figure 12.12);
- the fallback-mechanisms (a=inactive) as described in clause 12.9 are not applied; and
- resource reservation at Theresa's end is successful.

Consequently, Theresa's phone could immediately after receiving the INVITE request indicate the incoming call to Theresa and start sending early media towards Tobias.

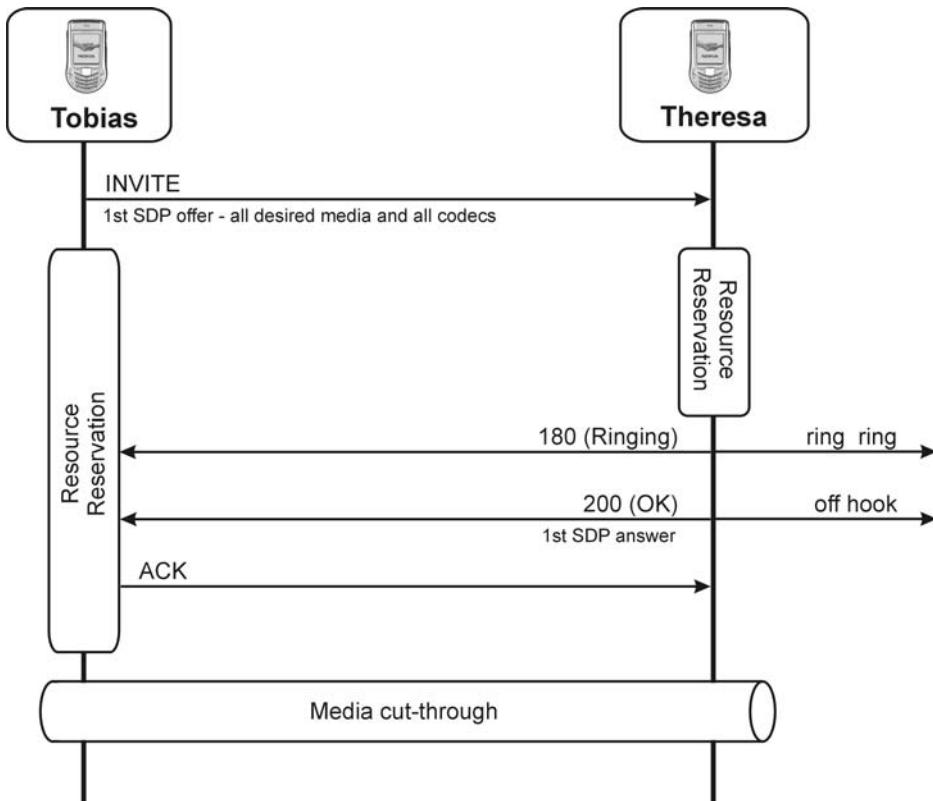
At this point in time, Tobias would not even have started to reserve his resources over the wireless link, as he is still waiting for the SDP answer from Theresa's end. Even had he started reserving resources immediately after sending out the initial INVITE request,



**Figure 12.11** SIP, SDP offer/answer and preconditions during session establishment

it cannot be guaranteed that this process would have finished when Tobias receives the 180 (Ringing) response from Theresa's UE.

Even worse, Theresa could pick up the phone – which would result in her terminal sending a 200 (OK) response – and start talking, while Tobias is still in the process of resource reservation. If resource reservation at Tobias's end fails in this scenario, Theresa



**Figure 12.12** SIP session establishment without preconditions

will be left with a call that was indicated as successful on the signalling level, but never had a chance to connect through the session on the media plane.

From this we see that the use of the precondition mechanism is essential to guarantee reliable media session establishment.

#### 12.6.4 Preconditions

[RFC3312] introduces the SDP preconditions mechanism that allows a UE to delay completion of a SIP session establishment until one or both ends have successfully completed their resource reservation. This extension to SIP and SDP is mandatorily supported by every UE that connects to IMS.

Up to now, Internet Engineering Task Force (IETF) has only defined the ‘qos’ precondition type, but there might be more to come. The ‘qos’ tag means that the precondition is set due to certain quality requirements of the related media stream, which is indicated in the m-line. The Quality of Service (QoS) depends mostly on the bandwidth that is reserved over the wireless link and the priority by which the routers in the network handle the packets that transport the ‘chunks’ of speech and video data.

#### 12.6.4.1 Preconditions in the First SDP Offer

As explained in Section 12.5 for the SDP offer/answer mechanism, Tobias's UE proposes three different media types, which are offered in the first INVITE to Theresa's UE. In order to illustrate the preconditions mechanism, we take just one of these three media types as an example and extend it with precondition-specific indications (written in bold):

```
m=audio 3458 RTP/AVP 0 96 97 98
a=rtpmap:0 PCMU
a=rtpmap:96 G726-32/8000
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
```

Every m-line block in SDP needs to indicate a separate set of preconditions. Remember that the m=audio line is only taken as an example, the two remaining m-lines in the original SDP would also include precondition setting. First, we look at the penultimate line:

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

This indicates the desired (des) quality of service (qos) precondition at the calling user (local) end. The resources for the calling user need to be reserved in both the sending and receiving directions (sendrecv), as the audio stream is bidirectional (i.e., both users can talk to each other and hear what the other says). It also states that the session will not take place if the indicated resources cannot be reserved by the calling user (mandatory).

The last line:

```
a=des:qos none remote sendrecv
```

indicates the desired (des) quality of service (qos)-related preconditions at the called user (remote) end. As the calling and the called terminal are not directly connected to each other, the calling terminal is unaware of how the remote terminal is attached to the network. It might be that Theresa's terminal does not need to perform any resource reservation, as it is connected via a CS telephony network. Therefore, the calling end can only indicate that, if the remote end needs to reserve resources, then they should be reserved in both the sending and receiving directions (sendrecv); however, the calling end is currently unaware that this is really required in order to get a media session established (none).

Up to now, the calling terminal has only expressed the desired (des) preconditions for each end, but it also needs to talk about the current status of resource reservation. This is the subject of the first two new lines:

```
a=curr:qos local none
a=curr:qos remote none
```

These two lines indicate that currently (curr) no (none) quality of service (qos)-related pre-conditions have been fulfilled by either the calling end (local) or the called end (remote).

The a=des lines make it possible to set preconditions for the local and the remote user, while the a=curr lines indicate the extent to which the set preconditions are already being fulfilled.

#### 12.6.4.2 Preconditions in the First SDP Answer

As we know, Theresa's terminal is also attached over a wireless link to IMS and supports the preconditions mechanism. Therefore, it will respond to the received SDP offer with a well-formed answer and will include its own preconditions. Once again, we only show here those preconditions for the first media type:

```
m=audio 4011 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
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
```

The important thing to note here is that the remote and local ends have changed, because from Theresa's point of view her terminal is local and Tobias's is remote. Her terminal now indicates its own preconditions line by line. Line 5:

**a=curr:qos local none**

it has currently not reserved any local qos-related resources. Line 6:

**a=curr:qos remote none**

it received information from the remote end (in the first offer) that no qos-related resources have been reserved at the moment. Line 7:

**a=des:qos mandatory local sendrecv**

it mandatorily requires that its own resources get reserved in both the sending and receiving directions, before the audio session can start. Note that the initial value 'none', as set from the calling end, has changed to 'mandatory' because Theresa's terminal is attached via the air interface and, therefore, is also mandated to reserve the resources locally before it can start sending media. Line 8:

**a=des:qos mandatory remote sendrecv**

it received information from the remote end (in the first offer) that resources are mandatorily reserved in both the sending and receiving directions. Line 9:

**a=conf:qos remote sendrecv**

the calling terminal (remote) should send a confirmation (conf) at the moment the resources (qos) have been reserved in the sending and receiving directions (sendrecv). This is a new line that the called end adds to SDP. It is a necessary addition because the called terminal is not intended to ring the called user or to start sending media until *both ends have reserved the resources*.

This SDP answer is now sent in the 183 (Session in Progress) response to the calling terminal.

#### 12.6.4.3 Resource Reservation

Theresa's phone can start with resource reservation at the same moment when the 183 (Session Progress) response with the SDP Answer is sent out, as it has all the information about media sessions and related codecs available.

Tobias's phone starts with resource reservation immediately after receiving the 183 (Session Progress) response. In addition, Tobias's phone will send out the SIP PRACK request.

Both UEs now try to reserve the requested resources. In the General Packet Radio Service (GPRS) case this would mean that they establish one or more media PDP contexts as secondary PDP contexts (see Section 12.12.2.2). It is possible for either Tobias's or Theresa's UE to finish the reservation first. In either case Theresa's terminal must not start to indicate the incoming call to Theresa (i.e., it must not ring) before knowing that both ends have successfully reserved resources.

Let us assume that the called end finishes the reservation first; in this case it will simply wait until it gets the confirmation that it requested from the calling end in the 183 (Session in Progress) response answer by setting the a=conf line in the first SDP answer.

As soon as the calling end has reserved the required resources, it will confirm this to the called terminal by sending a SIP UPDATE request as a subsequent request in the dialog that was established with the INVITE request. This UPDATE request will include a third SDP offer in its body, which shows that the resources at the calling end have been reserved:

```
m=audio 3458 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

The only difference here is that the first a=curr line has changed from 'none' to 'sendrecv'. Consequently, Tobias's terminal indicates that the status of the local QoS-related resources for the audio stream has changed. They have now been successfully reserved in both the sending and receiving directions.

As Theresa's terminal was the first to secure resource reservation, it can immediately start to ring after sending the 200 (OK) response for the UPDATE request, because it is now sure that both ends have sufficient resources reserved to send and receive the audio stream. When it starts to ring, it sends a 180 (Ringing) response to the INVITE request in parallel.

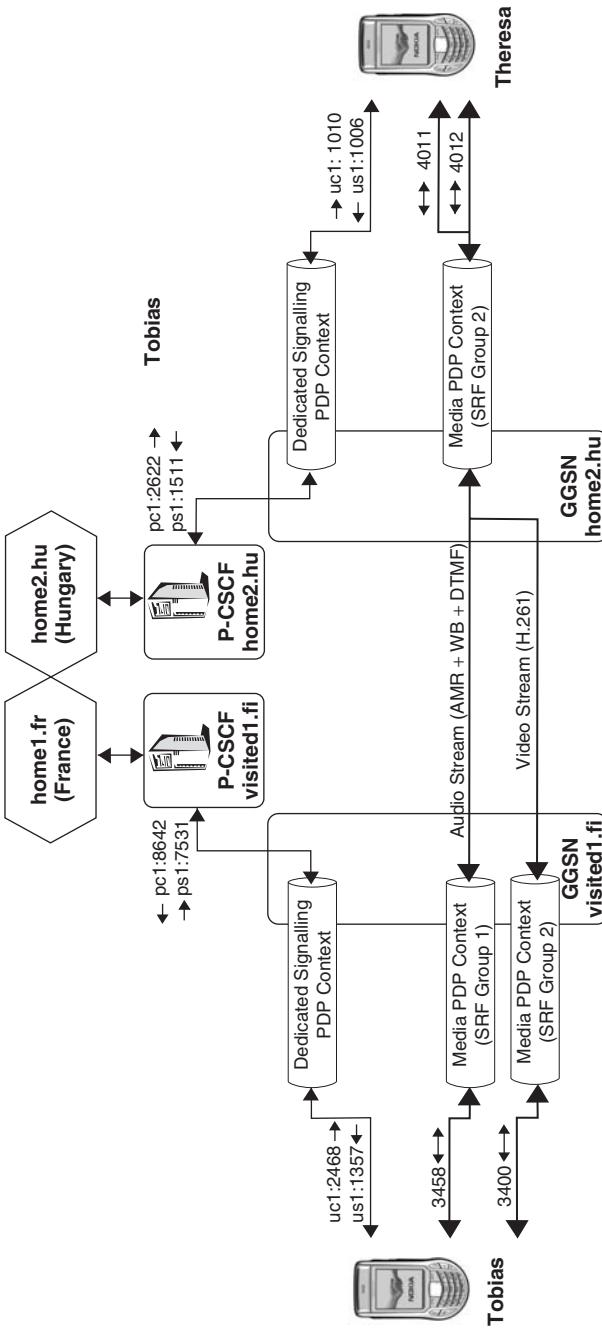


Figure 12.13 Media streams and transport in the example scenario

The 200 (OK) for the UPDATE will include a third SDP answer with the following SDP information about the audio stream:

```
m=audio 4011 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

We can see from this that all the current states for resource reservation match the desired states; so, the preconditions negotiation has been successful and has finished.

Let us assume that Theresa's resource reservation takes longer than Tobias's: in this case Theresa's UE will receive the UPDATE request while it is still reserving resources. Although it will not start to ring, it will send back a 200 (OK) response with the following SDP information about the audio stream:

```
m=audio 4011 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local none
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

The only difference here is that the status of local resource reservation is still set to 'none'. After local resources have been reserved, Theresa's UE will start to ring and send a 180 (Ringing) response that will not carry another SDP body. Nevertheless, Tobias's UE can interpret the 180 (Ringing) response as an indication that resources at the remote end (i.e., Theresa's) have been reserved successfully: this is the reason why the 180 (Ringing) response also has to be sent reliably, obliging Tobias's UE to answer it with a PRACK request (see Section 12.5.2).

Once both ends have reserved their resources, media can be exchanged between the two UEs.

### *12.6.5 Establishing the Media Resources and PCC Related Actions*

In this example we do not look into the detailed interactions with PCC that occur during the session establishment, they are generally outlined in Section 3.10. The P-CSCFs will inform the PCRF about the ongoing SDP offer/answer exchanges during the session establishment.

After the preconditions have been met, each UE can begin resource reservation for the agreed media. In the case of the GPRS and when using a dedicated signalling PDP context, this means that each UE will now try to establish one or more secondary PDP contexts for the media.

In our case we assume that Tobias sets up two different media PDP contexts, one for the audio stream and one for the video stream. Each of these PDP contexts is established as a secondary PDP context, as the signaling is done via the primary PDP context.

Theresa's UE, on the other side of the connection, creates only a single media PDP context, which is also a secondary one.

The GGSN, when receiving the request for a new PDP context, start to perform the PCC actions for QoS authorization. For example Tobias's GGSN will start to act as an Access Gateway and query the PCRF whether the request for two new PDP contexts is coming in. The PCRF in this case grants the request and therefore the GGSN can establish the PDP contexts and later on through-connect the media.

### 12.6.6 Media Policing

The P-CSCF and S-CSCF can reject certain media types or codecs that are offered in SDP. This might be due to operator policy. One reason for this could be that an operator does not allow the use of any unknown media types or unknown codecs, as the network would not be able to charge for such media.

If a CSCF detects that an unsupported media type or codec is included in the SDP offer, it will reject the request with 488 (Not Acceptable Here) and will indicate the supported media types in the body of the response.

In the example scenario this is not assumed to happen. Figure 12.14 shows a worst case scenario when every CSCF on the route does not like certain media elements. Such extreme scenarios are unlikely to occur in reality.

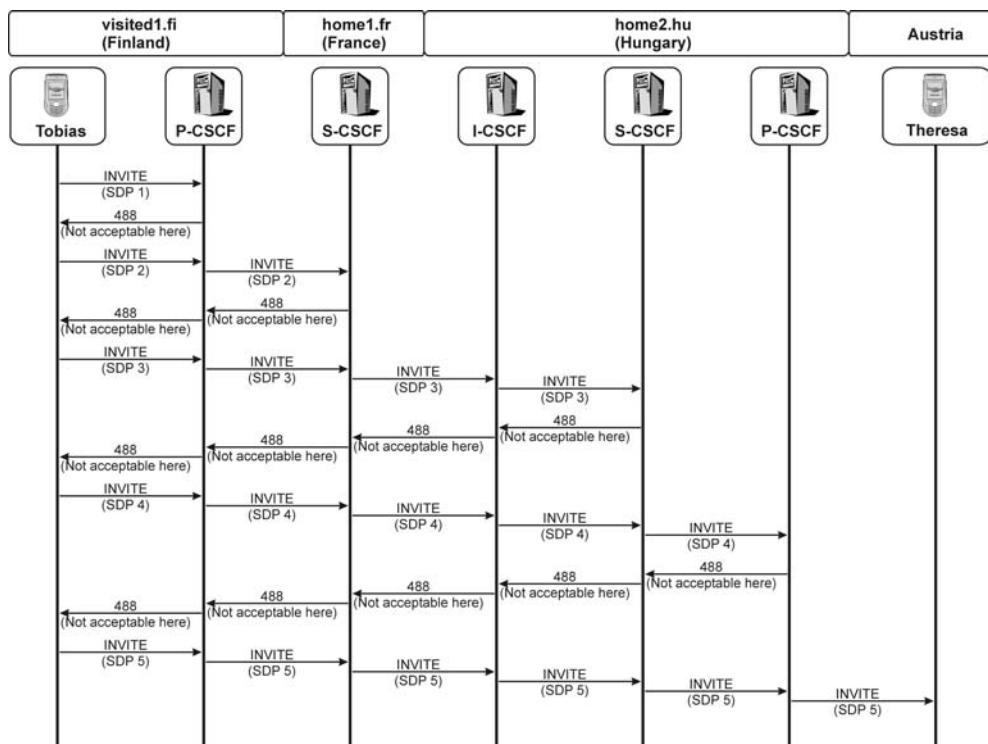


Figure 12.14 Worst case scenario for media policing

### 12.6.7 Related Standards

Specifications relevant to Section 12.6 are:

- |         |   |
|---------|---|
| RFC3311 | The Session Initiation Protocol (SIP) UPDATE Method.                |
| RFC3312 | Integration of Resource Management and Session Initiation Protocol. |

## 12.7 Charging-Related Procedures During Session Establishment for Sessions

### 12.7.1 Overview

In the case of a GPRS access network, charging is usually done on the basis of the amount of data that is sent via a PDP context. The network operator can decide to charge specific PDP contexts differently: for example, a Multimedia Message Service (MMS) could be charged differently from ‘normal’ Internet traffic. It is likely that operators will want to charge the signalling and media traffic that is related to the IMS in a different way than one based on the amount of transmitted data. To achieve this, IMS needs to correlate GPRS-specific charging records with IMS-specific charging information.

For the media session Tobias’s UE needs to establish several media PDP contexts with the GGSN. The GGSN will start generating charging records whenever data (i.e., media) are sent over that PDP context. These charging records are based on the GPRS Charging ID (GCID) that is generated by the access network.

As described in Section 11.12, an IMS Charging ID (ICID) is created during initial registration by the P-CSCF. During a session the P-CSCF creates an additional ICID for the charging of media streams that are transported over established media PDP contexts.

The GGSN sends GCIDs of these media PDP contexts to the P-CSCF, which then creates an ICID and sends it toward the home network of the served user.

These procedures will take place at both ends (i.e., for Tobias and for Theresa). Which of them will get charged what will be decided by the charging entities in their home networks.

The S-CSCFs of the users will, furthermore, distribute within their home networks the addresses of the Charging Data Function (CDF) and the Event Charging Function (ECF) that will collect the charging records for the two users. This information is needed by all entities that create charging records for the session.

Besides the Charging-IDs also the Inter-Operator Identifier (IOI) is exchanged for charging purposes between the network elements. The IOI is a globally unique identifier which is used to transfer the identities and roles of the different networks within a call. The following types of IOIs exists:

- originating IOI – set by those network entities that serve the originating user (e.g. the originating P-CSCF, S-CSCF, a MGCF in the originating network, an AS originating a call on behalf of the user);
- terminating IOI – set by those network entities that serve the terminating user (e.g. the terminating P-CSCF and S-CSCF);

- Type 1 IOI – used between the P-CSCF and the S-CSCF. A type 1 IOI uses the prefix ‘Type 1’ when being encoded;
- Type 2 IOI – used between S-CSCFs and also from and to MGCFs as well as for Application Servers using PSIs. A type 2 IOI does not have a specific prefix;
- Type 3 IOI – between the CSCFs and an AS. A type 3 IOI uses the prefix ‘Type 3’ when being encoded.

### 12.7.2 Inter-Operator Identifier Exchange of ICID for a Media Session

When Tobias’s P-CSCF receives the INVITE request for the session it will create a new ICID and include it in a P-Charging-Vector header that it adds to the INVITE request. It will also add the type 1 Inter-Operator Identifier (IOI).

```
INVITE sip:theresa@home2.hu SIP/2.0
P-Charging-Vector: icid-value="AyretyU0dm+
602IrT5tAFrbHLso=023551025"
;orig-ioi="Type 1 visited1.fi"
```

Tobias’s S-CSCF change the originating Inter-Operator Identifier (IOI) in the P-Charging-Vector header to a Type 2 IOI, which does not take a specific prefix:

```
INVITE sip:theresa@home2.hu SIP/2.0
P-Charging-Vector: icid-value="AyretyU0dm+602IrT5tAFrbHLso=023551025"
;orig-ioi=home1.fr
```

The S-CSCF of Theresa’s home network will store and remove the received orig-ioi parameter for the P-Charging-Vector header and will be sent further with the INVITE request until it reaches the P-CSCF of Theresa, where it will be removed. Charging-related headers are never sent toward a UE. All entities on the way that are involved in charging for this session will store the ICID.

When receiving the first response to the INVITE request – i.e., the 183 (Session in Progress) response – Theresa’s P-CSCF will once again add the P-Charging-Vector header, including the same ICID value as received in the INVITE request and also including the type 1 IOI, this time for the terminating side:

```
SIP/2.0 183 Session in Progress
P-Charging-Vector: icid-value="AyretyU0dm+
602IrT5tAFrbHLso=023551025"
;term-ioi="Type 2 home2.hu"
```

Theresa’s S-CSCF will change the terminating IOI information to a type 2 IOI, before sending it further:

```
SIP/2.0 183 Session in Progress
P-Charging-Vector: icid-value="AyretyU0dm+
602IrT5tAFrbHLso=023551025"
;term-ioi=home2.hu
```

The terminating IOI information will again be stored and removed by Tobias's S-CSCF, and his P-CSCF will remove the P-Charging-Vector from the response.

### 12.7.3 Correlation of GCID and ICID

The media PDP context at Tobias's end is established after successful resource reservation, and, as we saw in Section 12.6, Tobias's UE will then immediately send an UPDATE request toward Theresa. As this is the first request that is received by Tobias's P-CSCF after the media PDP context has been established, it will include in the UPDATE request the P-Charging-Vector header with the following information:

- The ICID that it created for this media session.
- The charging parameters for every media PDP context that were established for this session:
  - the GCID that was received from the GGSN;
  - the address of the related GGSN;
  - an indication ('pdp-sig') of whether the related PDP context is a signalling PDP context or not (in this case it is not a signalling PDP context); and
  - the flow identifier of the media stream.

The UPDATE request sent onwards from Tobias's P-CSCF now looks like:

```
UPDATE sip:[5555::5:6:7:8]:1006 SIP/2.0
P-Charging-Vector: icid-value="AyretyU0dm+
602IrT5tAFrbHLso=023551024"
;orig-ioi="Type 1 visited1.fi"
;ggsn=[5555::4b4:3c3:2d2:1e1];
;pdp-sig=no; gcid=723084371
;flow-id=1
;ggsn=[5555::4b4:3c3:2d2:1e1]
;pdp-sig=no;
gcid=723084372
;flow-id=2
```

As we saw in Section 12.7.3, Tobias has established two media PDP contexts; hence, two charging parameter lists are also included.

Tobias's S-CSCF will store and remove all data from the P-Charging-Vector header and add the originating and terminating IOI parameters, before sending the UPDATE request to Theresa's S-CSCF:

```
UPDATE sip:[5555::5:6:7:8]:1006 SIP/2.0
P-Charging-Vector: icid-value="AyretyU0dm+
602IrT5tAFrbHLso=023551024"
;orig-ioi=home1.fr
;term-ioi=home2.hu
```

Once again, these two parameters (orig-ioi and term-ioi) will be removed by Theresa's S-CSCF before sending the UPDATE request to the P-CSCF, which removes the header before sending the request toward Theresa's UE.

In our example Theresa's UE has already finished resource reservation at the moment the UPDATE request is received from Tobias's end (see Section 12.6.4). Therefore, Theresa's P-CSCF will include the P-Charging-Vector header again in the 200 (OK) response to the UPDATE request, this time with all the information related to the reserved media PDP context:

```
SIP/2.0 200 OK
P-Charging-Vector: icid-value="AyretyU0dm+
602IrT5tAFrbHLso=023551024"
;term-ioi="Type 1 home2.hu"
;ggsn=[5555::802:53:58:6]
;pdp-sig=no ;gcid=306908949
;flow-id=2
```

As Theresa established only one media PDP context (see Section 12.7.3), her P-CSCF includes exactly one charging parameter list.

The S-CSCFs now behave in the same way as before:

- Theresa's S-CSCF will remove PDP context-related information from the P-Charging-Vector header and will add the saved orig-roi and term-roi parameters.
- Tobias's S-CSCF will remove the orig-roi and term-roi parameters.
- Finally, the P-CSCF will again remove the P-Charging-Vector header from the 200 (OK) response before sending it to Tobias's UE.

#### *12.7.4 Distribution of Charging Function Addresses*

The addresses of the CDF and OCF are distributed within the home networks of the users. These addresses are available at the S-CSCFs, as they are downloaded during the user's registration from the HSS (see Section 11.5.6).

Every S-CSCF adds a P-Charging-Function-Address header to the INVITE request, which is then sent along the route until it reaches the border of the home network. The CSCF at the border of the home network will remove this header.

In the case of Tobias's home network this means that his S-CSCF:

- Adds the P-Charging-Function-Address header when it first receives the INVITE request and then sends it to all ASs that belong to the home network and are contacted due to the filter criteria in Tobias's user profile (see Section 12.3.8). As the S-CSCF is the last entity in Tobias's home network, it will remove the P-Charging-Function-Address header before sending it toward Theresa's home network.
- Adds the P-Charging-Function-Address header again when it receives the 183 (Session in Progress) response, sends it once more to all ASs that belong to the home network and are contacted due to the filter criteria, and removes the header before sending the response toward the P-CSCF that is located in the visited network.

The P-CSCF in Tobias's visited network will need to discover its local charging function addresses by other means.

On the other hand, Theresa's S-CSCF:

- Adds the P-Charging-Function-Address when it receives the INVITE request and then sends it toward the ASs, as stated above. However, in this case the S-CSCF does not remove the header from the request before sending it to the P-CSCF, as Theresa's P-CSCF is located in her home network. She uses access network roaming, not IMS roaming, to access her home network's P-CSCF (see Section 10.1). The P-Charging-Function-Address header will, therefore, be removed from the INVITE request by the P-CSCF.
- Adds the P-Charging-Function-Address header again when it receives the 183 (Session in Progress) response and sends it once more with the response. However, this time the header will be removed by the I-CSCF of Theresa's home network, which will also receive the response.

The P-Charging-Function-Address header that is added by Tobias's S-CSCF includes:

- the ‘ccf’ parameter set to the address of the CDF, as received by the S-CSCF from the HSS in the ‘Primary-Charging-Collection-Function-Name’ AVP in the Server Assignment Answer (SAA) during Tobias's registration – see Section 11.5.6;
- the ‘ecf’ parameter set to the address of the OCF, as received by the S-CSCF from the HSS in the ‘Primary-Event-Charging-Function-Name’ AVP in the SAA during Tobias's registration:

```
INVITE sip:theresa@home2.hu SIP/2.0
P-Charging-Function-Addresses: ccf=[5555::a55:b44:c33:d22]
;ecf=[5555::f66:e77:d88:c77]
```

### 12.7.5 Related Standards

IMS-specific SIP headers for these charging procedures are specified in:

[RFC3455] Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the 3rd-Generation Partnership Project (3GPP).

## 12.8 Release of a Session

### 12.8.1 User-Initiated Session Release

Of course, Tobias and Theresa will at one point stop their conversation. Let us say that Theresa meets one of her friends at Stephansdom in Vienna and has to say goodbye to her brother (Figure 12.15). She will be the one who drops the call by pressing the red button on her mobile phone.

Consequently, her UE will generate a BYE request, which is sent to Tobias's UE in the same way as any other subsequent request. In parallel to this, her UE will also drop the media PDP context that was established for the session:

```
BYE sip:[5555:1:2:3:4]:1357 SIP/2.0
Route: <sip:pcscf2.home2.hu:1511;lr>
```

```

Route: <sip:scscf2.home2.hu;lr>
Route: <sip:scscf1.home1.fr;lr>
Route: <sip:pcscf1.visited1.fi;lr>
To: "Your Brother" <sip:tobi@brother.com>;tag=veli
From: "My beloved Sister" <sip:Theresa@sister.com>;tag=schwester

```

We can see from this BYE request that the information in the To and From headers has been swapped, as this request is now sent from Theresa's end.

Tobias's UE will drop its PDP context immediately after it receives the BYE request. It will also respond to the request with a 200 (OK) response, which will be sent back toward Theresa. The four CSCFs and all ASs on the route will clear all dialog states and information related to the session.

### 12.8.2 P-CSCF Performing Network-Initiated Session Release

There might be situations in which it is necessary for one of the CSCFs to release an ongoing session, rather than the user.

For example, Theresa's P-CSCF would need to release an ongoing session when it realizes that Theresa's UE has lost radio coverage and is no longer connected to the access network (Figure 12.16). In that case the P-CSCF would need to send a BYE request on behalf of Theresa:

```

BYE sip:[5555:1:2:3:4]:1357 SIP/2.0
Route: <sip:scscf2.home2.hu;lr>
Route: <sip:scscf1.home1.fr;lr>
Route: <sip:pcscf1.visited1.fi;lr>
To: "Your Brother" <sip:tobi@brother.com>;tag=veli
From: "My beloved Sister" <sip:Theresa@sister.com>;tag=schwester

```

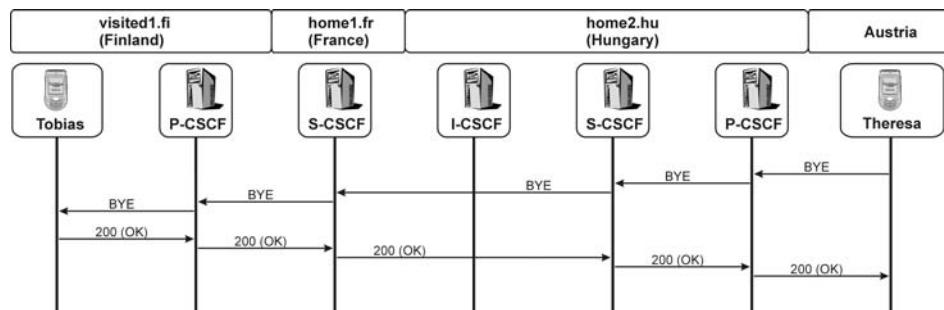


Figure 12.15 Theresa releases the session

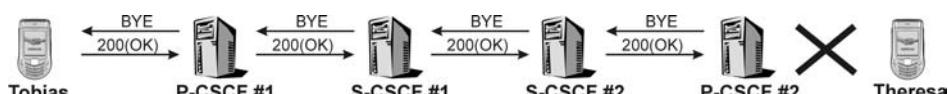
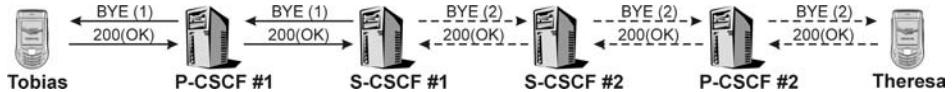


Figure 12.16 P-CSCF terminates a session



**Figure 12.17** S-CSCF terminates a session

### 12.8.3 S-CSCF Performing Network-Initiated Session Release

There may be occasions when Tobias's S-CSCF needs to be shut down or Tobias may be using a pre-paid card and runs out of money. In such cases Tobias's S-CSCF would release the session (Figure 12.17) by issuing one BYE request toward Tobias's UE:

```

BYE sip:[5555:1:2:3:4]:1357 SIP/2.0
Route: <sip:pcscf1.visited1.fi;lr>
To: "Your Brother" <sip:tobi@brother.com>;tag=veli
From: "My beloved Sister" <sip:Theresa@sister.com>;tag=schwester

```

and another BYE request toward Theresa's UE:

```

BYE sip:[5555:1:2:3:4]:1006 SIP/2.0
Route: <sip:scscf2.home2.hu;lr>
Route: <sip:pcscf2.home2.hu;lr>
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister" <sip:Theresa@sister.com>;tag=schwester

```

To generate the BYE request with the correct set of Route headers, the P-CSCFs and S-CSCFs need to keep track of all routing information that is collected during the establishment of any dialog.

## 12.9 Alternative IMS Session Establishment Procedures

### 12.9.1 Overview

Up till now we looked at a single example for an IMS session establishment call flow, which was based on certain assumptions, e.g. that several media bi-directional media streams will be established between two users and that both users need to reserve their resources during the session establishment.

In the following chapters we will look at different scenarios for session establishment under alternative circumstances, e.g. scenarios in which:

- only a uni-directional video stream needs to be established (Sections 12.9.2, 12.9.3 and 12.9.4);
- resources reservation is needed only at A-side (Sections 12.9.4 and 12.9.6);
- resource reservation is needed only at B-side (Sections 12.9.2 and 12.9.5);
- no resource reservation is needed at neither A- nor B-side (Section 12.9.7);
- the network initiates the resource reservation, instead of the user (Sections 12.9.6 and 12.9.7);
- early media is sent before the SIP session establishment procedures have been finished (Section 12.9.8);

- sessions with non-IMS SIP terminals are established (Sections 12.9.9 and 12.9.10).

These scenarios are only a subset of the possible combinations of resource reservation procedures, access network capabilities and required media capabilities – they are not complete by far. What they try to show is, how the SIP and SDP signalling mechanisms, most important of all the very powerful preconditions mechanism, allow a flexible call establishment which is kept most efficient for each of the examples. The reader should get aware, that the different message flows shown in this section do not need to be implemented separately, but are a direct result of the signalling indications, which are set by the involved phones and network elements, based on their local capabilities, the current status of resource reservation and the characteristics of the access network.

The first few examples explain especially the setting of the SDP parameters in detail, as they are essential for the understanding of the session establishment call flows. This is partly a repetition of what has been described in Sections 12.5.3 and 12.6.4, but is seen as necessary to give a clear overview how these parameters influence the signalling. In the later examples only those signalling elements specific for the individual scenario are highlighted.

### 12.9.2 Session with a Uni-Directional Media Stream and Available Resources on A Side

There are several scenarios, in which resource reservation is not needed at least on one side of the communication, for example one or both of the involved UEs:

- is connected to a network, which does not require resource reservation (e.g. a fixed broadband access);
- has pre-reserved the related resources, e.g. due to a prior session of which it did not release the related resources;
- requires/accepts only a media type, for which no resource reservation is needed (e.g. a messaging session, for which in GPRS environments usually an available background PDP context is used).

In this example (see Figure 12.18) we assume, that Tobias calls Theresa from a phone that is connected to a fixed broadband access, which does not require Tobias to reserve resources. Tobias in this example wants to establish two media sessions:

- a full-duplex audio session;
- an uni-directional video session, i.e. this video stream will only be sent to his sister.

We do not look, in this example, at how the audio session is negotiated via SDP or established on the media level. It is assumed, that the resources for both, the audio, as well as for the video session, are available at Tobias's side of the communication, but need to be reserved on Theresa's side.

Tobias's UE constructs the initial INVITE with the following SDP for the video stream (the SDP for the audio stream is not shown here):

```
m=video 3400 RTP/AVP 31
a=sendonly
```

```

a=curr:qos local send
a=curr:qos remote none
a=des:qos mandatory local send
a=des:qos none local recv
a=des:qos none remote sendrecv
a=tcap:1 RTP/AVP RTP/AVPF
a=pcfg:1 t=1|2

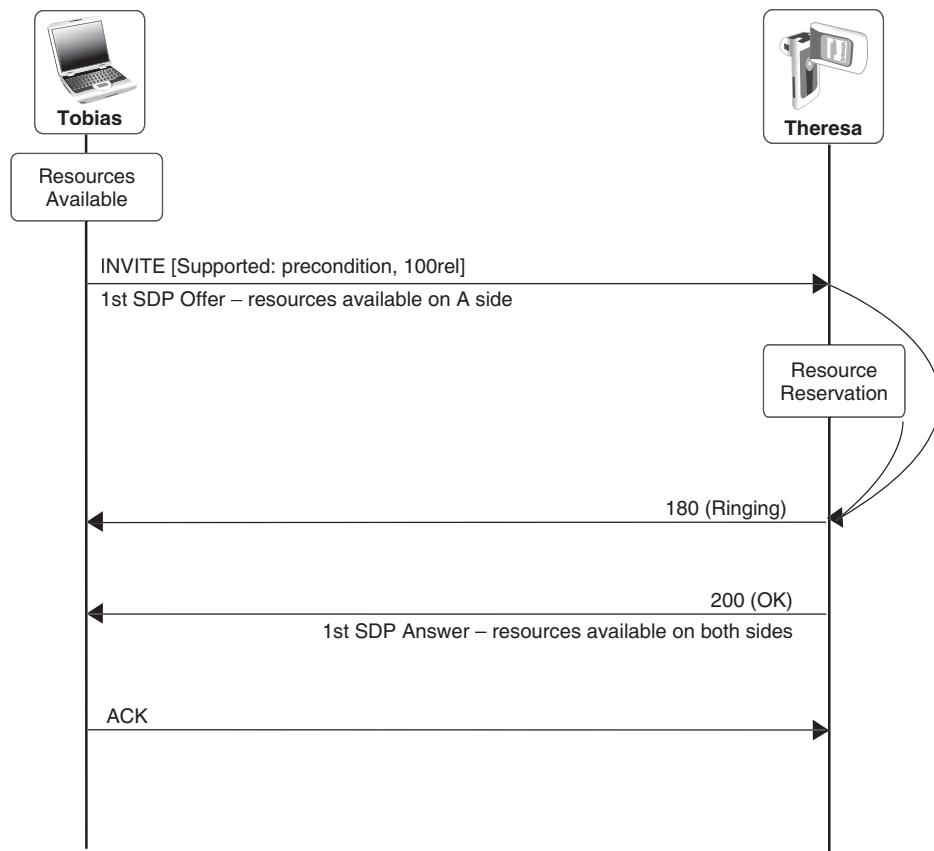
```

In this SDP Offer, Tobias's UE indicates:

```
m=video 3400 RTP/AVP 31
```

- in the first line, that it wants to establish a video media session from its local port 3400 with the RTP Audio-Visual Profile (AVP), using the H.261 codec (AVP Payload type 31, see RFC 3551);

```
a=sendonly
```



**Figure 12.18** Session establishment – resources available at A side

- in the second line, that the media stream will only be sent from Tobias's side ('sendonly'), i.e. Tobias's UE is not prepared to receive a video stream on this port;

a=curr:qos local send

in the third line, that on the local (Tobias's) side, the resources for sending the video stream are currently available, i.e. no resource reservation is needed on the A-side;

a=curr:qos remote none

- in the fourth line, that currently no information about the resource reservation status of the remote (Theresa's) UE is available and therefore it is assumed, that Theresa's UE does not have any resources available;

a=d es:qos mandatory local send

in the fifth line, that the local (Tobias's) side needs to mandatorily reserve the resources in sending direction – this condition is met by the current status indicated in the third line, therefore the local preconditions at Tobias's side are met;

a=des:qos none local recv

- in the sixth line, that the local (Tobias's) side has no need to reserve any resources in receiving direction, as the media stream is sendonly – it is important to note here, that RFC 3312 mandates, that the qos precondition requirements for both sides must always be expressed for both sending and receiving directions, therefore in this example, the A-side must also express clearly, that it does not have any requirement for resources in receiving direction;

a=des:qos none remote sendrecv

- in the seventh line, that the local (Tobias's) side does not ('none') put any requirement on the remote (Theresa's) side on which resources need to be reserved, neither in sending nor in receiving side;

a=tcap:1 RTP/AVP RTP/AVPF  
a=pcfg:1 t=1|2

- in the eighth and ninth line, that AVPF can be used for this session, but as Tobias's is not aware, whether the remote (Theresa's) UE supports AVPF as well, it uses the SDP Capability Negotiation mechanism (see Section 12.5.3.5) to allow a switchover to AVPF if Theresa's UE supports this profile.

The a-lines in this example can be indicated in any sequence, but must all be below the 'm=video' line.

With this information the SIP INVITE request is sent out and routed through the network as described in Section 12.3 and finally received at Theresa's UE.

In this example we assume, that on Theresa's side the resources for at least the video stream are not available and therefore need still to be reserved. Based on the received SDP Offer in the initial SIP INVITE request Theresa's UE gets aware, that the A-side has the required resources already available (see line 3, 5 and 6 of the received SDP Offer). Theresa's UE therefore does not need to request a confirmation from the A-side (see Section 12.6.4.2). There is also no additional codec negotiation needed, as the A-side did only indicate one possible video codec (we assume the same for the not-shown audio stream). Therefore there is no need for sending a SIP 183 (Session Progress) response with an immediate SDP Answer. Due to this, Theresa's UE can start immediately with resource reservation without any further SIP or SDP level communication.

Note, that this 'short call flow' only applies if Theresa's UE is in charge of resource reservation. If Theresa's UE got aware during Signalling PDP Context establishment (see Section 11.3) that the network will initiate resource reservation, then the procedures described in Section 12.9.5 need to be applied.

After Theresa's UE has reserved the resources, it starts ringing and sends out a SIP 180 (Ringing) response to the INVITE request, which is an indication to Tobias's UE, that resources have now been reserved on Theresa's side (see RFC 4032). Theresa's UE can include already in this SIP 180 (Ringing) response an SDP Answer – which in this case (based on RFC 3261 procedures) it does not need to send reliably, i.e. there is no subsequent SIP PRACK / 200 (OK) exchange needed. In this example we assume that Theresa's UE does not send the SDP Offer in the SIP 180 (Ringing) response.

Once Theresa accepts the call, the UE sends out a SIP 200 (OK) response to the INVITE, including the SDP Answer:

```
m=video 4011 RTP/AVP 31
a=recvonly
a=curr:qos local recv
a=curr:qos remote recv
a=des:qos mandatory local recv
a=des:qos none local send
a=des:qos mandatory remote recv
a=des:qos none remote send
```

In this SDP Answer, Theresa's UE indicates:

```
m=video 4011 RTP/AVP 31
```

- in the first line, that it will receive the H.261 encoded (AVP Payload type 31) video stream on port 4011;
- also in the first line, that it supports only the RTP Audio-Visual Profile (RFC 3551), therefore the suggested AVPF (see lines 8 and 9 of the received SDP Offer) will not be used;

```
a=recvonly
```

- in the second line, that the video stream will only be received at ('recvonly') at Theresa's UE, i.e. it will not send out a video stream from port 4011;

a=curr:qos local recv

- in the third line, that the local (Theresa's) side has reserved the resources and has them available in receiving side – note that ‘local’ here now means the side of the sender of the SDP Answer, i.e. Theresa's side;

a=curr:qos remote recv

- in the fourth line (repetition of line three in the received SDP Offer), that the remote (Tobias's) side has also reserved the related resources, as it was indicated in the third line of the received SDP Offer – not here, especially as Theresa's UE must always indicate the related media stream as a ‘recv’, i.e. as a media stream that can only be received, even when indicating it for the remote (Tobias's) side (from which the media stream will actually be sent);

a=des:qos mandatory local recv

- in the fifth line, that the local (Theresa's) side has a mandatory requirement to reserve the resources for receiving direction – this condition is met by the current status indicated in the third line, therefore the local preconditions at Theresa's side are met;

a=des:qos none local send

- in the sixth line, that the local (Theresa's) side has no requirement to reserve resources for sending direction – this needs to be indicated, as RFC 3312 mandates that each side indicates the requirements for local resource reservation for both directions, only by indicating this, the remote (Tobias) can check from the desired (‘des’) and current (‘curr’) precondition statuses, whether Theresa's UE has reserved all required resources;

a=des:qos mandatory remote recv

- in the seventh line (repetition of line five in the received SDP Offer), that the remote (Tobias's) side has a mandatory requirement to reserve the resources – note again (as in the fourth line), that Theresa's side indicates the video stream always as receiving (‘recv’) stream, although Tobias's side has in fact reserved its local resources in sending (‘send’) side – this condition is met by the current status indicated in the fourth line, therefore also the local preconditions at Tobias's side are met;

a=des:qos none remote send

- in the eighth line (repetition of line six in the received SDP Offer), that the remote (Tobias's) side has no requirement to reserve resources in sending side, as this video stream is (from Theresa's UEs perspective) a receive-only stream – again note, that the sending/receiving indications have changed here in the same way as in the seventh and fourth line.

### 12.9.3 Session with a Uni-Directional Media Stream and Resources Need to be Reserved on A and B Side

In the above example it was assumed that Tobias's side has the resources for the sendonly video stream already available. This example now shows the procedures if the related resources are not available on the A-side. It is assumed that only the resources for the video stream need to be reserved.

The SDP Offer, that is sent by Tobias's UE in the initial SIP INVITE requests looks similar to the one shown in the example before, i.e. like this:

```
m=video 3400 RTP/AVP 31
a=inactive
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local send
a=des:qos none local recv
a=des:qos none remote sendrecv
a=tcap:1 RTP/AVP RTP/AVPF
a=pcfg:1 t=1|2
```

The differences to the initial SDP Offer shown in Section 12.9.2 are:

`a=inactive`

- in line 2, as the resources have not yet been reserved on Tobias's side, the stream must be indicated as ‘inactive’, this is due to interworking procedures with non 3GPP IMS compliant SIP terminals (see Sections 12.9.9 and 12.9.10);

`a=curr:qos local none`

- in line 3, as the resources have not yet been reserved on Tobias's side, the current status of qos preconditions is indicated as ‘none’.

As only one codec option (H.261 – RTP Profile Type 31) is offered in the m-line, no further codec negotiation for this video stream is to be expected. Therefore Tobias's UE is allowed to start resource reservation immediately after sending this initial SDP Offer in the SIP INVITE request.

Theresa's UE now needs to request a confirmation from the A-side, in order to be made aware once the resources have been successfully reserved by Tobias's UE. It needs this confirmation (as described in Section 12.6.4.2) to be able to start ringing only in the case when both sides have successfully reserved their resources. In order to request this confirmation, Theresa's UE sends back a SIP 183 (Session Progress) response, including the below shown SDP answer. As the SIP 183 (Session Progress) response is an provisional (1xx) response which carries SDP information, it needs to be sent reliably. This is achieved by using the procedures of RFC 3162 as described in Section 12.5.2, i.e. by adding the following two headers to the SIP 183 (Session Progress) response:

SIP/2.0 183 Session Progress

Require: 100rel  
RSeq: 93

The SDP Answer sent within this SIP 183 (Session Progress) response looks as follows:

```
m=video 4011 RTP/AVPF 31
a=inactive
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local recv
a=des:qos none local send
a=des:qos mandatory remote recv
a=des:qos none remote send
a=conf:qos remote recv
a=acfg:1 t=2
```

The following changes are different from those given in the first SDP Offer in the example shown in Section 11.10.2:

```
m=video 4011 RTP/AVPF 31
a=acfg:1 t=2
```

- line one and ten, in which Theresa's UE indicates that it supports AVPF – this switchover from AVP to AVPF is done based on the SDP Capability Negotiation Framework (see Section 12.5.3.5);

a=inactive

- line two, as the stream was indicated 'inactive' in the received SDP Offer and Theresa's UE does not have the resources yet available, it must return the 'inactive' flag as received – this is due to interworking procedures with non 3GPP IMS compliant SIP terminals (see Sections 12.9.9 and 12.9.10);

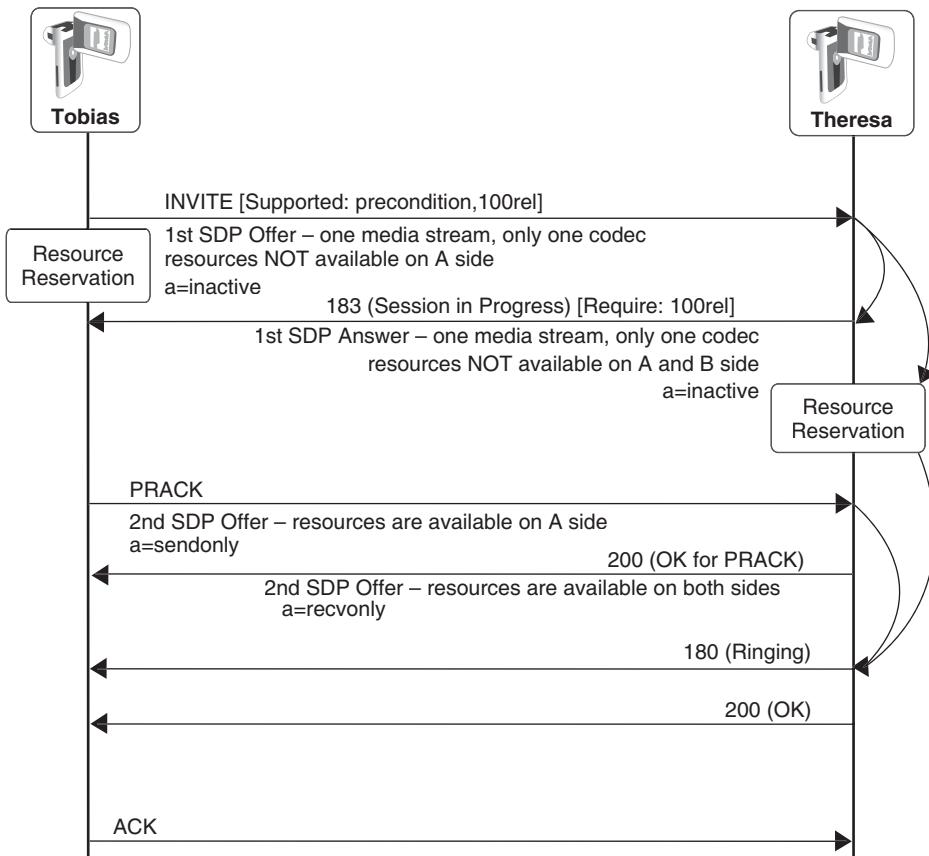
a=curr:qos local none

- line three, which indicates that the local (Theresa's) side has currently not reserved any resources;

a=conf:qos remote recv

- line nine, in which Theresa's UE requests a confirmation ('conf') from the remote (Tobias's) side, once the remote side has reserved its resources in receiving ('recv') direction – note here once more that Theresa's UE always refers to the video stream as a receiving stream, although from Tobias's UE point of view this stream is a send-only.

Immediately after sending this SDP Answer in the SIP 183 (Session Progress) response, Theresa's phone starts to reserve the required resources for receiving the H.261 video stream. Theresa's phone is aware that it only needs to reserve resources in receiving direction, as the received SDP Offer (in the SIP INVITE request) indicated, that Tobias's side will reserve resources only in a sending direction.



**Figure 12.19** Session establishment – uni-directional stream with resource reservation on both sides

In this example it is assumed (Figure 12.19), that the resource reservation on Tobias's side has finished already when the SIP 183 (Session Progress) response is received. In this case, Tobias's UE can avoid sending the SIP UPDATE request and can use the SIP PRACK request to immediately confirm its successful resource reservation.

Tobias's UE immediately sends out a SIP PRACK request in order to confirm the receipt of the reliably sent SIP 183 (Session Progress) response (see Section 12.5.2) and includes the following second SDP Offer:

```
m=video 3400 RTP/AVPF 31
a=sendonly
a=curr:qos local send
a=curr:qos remote none
a=des:qos mandatory local send
a=des:qos none local recv
a=des:qos mandatory remote send
a=des:qos none remote recv
```

This second SDP Offer is practically the same as the first SDP Offer shown in Section 11.10.2. It just no longer includes the SDP Capability Negotiation Framework lines for AVP/AVPF negotiation, as this has taken place already during the first SDP Offer/Answer exchange. Note, the resources are available now at Tobias's side and therefore the ‘inactive’ flag is no longer needed and is replaced with the ‘sendonly’ indication (see Sections 12.9.9 and 12.9.10).

When receiving the second SDP Offer in the PRACK request, Theresa's phone has also finished resource reservation. Theresa's UE immediately sends back a second SDP Answer in the SIP 200 (OK) response to the received SIP PRACK request:

```
m=video 4011 RTP/AVPF 31
a=recvonly
a=curr:qos local recv
a=curr:qos remote recv
a=des:qos mandatory local recv
a=des:qos none local send
a=des:qos mandatory remote recv
a=des:qos none remote send
```

Also, this SDP Answer is practically identical to the one shown in Section 11.10.2, besides the AVPF indication.

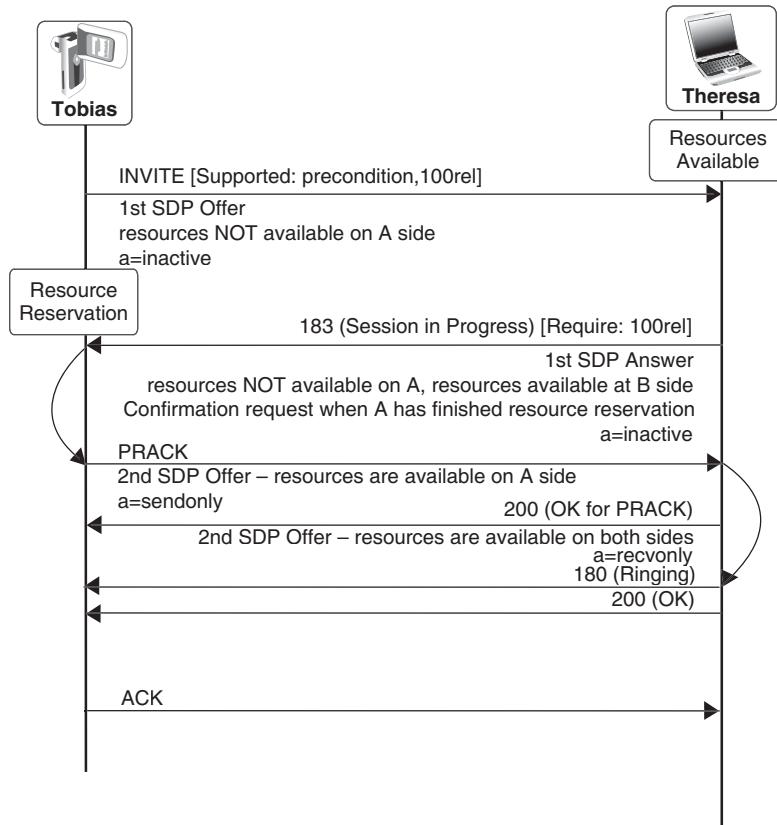
As Theresa's UE has finished resource reservation it can already start ringing, as it has got the requested confirmation from Tobias's side in the second SDP Offer, that was received in the SIP PRACK request. After this, normal SIP session procedures are applied, i.e. the SIP 180 (Ringing) and SIP 200 (OK) responses are sent from Theresa's side and Tobias's UE sends the SIP ACK response.

#### *12.9.4 Resources Available on B Side Only*

In the example given before we assumed that on both sides of the communication the resources needed to be reserved. In this example, as shown in Figure 12.20, we look at a similar call flow, with the following assumptions:

- the resources only need to be reserved at the A-side, i.e. Theresa's UE has the required resources already available;
- the media is a uni-directional video stream for which only one codec is indicated, which allows the A-side to start resource reservation immediately after sending the initial INVITE request;
- resource reservation at the A-side finishes before the SIP 183 (Session Progress) response is received;
- Theresa's UE (B-side) supports AVPF.

In this scenario, the initial SDP Offer, which Tobias's UE sends in the initial INVITE request, looks the same as in the example given in Section 12.9.3, i.e.:



**Figure 12.20** Session establishment – resources available at B side

```
m=video 3400 RTP/AVP 31
a=inactive
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local send
a=des:qos none local recv
a=des:qos none remote sendrecv
a=tcap:1 RTP/AVP RTP/AVPF
a=pcfg:1 t=1|2
```

Theresa's UE again needs to send out the SDP Answer immediately, in order to request confirmation of successful resource reservation from the A-side, as Theresa's phone will only start ringing when the resource reservation has finished on both sides. Therefore a SIP 183 (Session Progress) Response is sent, including the '100rel' option tag in the

Require header as well as an RSeq header in order to send the provisional response reliably. The following SDP Answer I transported by the response:

```
m=video 4011 RTP/AVPF 31
a=inactive
a=curr:qos local recv
a=curr:qos remote none
a=des:qos mandatory local recv
a=des:qos none local send
a=des:qos mandatory remote recv
a=des:qos none remote send
a=conf:qos remote recv
a=acfg:1 t=2
```

The only difference to the flow in Section 12.9.3 is in the third line, in which Theresa's UE indicates that it has currently its own (local) resources already available, i.e. the preconditions on Theresa's side are met. Still in the second line the 'inactive' flag is indicated, as this was received from the A-side. Theresa's phone assumes the media stream to be inactive, until it gets different information from Tobias's side. The main reason why this SDP Answer is sent, is to request confirmation of successful resource reservation ('conf' – second to last line) from the remote side.

As Tobias's side has already reserved the resources when the SIP 183 (Session Progress) response is received, it will confirm the resource reservation immediately in an SDP Answer in the SIP PRACK request and Theresa's phone will send the related SDP Answer in the SIP 200 (OK) response. From here on the procedures and protocol messages are nearly identical to those described in Section 12.9.3.

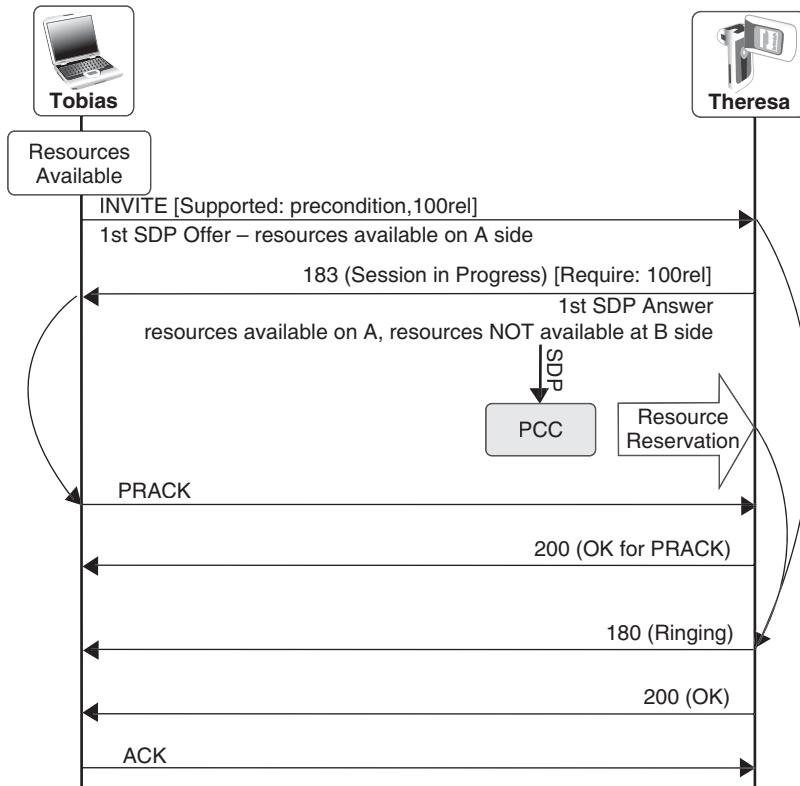
### *12.9.5 Network Initiated Resource Reservation, Resources Available Only at A-Side*

In the examples given so far we have looked only at cases in which the resource reservation is either not necessary to be performed on one side (due to the UE having the related resources already available) or in which the UE initiated the resource reservation, e.g. by sending a PDP Context Establishment request for the related media stream to the network.

With the introduction of PCC in 3GPP Rel-7 IMS (see Section 3.10) there is also the possibility that the network takes control of resource reservation. In this case, the network indicates to the UE during the Signalling PDP Context establishment procedures (see Section 12.12), that network initiated resource reservation procedures have to be applied.

In this example (see Figure 12.21) we assume that:

- Tobias's phone is connected to the IMS via a fixed-broadband network, i.e. it does not need to reserve its resources;
- Theresa's phone is connected to the IMS via a GPRS network in which PCC is used;
- Theresa's phone got an indication from the network, during Signalling PDP Context establishment, that network initiated resource reservation procedures will be applied, i.e. Theresa's phone will not attempt to initiate resource reservation on its own;



**Figure 12.21** Session establishment – network initiated resources at B side

- Tobias wants to establish an audio session with his sister;
- both phones support AVPF.

Tobias's phone constructs the initial INVITE request and includes an SDP Offer as follows:

```
m=audio 3458 RTP/AVP 0 96 97 98
a=rtpmap:96 G726-32/8000
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
a=tcap:1 RTP/AVP RTP/AVPF
a=pcfg:1 t=1|2
```

This SDP Offer expresses (in short):

```
m=audio 3458 RTP/AVP 0 96 97 98
a=rtpmap:96 G726-32/8000
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
```

- Tobias wants to establish an audio session with his sister, for which his phone will send and receive media via its local port 3458. The phone offers three different codecs for the session (fixed Audio Video Profiles 0: PCMU, as well as G.726 and AMR-WB) as well as the telephone-event extension for the transport of DTMF tones (for further explanation of these lines see Sections 12.5.3.2 and 12.5.3.3);

a=curr:qos local sendrecv

```
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
```

- Tobias's phone has the local resources available, therefore the preconditions on the local side are met, as indicated by the current ('curr') and desired ('des') lines for the local side, which are both set to the same value ('sendrecv'). As currently there is no information about the resource situation on the remote (Theresa's) side, Tobias's phone does not set any qos precondition requirements for the remote side ('none' in the last desired line) and indicates that to its knowledge, resources are not available on the remote side ('none' indication in the second current-line) (for further explanation of these lines see Section 11.6.4);
- there is no 'a=inactive' indication in this SDP Offer, as Tobias's phone has its resources available and therefore, when interworking with non-3GPP IMS SIP clients, the remote side does not need to be set 'on hold' (see Section 12.9.9);

```
a=tcap:1 RTP/AVP RTP/AVPF
a=pcfg:1 t=1|2
```

- Tobias's UE supports AVPF and uses the SDP Capability Negotiation Framework to allow a switchover to AVPF in case the remote (Theresa's) side supports AVPF as well (see Section 12.5.3.5).

After Theresa's UE receives this SDP Offer, it could immediately start to reserve the related resources. But in the given example, Theresa's phone is aware, that the network will have initiated resource reservation. In this case, the network does not know which resources to reserve towards Theresa's UE, as the received SDP Offer indicated three different audio codecs, of which only one will be used. In the scenarios given so far, Theresa's phone would immediately select the final codec (e.g. AMR-WB) and would request the related resources by sending a PDP Context Establishment request to the network, indicating in the TFT the resource requirements for AMR-WB.

As it is now the network, that has to initiate the resources towards Theresa's phone, the UE must indicate to the network which codec got selected, in order to allow the network to

reserve the appropriate resources. Therefore Theresa's phone, immediately after receiving the initial SIP INVITE request, sends a SIP 183 (Session Progress) response with an SDP Answer. The response needs to be sent reliably, as it carries SDP information.

The SDP Answer includes the following information:

```
m=audio 4011 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local none
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=acfg:1 t=2
```

This SDP Answer expresses, in short:

```
m=audio 4011 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
```

- Theresa's phone has performed final codec selection and chose AMR-WB. It also supports the telephone-event for transport of DTMF tones (for further explanation of these lines see Sections 12.5.3.2 and 12.5.3.3);

```
a=curr:qos local none
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

- Theresa's phone does currently not have resources available (first current-line set to 'none'), but mandatorily requires to reserve them in sending and receiving direction (first desired-line set to 'mandatory' and 'sendrecv'). Theresa's phone is aware that the remote (Tobias's) side has already reserved its resources (see Section 12.6.4);
- there is no need to request confirmation of successful resource reservation from the remote (Tobias's) side, as from the preconditions in the received SDP Offer it is clear, that Tobias's side has the resources already available. Therefore no 'conf' line is sent in this SDP Answer, i.e. the SDP Answer is not sent in order to trigger any events in the remote (Tobias's) UE, but only to inform the network about the result of the final codec selection;

```
a=acfg:1 t=2
```

- Theresa's phone indicates that it supports and uses AVPF for this session (see also the 'AVPF' indication in the m-line).

Once Theresa's network (P-CSCF) has received this SDP Answer, it will indicate the required resources to the PCC elements, which then can trigger resource reservation towards Theresa's phone.

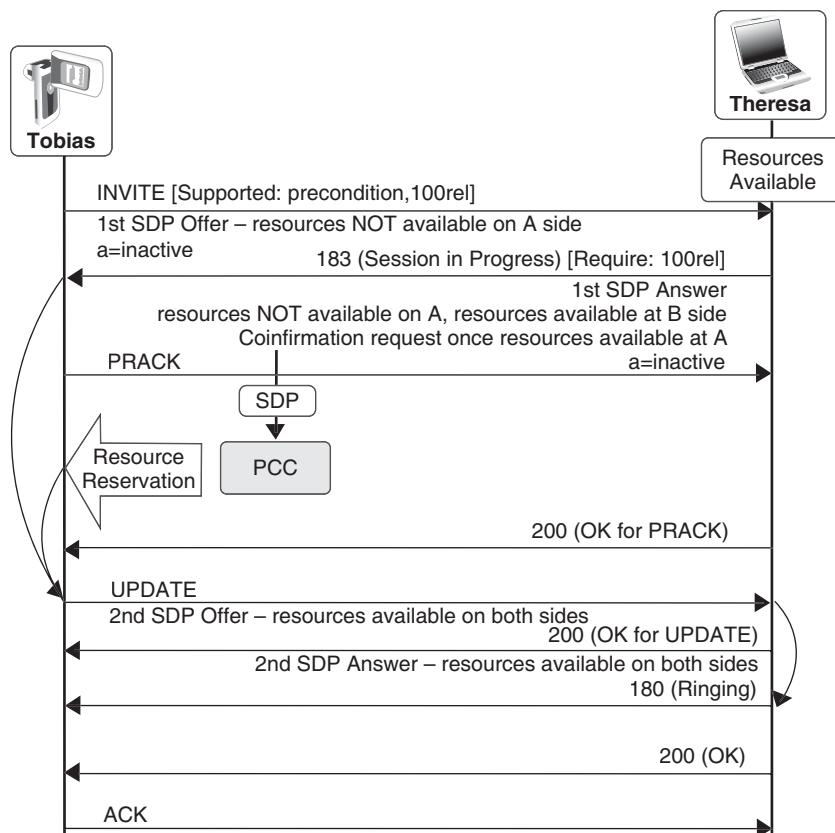
The SIP 183 (Session Progress) response is sent to Tobias's phone, which must reply to it with a SIP PRACK request, as the response was sent reliably. The PRACK request in this example does not include any SDP related information, as the status of the resource has not been changed at Tobias's phone since it sent out the SIP INVITE request.

After this, the session establishment follows the normal procedures as outlined before in other chapters. Theresa's phone starts to ring right after successful resource reservation, it is aware that resources are then available on both sides and after Theresa accepted the call, the AMR-WB encoded audio stream will be exchanged between the two phones.

#### 12.9.6 Network Initiated Resource Reservation at A Side

In the above scenario we have seen how network initiated resource reservation works on the called users side. This example (see Figure 12.22) shows how the network initiates resource reservation at the calling side. For this example it is assumed that:

- Tobias calls Theresa for an audio session, for which Tobias's phone offers several possible codecs;



**Figure 12.22** Session establishment – network initiated resources at A side

- Tobias's phone is connected to a GPRS access network, which indicated during Signalling PDP Context establishment procedures, that all media resources will be reserved by the network;
- for simplicity, Theresa's network is connected to a fixed-broadband network, i.e. it does not need to reserve any resources.

When sending out the SIP INVITE request, Tobias's phone does not have any resources available. It is aware that it cannot initiate the resource reservation on its own, but rather has to wait for the network to get active. Therefore it includes the following SDP Offer in the SIP INVITE request:

```
m=audio 3458 RTP/AVP 0 96 97 98
a=rtpmap:96 G726-32/8000
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=inactive
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
a=tcap:1 RTP/AVP RTP/AVPF
a=pcfg:1 t=1|2
```

For this scenario, the highlighted line is important, as it indicates that the current and the desired local preconditions are not met, i.e. that currently the resources on Tobias's side have not been reserved.

Tobias's P-CSCF and PCC elements, after receiving this SIP INVITE request cannot yet start with the resource reservation, as the SDP Offer includes several different codecs, of which only one will be selected in the end. Therefore Tobias's P-CSCF just forwards the SIP INVITE request towards the remote side.

Theresa's phone, upon receiving the SIP INVITE request, needs to request confirmation from Tobias's UE in order to become aware once the resources have been reserved at the remote side. It therefore sends out immediately an SDP Answer in a SIP 183 (Session Progress) response:

```
m=audio 4011 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=inactive
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=conf:qos remote recv
a=acf:1 t=2
```

Upon receiving this SDP Answer, Tobias's P-CSCF and PCC elements become aware that Theresa's phone has selected the final codec for the audio session and based on

this, resource reservation can be initiated. The SIP 183 (Session Progress) response is forwarded to Tobias's UE and the media PDP Context is getting established in parallel.

Tobias's phone needs to immediately confirm the receipt of the SIP 183 (Session Progress) response by sending a SIP PRACK request back to Theresa's phone. As at this moment, the media resources are not yet available at Tobias's side, the phone does not need to include an additional SDP Offer in the SIP PRACK request.

Once the media resources have successfully been reserved at Tobias's side, the phone needs to confirm this to Theresa's UE, as the SDP Answer in the received SIP 183 (Session Progress) response included a confirmation ('conf') line. Therefore Tobias's UE sends an SIP UPDATE request with the following SDP Offer:

```
m=audio 3458 RTP/AVP 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
```

Theresa's phone now can start ringing and in parallel responds to the received second SDP Offer with an SDP Answer, which it sends in the SIP 200 (OK) response to the SIP UPDATE request:

```
m=audio 4011 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
```

In parallel Theresa's phone sends a SIP 180 (Ringing) response. From here on the session establishment procedures are identical to those outlined in the main part of Chapter 11.

### 12.9.7 Resources Available on A Side and B Side

Figure 12.23 shows the example session establishment flow, when both phones have their resources already available before the SIP INVITE request is sent or received, e.g. both UEs are connected to a fixed-broadband access network.

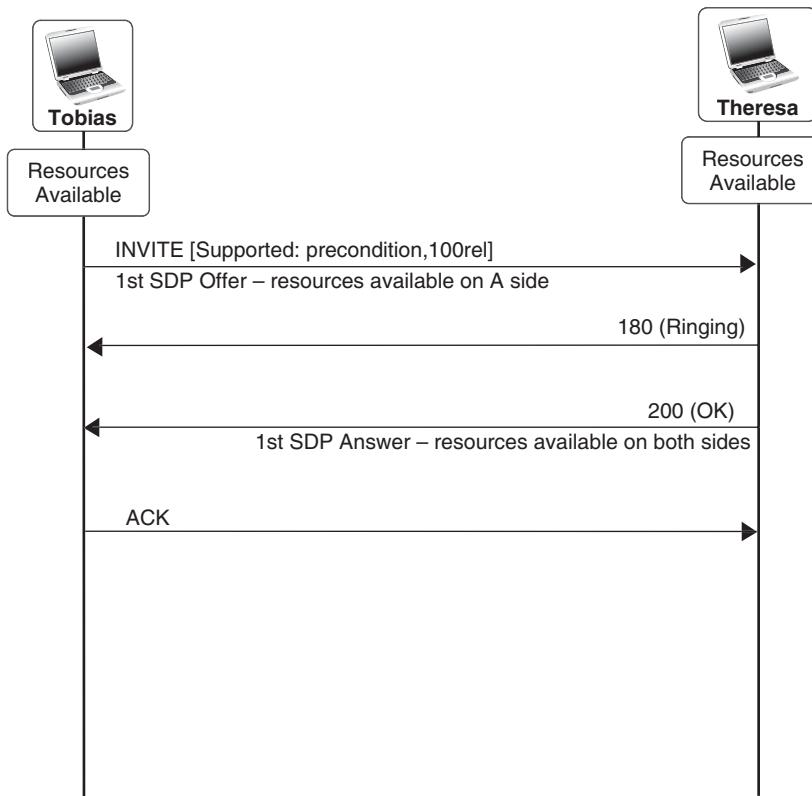
In this case, Tobias's UE sends out a SIP INVITE request, including an SDP Offer that indicates that resources are already available at the originating side, which looks, for example, as follows:

```
m=audio 3458 RTP/AVP 0 96 97 98
a=rtpmap:96 G726-32/8000
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local sendrecv
a=curr:qos remote none
```

```
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
a=tcap:1 RTP/AVP RTP/AVPF
a=pcfg:1 t=1|2
```

Due to this SDP Offer and the availability of resources at Theresa's side, the IMS session setup in this case gets shortened down:

- as Tobias's side indicated already in the initial SDP Offer, that resources are available, by setting the preconditions to be met, Theresa's phone does not send out a SIP 183 (Session Progress) response, asking for confirmation for successful resource reservation;
- as the SIP 183 (Session Progress) response is not sent, there is no need for the SIP PRACK request (and the subsequent SIP 200 (OK) response) to be sent. The SIP PRACK request is only needed when a provisional response needs to be sent reliably;
- as no confirmation is required anymore, i.e. no 'conf'-line in an SDP Answer is sent from Theresa's phone to Tobias, there is also no SIP UPDATE request (and its subsequent SIP 200 (OK) response) sent.



**Figure 12.23** Session establishment – resources available on both sides

Theresa's phone therefore immediately starts ringing and sends out a SIP 180 (Ringing) response. On Tobias's side the receipt of the SIP 180 (Ringing) response is an indication that resources are available on both sides. Once Theresa accepts the call, her UE sends back a SIP 200 (OK) response, indicating the following SDP Answer:

```
m=audio 4011 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=acfg:1 t=2
```

In this SDP Answer Theresa's phone indicates:

```
m=audio 4011 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
```

- the result of final codec selection (AMR-WB is used);

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

- that resources are available on both sides as all preconditions have been met;

```
m=audio 4011 RTP/AVPF 97 98
a=acfg:1 t=2
```

- that AVPF is supported and used for this session.

Here it becomes obvious, that if both sides have their resources available, the IMS session setup procedures are very efficient, as the resource reservation does not need to be initiated, negotiated and performed during the session setup, as in the other scenarios that have been shown so far. The flexibility, that allows the session establishment procedures to be performed adequately based on the resource reservation situation is given mostly by the preconditions mechanism.

#### *12.9.8 Early Media and Reliable Ring-Back Tone*

During some session establishments, it is desirable to allow the transport of media already prior to successful session establishment, i.e. before the remote side has accepted the call. This can for example be required:

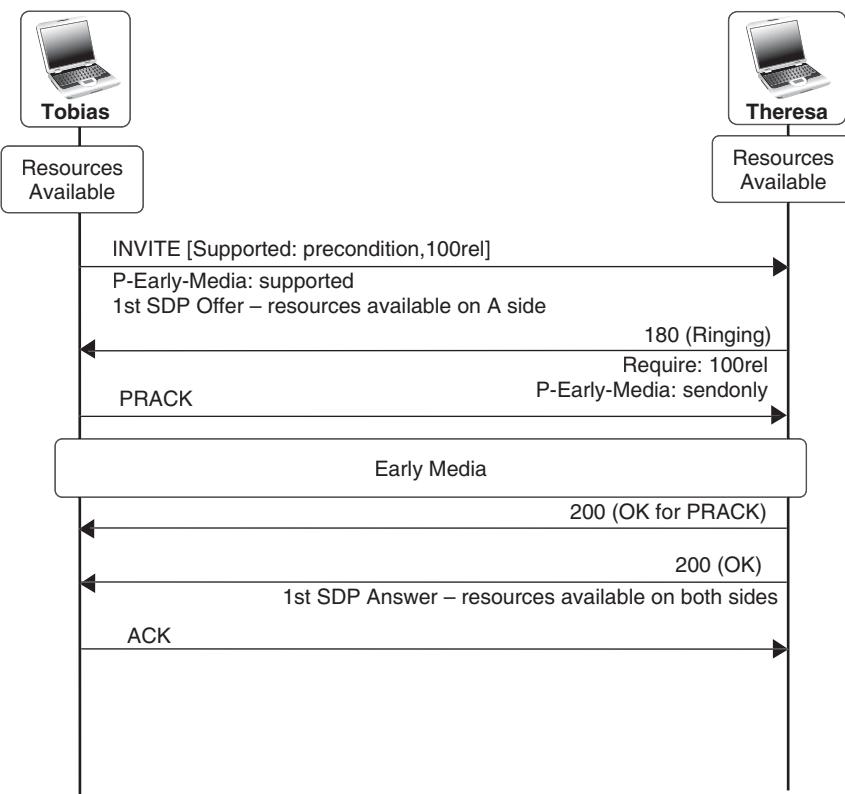
- if the network wants to send an announcement to the user, e.g. ‘The dialed address is incorrect’;

- if the called side wants to send a specific ringing tone to the calling side, e.g. a call centre that sends an advertisement announcement from its office PBX to the calling user, whilst waiting for the next agent to become free.

In the second case it becomes necessary, that the called side gets an reliable indication, that the calling side is aware of the ringing. This scenario is called ‘reliable ring-back tone’.

In this example (see Figure 12.24) we assume:

- that Theresa wants to send a special ringing tone to Tobias, which is configured at Theresa’s phone. This ringing tone needs to be sent via the media connection already before Theresa accepts the call (early media);
- both phones are connected to a fixed-broadband access network and do not need to perform resource reservation procedures (the SIP/SDP signaling for such scenarios is shown in Section 12.9.7);
- the IMS network provider wants to charge the media stream, that is exchanged between the two users, therefore additional information is needed in the exchanged SIP messages.



**Figure 12.24** Session establishment – early media and ringback tones

When sending the SIP INVITE request, Tobias's phone indicates, that it supports early media in sending and in receiving direction, by including an indication, that it supports early media, as defined by RFC 5009:

```
INVITE sip:theresa@home2.hu SIP/2.0
P-Early-Media: supported
```

After receiving this indication, Theresa's phone decides to make use of early media, in order to send the special ringing tone prior to successful call establishment. As Theresa's phone does not need to reserve resources and got aware, that resources are also available on Tobias's side, it immediately starts to ring and sends out a SIP 180 (Ringing) response. This response must be sent reliably due to two reasons:

- an SDP Answer must be included, in order to allow the network to charge the early media that is used. The SDP Answer is identical to the one sent in the 200 (OK) response in Section 12.9.6;
- Tobias's phone must get aware of the fact, that Theresa's phone is ringing (reliable ring-back tone).

Theresa's phone also includes the P-Early-Media header set to 'sendonly', indicating that it will use early media only to send the ringing tone to Tobias's side.

```
SIP/2.0 180 Ringing
Require: 100rel
RSeq: 70
P-Early-Media: sendonly
```

The P-Early-Media header is used by the CSCFs and other IMS network entities to become aware, that early media is being sent and therefore the related media authorization and charging mechanism will be applied (see Sections 12.6.5 and 12.7).

When receiving the SIP 180 (Ringing) response, Tobias's UE will not provide any locally generated ring-back tone to Tobias, but will wait for the early media to be received from the remote side. In addition to this, it must send back a SIP PRACK request, in order to confirm the receipt of the SIP 180 (Ringing) response, which was sent reliably. Based on receipt of the SIP PRACK request, Theresa's phone becomes aware, that on the remote (Tobias's) side, the ring-back indication can be provided.

The media session is cut-through in the network once the SIP 180 (Ringing) response was received by the related P-CSCFs. Due to that, Tobias will receive the ringing indication, which is sent from Theresa's phone, earliest after receiving the SIP 180 (Ringing) response. Due to that, Theresa's phone should not send out any early media, before it did not receive the SIP PRACK request, acknowledging the receipt of the SIP 180 (Ringing) response at Tobias's side.

### 12.9.9 Session Towards a Non-IMS SIP Terminal

IMS is not a closed system, it was designed to allow flexible multimedia communication towards all sorts of telecommunications systems. For communication with networks that use different signalling protocols than SIP and SDP, interworking is required (see Section

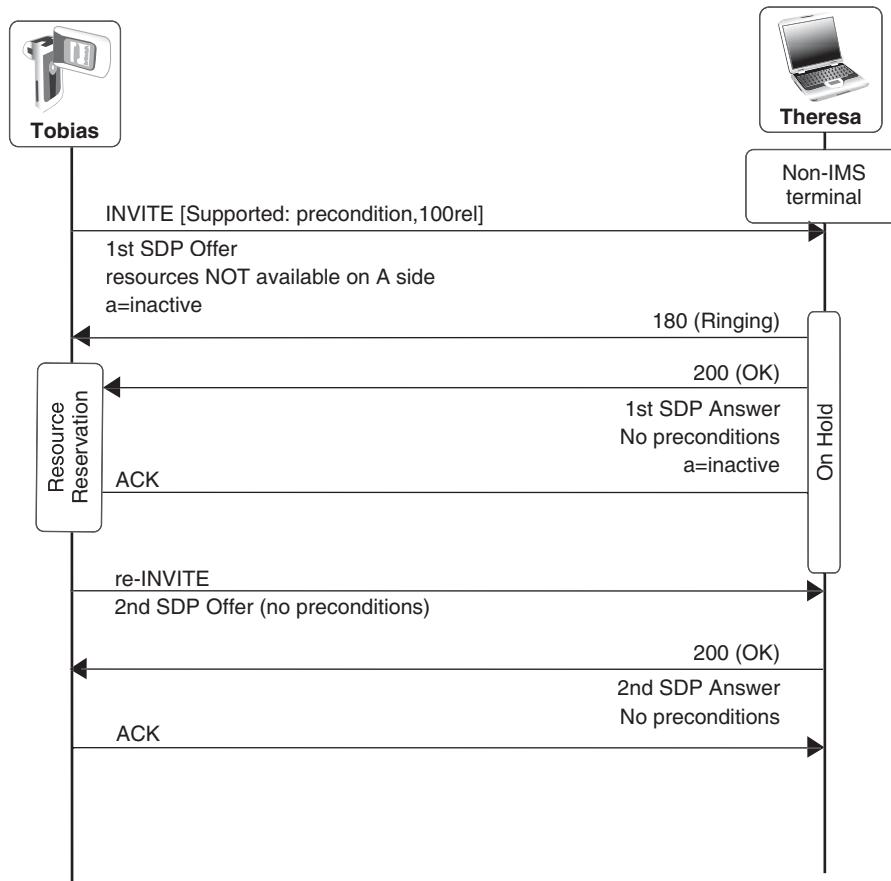
3.14). When it comes to communication with non-IMS SIP terminals, the flexibility of the SIP and SDP protocols allows it, that no interworking needs to be performed, as long as certain fall-back mechanisms are supported by the IMS terminals involved in the call.

We have seen before, that the IMS client is required to support many SIP and SDP extensions, which are not necessarily supported by non-IMS clients. Some of these extensions are:

- the SDP preconditions for negotiation of resource reservation (see Section 12.6.4);
- the SIP PRACK request (see Section 12.5.2);
- the SIP UPDATE request (see Section 12.6.1).

In this example (see Figure 12.25) we look at the following scenario:

- Tobias is using an IMS phone and calls his sister Theresa for an audio session;
- Tobias is connected to a GPRS network and has not reserved the local resources;



**Figure 12.25** Session establishment towards a non-IMS terminal

- Theresa is using a nonIMS SIP client on her PC desktop at work, which does **not** support:
  - the SDP precondition mechanism;
  - the SIP PRACK method and the related ‘100rel’ mechanism;
  - the SIP UPDATE method;
  - AMR-WB and G.726 audio codecs;
  - the telephone-event for transporting DTMF tones;
  - the SDP Capability Negotiation Framework;
  - the RTP Audio-Video Profile with Feedback (AVPF).

As Tobias’s phone is not aware of the capabilities of Theresa’s phone, it will construct a SIP INVITE request with the default content. For this scenario it is important that the UE does not include a Require header with the option tag ‘precondition’ within the SIP INVITE request, but only a Supported header, i.e.

```
INVITE sip:theresa@home2.hu SIP/2.0
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister" <sip:theresa@home2.hu>
Supported: precondition, 100rel
CSeq: 1112 INVITE
Call-ID: apb03a0s09dkjdfglkj49555
```

If the ‘precondition’ option tag would be included in a Require header additionally, Theresa’s UE would reject the SIP INVITE request with a SIP 420 (Bad Extension) response, as it does not support the SDP preconditions mechanism. In order to avoid such a rejection, which would cause an additional roundtrip and therefore a delay in session establishment, within IMS the ‘precondition’ option tag is restricted to be only included in the Supported header, but never the Require header within an INVITE request. This means that Theresa’s phone will start processing the received SIP and SDP messages and will disregard (i.e. discard) the included SDP a-lines, which it does not understand.

Therefore, the SDP Offer, included by Tobias’s UE in the SIP INVITE request, includes the following lines:

```
m=audio 3458 RTP/AVP 0 96 97 98
a=rtpmap:96 G726-32/8000
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=inactive
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
a=tcap:1 RTP/AVP RTP/AVPF
a=pcfg:1 t=1|2
```

But as Theresa’s phone does not support the above mentioned SIP and SDP extensions, it only understands a subset of the received SDP Offer, i.e. it only understands the following lines:

---

```
m=audio 3458 RTP/AVP 0 96 97 98
a=rtpmap:96 G726-32/8000
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=inactive
```

At this moment, no resources are available at Tobias's side which means, that Tobias's phone is not capable yet of receiving any media. Due to that, Theresa's UE needs to be made aware that it must not send any media right away. This is achieved by the 'inactive' indication, which causes Theresa's phone to be put on hold. This means that Theresa's phone will ring nevertheless immediately and send out a SIP 180 (Ringing) response, but once Theresa accepts the call, the phone is assumed to give an indication to Theresa, that the call is currently put on hold, i.e. Theresa should wait until Tobias is answering the call.

When Theresa accepts the call, the called UE sends out a SIP 200 (OK) response which includes the first SDP Answer, which looks as follows:

```
m=audio 4011 RTP/AVP 0
a=inactive
```

Theresa's UE has selected the PCMU codec (RTP AVP Payload Type '0' – see RFC 3551) and returns the 'inactive' indication, as this was received from the calling UE.

Once Tobias's UE receives the SIP 200 (OK) response with this SDP Answer, it can start to reserve the required resources. After resource reservation has finished, Tobias's UE needs to set the media stream to active, in order to make Theresa's phone aware, that the audio stream is no longer on hold. In order to do that, Tobias's UE sends out another INVITE request, which is sent on the same dialog as the first INVITE request. This second INVITE request is called a re-INVITE request and is routed as a subsequent request (see Section 12.3.6).

```
INVITE sip:[5555::5:6:7:8]:1006 SIP/2.0
Route: <sip:pcscf1.visited1.fi:7531;lr>
Route: <sip:scscf1.home1.fr;lr>
Route: <sip:scscf2.home2.hu;lr>
Route: <sip:pcscf2.home2.hu;lr>
From: "Your Brother" <sip:tobi@brother.com>;tag=veli
To: "My beloved Sister" <sip:theresa@home2.hu>;tag=schwester
Supported: precondition, 100rel
CSeq: 1113 INVITE
Call-ID: apb03a0s09dkjdfglkj49555
```

In this subsequent SIP INVITE request, Tobias's UE only needs to set the media streams, that before were indicated as 'inactive' to 'active'. It does this, by simply removing the 'inactive' indication, as 'active' is the default status of a media stream. Therefore the SDP Offer sent in this subsequent SIP INVITE request indicates only a single line for the audio stream:

```
m=audio 3458 RTP/AVP 0
```

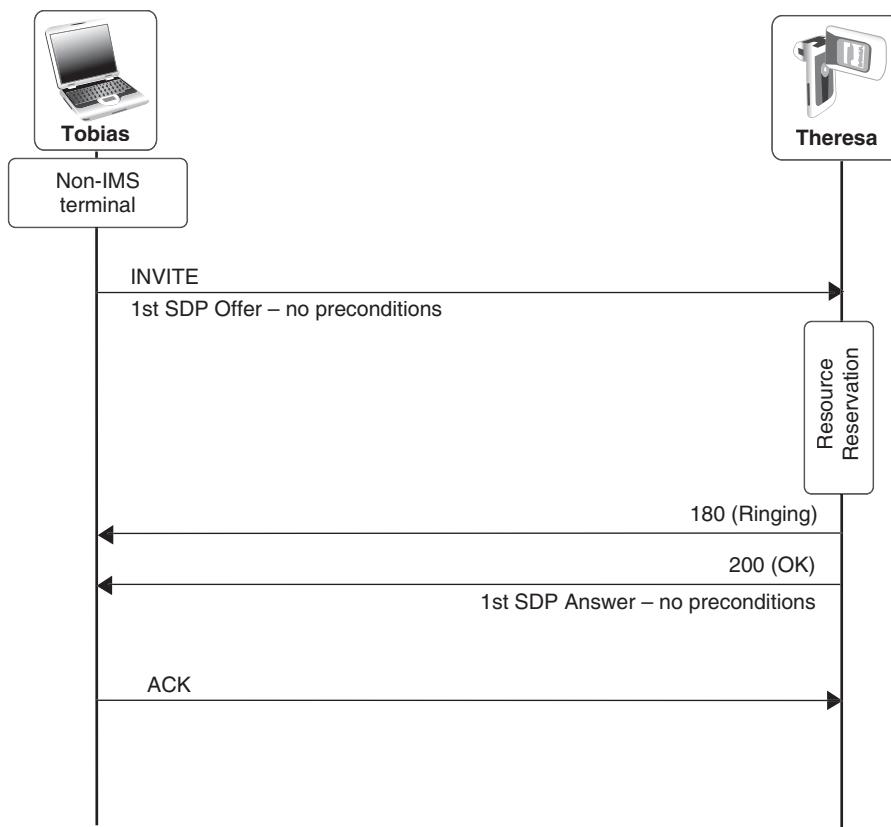
When receiving this, Theresa's phone indicates to Theresa that she now can speak and sends back a SIP 200 (OK) response with an SDP Answer to Tobias's UE, repeating that the audio media stream has been set to active:

```
m=audio 4011 RTP/AVP 0
```

As Theresa has already accepted the call, the UE does not need to send a SIP 180 (Ring-ing) response anymore. After this second SIP INVITE transaction, the audio connection between Tobias and Theresa is active.

#### 12.9.10 Session From Non-IMS SIP Terminal

In the example above it was assumed that an IMS Terminal sets up a call to a nonIMS SIP terminal, which does only support the basic SIP and SDP functionalities. In this example (see Figure 12.26) it is assumed that a basic SIP terminal calls towards an IMS client. Again this scenario is handled locally at the involved IMS UE, i.e. no specific interworking units or procedures within the IMS network are required to make call establishment work.



**Figure 12.26** Session establishment from a non-IMS terminal

This scenario is rather simple, as in this case Tobias's UE will only include in the first SDP Offer the audio media characteristics it wants to support, i.e. the SIP INVITE request will include the following SDP media line:

```
m=audio 3458 RTP/AVP 0 3
```

When Theresa's phone receives this SDP Offer, it will start immediately to reserve the required resources. After successful resource reservation it will start to ring and at the same time send back a SIP 180 (Ringing) response, which is sent unreliable (as the remote side does not support the '100rel' mechanism – see Section 11.5.2) and includes the SDP Answer, i.e.:

```
m=audio 4011 RTP/AVP 0
```

This SDP Answer is repeated in the SIP 200 (OK) response, which is sent out once Theresa accepts the call.

### 12.9.11 Related Standards

3GPP TS 24.930	Signalling flows for the session setup in the IP Multimedia core network Subsystem (IMS) based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3
RFC 3312	Integration of Resource Management and Session Initiation Protocol (SIP)
RFC 4032	Update to the Session Initiation Protocol (SIP) Preconditions Framework
RFC 5009	Private Header (P-Header) Extension to the Session Initiation Protocol (SIP) for Authorization of Early Media

## 12.10 Routing of GRUUs

During the SIP dialog establishment the phones of Tobias and Theresa have exchanged their GRUUs, which were assigned to them during registration (Section 11.13.5). In this section we now look at a possible usage of the GRUU.

### 12.10.1 Theresa Registers her Laptop

Let us assume that during the ongoing call Theresa registers from her laptop with the following information in a new REGISTER request from the laptop towards her S-CSCF:

```
REGISTER home2.hu SIP/2.0
From: <sip:theresa@home2.hu>;tag=6bnz4yio4
To: <sip:theresa@home2.hu>
Contact: "Laptop - Theresa" <sip:[5555:166:77:88:12]:4444>;+sip-
instance="sntr4hb7nsbn"
```

The laptop gets successfully registered at the S-CSCF that was assigned to Theresa during the registration of her mobile phone. Remember that all registrations for a user must end

up at the same S-CSCF. After successful authentication, the S-CSCF returns a 200 (OK) to Theresa's laptop, indicating the temporary and public GRUUs assigned to her in the Contact header (Section 11.13.5). For this example only the public GRUU is of relevancy, which is: "sip:theresa@home2.hu;gr=sntr4hb7nsbn".

After the laptop's registration, Theresa's S-CSCF sends out registration-state information to all of Theresa's terminals that are currently registered, indicating the newly registered contact address of the laptop, which includes (besides other) the following information, of which also Theresa's mobile phone is made aware:

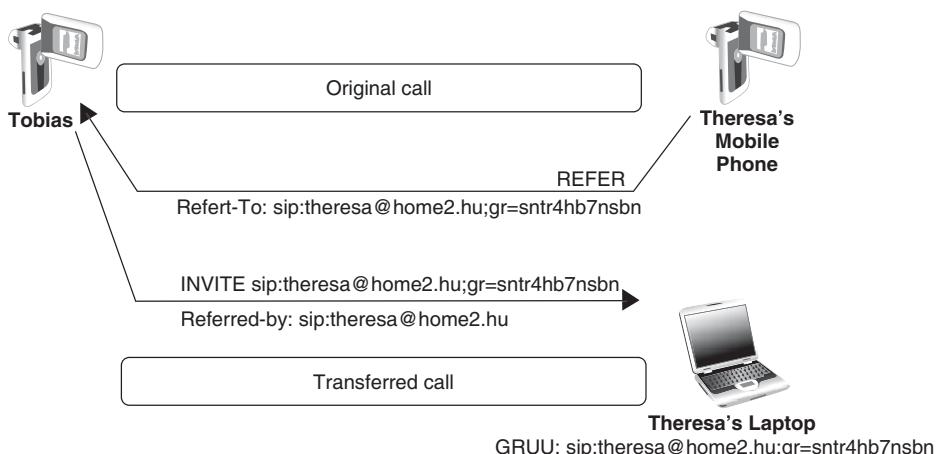
```
<registration aor="sip:theresa@home1.fr" id="x1" state="active">
    <contact id="78" state="active" event="registered">
        <uri>sip:[5555:166:77:88:12]:4444</uri>
        <display-name>Laptop - Theresa</display-name>
        <gr:pub-gruu uri="sip:theresa@home2.hu
        ;gr=sntr4hb7nsbn" />
    </contact>
</registration>
```

Please note that in this example we do not look at other parameters provided by Theresa's laptop, such as feature tags, ICSI and IARI values etc. Also, the already registered contacts as well as the implicitly registered public user identities of Theresa are not shown here.

From this registration state information Theresa's mobile phone learns that the public user identity 'sip:theresa@home2.hu' (registration aor) has been registered newly (contact event='registered') from a device which is known as 'Laptop – Theresa' (display-name) and has a public GRUU with the value 'sip:theresa@home2.hu;gr=sntr4hb7nsbn' (gr:pub-gruu) to it.

#### 12.10.2 REFER Request in Order to Transfer the Ongoing Call to Theresa's Laptop

Theresa now selects from her mobile phone that she wants to transfer the ongoing call to the laptop (see Figure 12.27).



**Figure 12.27** Routing of GRUU

In order to transfer the ongoing call from Theresa's mobile phone to Theresa's laptop, Theresa's mobile phone sends out a REFER request on the existing SIP dialog with Tobias:

```
REFER sip:[5555:1:2:3:4]:1357 SIP/2.0
To: "Your Brother" <sip:tobi@brother.com>;tag=veli
From: "My beloved Sister" <sip:Theresa@sister.com>;tag=schwester
Refer-To: sip:theresa@home2.hu;gr=sntr4hb7nsbn
Call-ID: apb03a0s09dkjdfglkj49555
```

The Refer-To header includes the address to which the ongoing call should be transferred to, which is the GRUU assigned to Theresa's laptop, which the mobile phone obtained from the registration state information.

In order to address the laptop as the specific destination device, Theresa's phone needs to indicate the public GRUU (that it obtained from the registration state information) into the Refer-To header, as only the GRUU will deliver the call directly and only to the laptop.

The REFER request is sent on the existing dialog between Theresa and Tobias and therefore has the same Call-ID, To and From tag values. As the request is sent from Theresa's phone to Tobias's phone, the values of the To and From headers are swapped compared to the initial INVITE request, which was sent from Tobias's to Theresa's phone.

Tobias's phone receives the REFER request and will:

- accept the call transfer request by returning a 202 (Accepted) response for the REFER request;
- send out a new INVITE request to the referred-to terminal, i.e. the laptop, see below; and
- keep Theresa's mobile phone updated about the ongoing call establishment with Theresa's laptop by sending NOTIFY messages on the dialog between Tobias's and Theresa's mobile phones;
- once the new session with Theresa's laptop has successfully been established, send a SIP BYE request on the original dialog between the two mobile phones, switch the media over to the newly established media connection (towards the laptop) and go on with the call.

All this can be performed completely seamlessly for Tobias, i.e. his mobile phone might perform all these procedures in the background, without giving indications about the ongoing transfer to Tobias. Alternatively, the mobile phone can of course request a confirmation from Tobias before transferring the call.

#### *12.10.3 Setting up the New Call to Theresa's Laptop*

In order to transfer the call to Theresa's laptop, Tobias's phone sends out a new INVITE request:

```
INVITE sip:theresa@home2.hu;gr=sntr4hb7nsbn SIP/2.0
From: "Your Brother" <sip:tobi@brother.com>;tag=xxggf
```

---

```
To: "My beloved Sister" <sip:Theresa@sister.com>
CSeq: 89 INVITE
Call-ID: apb03a0s09dkjdfglkj49555
```

This INVITE request creates a completely new SIP dialog, which is independent of the original dialog between the two mobile phones and has assigned to it a new Call-ID, From tag and not yet assigned To tag. The request is routed in the same way as shown in Section 12.3, with the exception that the S-CSCF of Theresa's network gets based on the request URI, that this request is routed based on a GRUU that was assigned to Theresa's laptop. It therefore does not fork the incoming INVITE request to towards any other registered terminal of Theresa (e.g. her mobile phone), but only sends it to the contact address related to this GRUU, which is the registered IP address of Theresa's laptop (sip:[5555:166:77:88:12]:4444).

After forwarding the INVITE request to Theresa's laptop, the session gets established in the same way as described in this chapter. After the resources are available on both sides, the laptop will ring, Theresa will pick up the call and Tobias's phone will release the ongoing call between the two mobile phones and switch over the media to the media connection towards the laptop.

### 12.10.3.1 Related Standards

Specifications relevant to Section 12.10 are:

RFC 3515	The Session Initiation Protocol (SIP) Refer Method
draft-ietf-sip-gruu-15	Obtaining and Using Globally Routable User Agent (UA) URIs (GRUU) in the Session Initiation Protocol (SIP)
draft-ietf-sipping-gruu-reg-event	Registration Event Package Extension for Session Initiation Protocol (SIP) Globally Routable User Agent URIs (GRUUs)

## 12.11 Routing of PSIs

The concept of a Public Service Identity (PSI; i.e., a URI that is not related to a user but to a service) is explained in Section 3.5.5.

This section is a basic introduction to the routing principles of PSIs, as they are quite different from those that are applied between two IMS users.

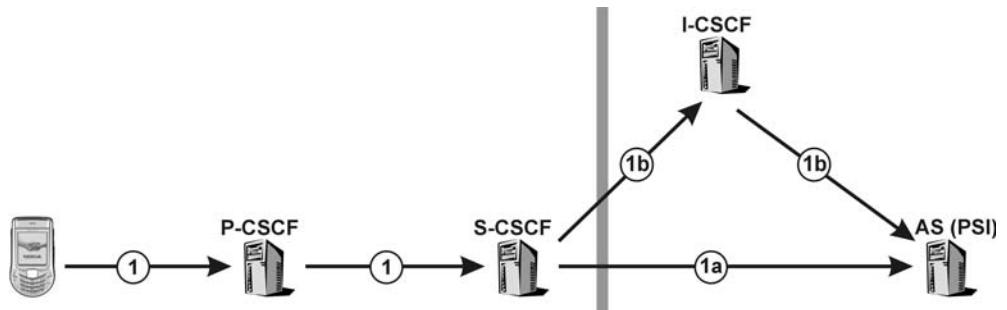
As PSIs are not registered, requests to and from them usually do not need to traverse any S-CSCF.

There are three scenarios for PSI routing.

### 12.11.1 Scenario 1: Routing From a User to a PSI

This occurs, for example, when a user calls into a conference (Figure 12.28). In this case the request needs first to traverse the user's S-CSCF, which can then:

1. Either, resolve the PSI immediately and route it directly to the AS.



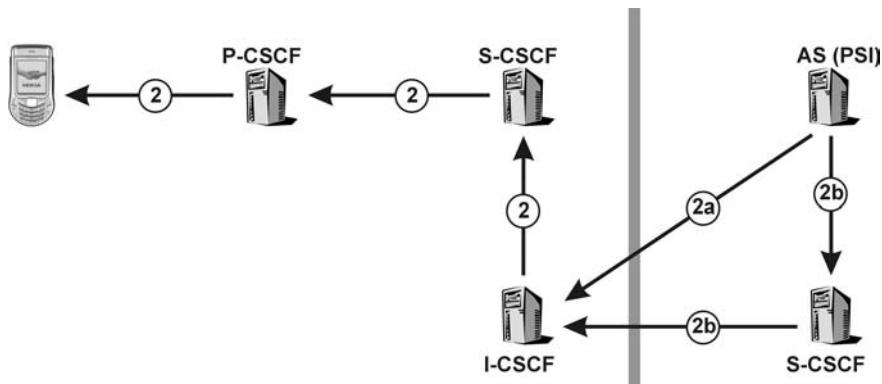
**Figure 12.28** Routing from a user to a PSI

2. Or, is able to resolve the address of an I-CSCF in the home network of the PSI. The I-CSCF will then query the HSS – where routing information about the PSIs will be stored – and route the request directly to the hosting AS. Note that an S-CSCF can also be assigned for the PSI, in which case this would be contacted first. This scenario is not shown here.

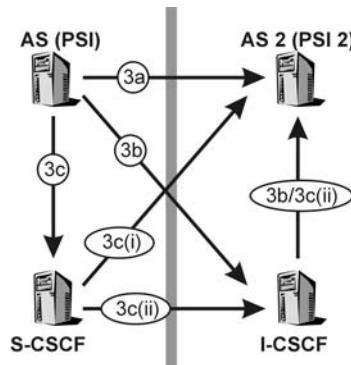
#### 12.11.2 Scenario 2: Routing From a PSI to a User

This occurs, for example, when a conference server (the focus) invites a user to a conference (Figure 12.29). In this case the AS sends the request:

1. Either, directly to the I-CSCF of the terminating user's home network, if the AS can resolve this address on its own.
2. Or, to an S-CSCF in the AS's home network, which then resolves the address of the I-CSCF of the terminating user's home network.



**Figure 12.29** Routing from a PSI to a user



**Figure 12.30** Routing from an AS to a PSI

### 12.11.3 Scenario 3: Routing From a PSI to Another PSI

This occurs, for example, when two conferences are interconnected such that one conference server (focus) sends an INVITE request to another focus (Figure 12.30). For this case, several routing possibilities exist. The originating AS can route the request:

1. Either, directly to the terminating AS that is hosting the second PSI, if the originating AS is able to resolve the PSI directly.
2. Or, if it cannot resolve the address of the second PSI directly, to an I-CSCF of the terminating AS, which then queries the HSS with the second PSI and sends the request directly to the terminating AS. Note that an S-CSCF can also be assigned for the PSI, in which case this would be contacted first. This scenario is not shown here.
3. Or, if it cannot resolve any part of the terminating address, to an S-CSCF in its own home network, which will then:
  - Either, resolve the address of the second PSI and send the request to the terminating AS directly.
  - Or, cannot resolve the address of the second PSI directly and, therefore, sends the request to an I-CSCF in the home network of the second PSI. This I-CSCF will then act in the same way as item (2) in Scenario 3.

## 12.12 A Short Introduction to GPRS

### 12.12.1 Overview

As in this example GPRS is used as the underlying access technology to IMS for both users, this section describes the basic principles used by GPRS, especially those relevant for IMS.

The General Packet Radio Service (GPRS) is the Packet-Switched (PS) domain of the Global System for Mobile communications (GSM) and the Universal Mobile Telecommunications System (UMTS) network. It provides Internet Protocol (IP) connectivity to attached User Equipment (UE) via so-called Packet Data Protocol (PDP) contexts. As

expressed in the name, it is a logical connection (context) that is related to a specific packet-based protocol.

The UE will be able to send IP packets over the air interface after it has established a PDP context.

This chapter concentrates only on those parts of GPRS that are necessary for a UE to access an IP Multimedia Subsystem (IMS) network. The detailed procedures within the Gateway GPRS Support Node (GGSN) or the Serving GPRS Support Node (SGSN), as well as detailed message coding and flows are not discussed here. The aim of this chapter is to provide the reader with a short overview of the basic principles that lie behind GPRS and its PDP contexts.

### *12.12.2 Packet Data Protocol (PDP)*

A Packet Data Protocol (PDP) context offers a packet data connection over which the UE and the network can exchange IP packets. Usage of these packet data connections is restricted to specific services. These services can be accessed via so-called access points.

#### **12.12.2.1 Primary PDP Context Activation**

This procedure is used to establish a logical connection with the Quality of Service (QoS) functionality through the network from the UE to the GGSN. PDP context activation is initiated by the UE and changes the session management state to active, creates the PDP context, receives the IP address and reserves radio resources. After a PDP context activation the UE is able to send IP packets over the air interface. The UE can have up to 11 PDP contexts active concurrently.

#### **12.12.2.2 Secondary PDP Context Activation**

A secondary PDP context activation allows the subscriber to establish a second PDP context with the same IP address as the primary PDP context. The two contexts may have different QoS profiles, which makes the feature useful for applications that have different QoS requirements (e.g., IP multimedia). The access point name, though, will be the same for the primary and secondary PDP contexts.

#### **12.12.2.3 PDP Context Modification**

The UE, the SGSN or the GGSN initiate this procedure for updating the corresponding PDP context. Additionally, the radio access network is able to request a PDP context modification from the SGSN (e.g., when coverage to the UE has been lost). The procedures modify parameters that were negotiated during an activation procedure for one or several PDP contexts.

#### **12.12.2.4 PDP Context Deactivation**

This procedure is used to delete a particular logical connection between the UE and the GGSN. The initiative to deactivate a PDP context can come from the UE, the SGSN, the Home Location Register (HLR) or the GGSN.

### 12.12.2.5 Access Points

Access points can be understood as IP routers that provide the connection between the UE and the selected service. Examples of such services are:

- Multimedia Messaging Service (MMS);
- Wireless Application Protocol (WAP);
- direct Internet access;
- IP Multimedia Subsystem (IMS).

Depending on the operator of the network, more than one of these services might be provided by the same access point. The UE needs to be aware of an Access Point Name (APN) – the address of a GGSN – which gives access to the service-providing entity (e.g., an MMSC, the Internet or the P-CSCF). One GGSN may provide different services that can be accessed by different APNs.

When establishing a primary PDP context with an APN the UE receives an IP address or – in the case of IPv6 – an IPv6 prefix that it has to use when communicating over that PDP context. This means that when a UE has established several connections to different APNs the UE will have different IP addresses for each of the provided services.

### 12.12.3 PDP Context Types

From the viewpoint of the IMS, it is important to distinguish between the PDP context types shown in Figure 12.31. A primary PDP context is used whenever the first connection with a specific APN is established. If further connections with the same APN are needed, secondary PDP contexts are established.

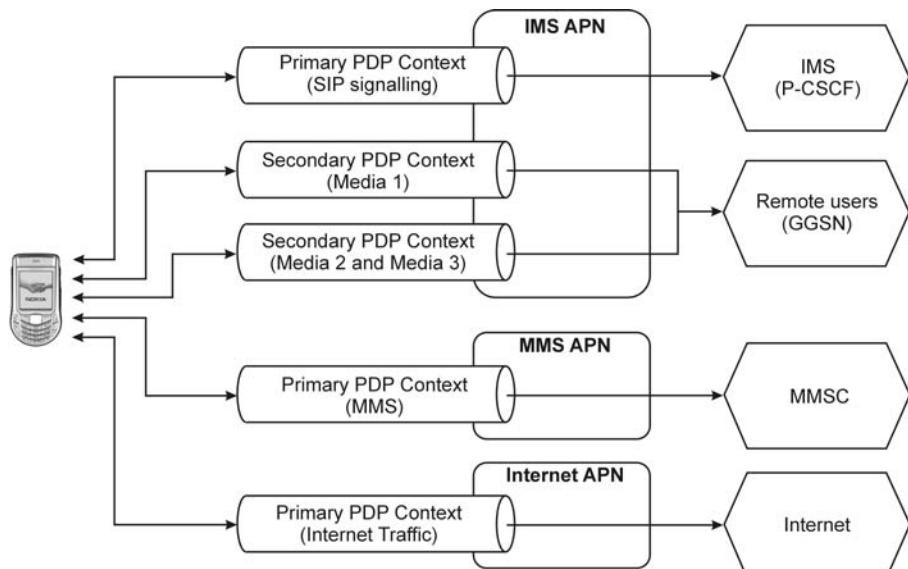


Figure 12.31 PDP context types

IMS signalling can be transported either in a dedicated signalling PDP context or a general purpose PDP context. If a dedicated signalling PDP context is established, all IMS-related media streams must be put into one or more separate secondary PDP contexts.

Over a general IMS PDP context the UE may send signalling and media streams together. Nevertheless, it still has the option to create separate PDP contexts for media in this case as well.



# 13

## An example IMS Voice Call Continuity Procedures

### 13.1 Overview

This chapter provides a detailed description of how a specific voice call continuity (VCC) example works. It handles anchoring of CS and IMS calls as well as domain transfers from CS to IMS domains and vice versa. VCC is generally described in Section 3.20.

VCC allows circuit switched voice calls to be transferred to IMS multimedia telephony sessions and vice versa. CS and IMS are seen as two different communication domains, therefore the procedure of handing over a call from one domain to another is called domain transfer.

In order to allow domain transfer, every call that is set up by the user in either CS or IMS domain, will be checked whether it could be subject to such a domain transfer. If the call could be subject to domain transfer, it will be routed to the VCC Application Server (VCC-AS), which then anchors this call. As the VCC-AS is located in the IMS domain this means, that all CS calls that can be subject to domain transfer, need to be interworked into IMS.

In this example we will look at the following scenario:

- Tobias dials the telephone number of his sister Theresa in order to set up a CS voice call (Section 13.3.1);
- during the routing of the call, the CS domain of Tobias's network is made aware that the call can be subject to VCC domain transfer, therefore the call gets interworked at the MGCF into Tobiases IMS domain (Section 13.3.3);
- the call gets anchored at the VCC-AS within Tobiases IMS domain and then gets forwarded to Theresa's IMS domain (Section 13.3.5);
- Theresa's network also sees the call as a possible subject to VCC domain transfer and therefore forwards the call to the local VCC-AS, where the call is anchored again (Section 13.3.7);

- Theresa's VCC-AS decides to deliver the call to Theresa via the IMS domain, the audio connection between the two users is established (Section 13.3.8);
- whilst the call is ongoing, Tobias's walks into I-WLAN coverage, therefore his UE triggers the procedures for a VCC domain transfer from CS to IMS, i.e. it sets up a parallel call via Tobias's IMS domain (with I-WLAN as an access network) (Section 13.4.1);
- the parallel call is delivered to Tobias's VCC-AS, which then performs the domain transfer procedures – after this, Tobias is no longer using his CS domain but his IMS domain for the call towards Theresa. The domain transfer happens automatically and is apparently seamless to the users, i.e. without any of the two users having to invoke manually any specific procedures and without the users getting even aware, that the domains have changed (Section 13.4.3);
- Theresa adds a video media stream to the call (Section 13.5);
- whilst the call is ongoing, Theresa's phone is made aware that it is about to lose GPRS coverage, but CS coverage is still available, therefore it triggers the procedures for a VCC domain transfer from IMS to CS, i.e. it sets up a parallel call via Theresa's CS domain (Section 13.6.1);
- whilst routing this parallel call, Theresa's CS domain is made aware that the call is related to a VCC domain transfer and therefore interworks it to Theresa's IMS domain at the MGCF (Section 13.6.2);
- the parallel call is routed through Theresa's IMS domain to her VCC-AS, which then performs the domain transfer procedures from IMS to CS. After this, Theresa is no longer using her IMS domain but her CS domain for the call towards Tobias. The domain transfer happens automatically but in this case is not seamless to the users, i.e. the users do not have to invoke anything manually at their phones, but the recently added video component is lost, as the CS domain does not support video calls (Section 13.6.3).

For this example, several assumptions have been taken (see also Table 13.1 and Table 13.2)

- Tobias is making use of a multi-radio mobile phone, which supports connections to 3GPP CS domain as well as to 3GPP I-WLAN.
- Theresa is making use of mobile phone which supports connections to 3GPP CS and 3GPP GPRS domains;
- Both UEs have been configured with the VCC specific settings of their network operators (see Section 13.2).

This example concentrates mostly on SIP and SDP related procedures within the given scenario, as well as interworking issues. It will be shown how an IMS AS works as an SIP Back-to-Back User Agent (B2BUA) and also how IMS routing in the case of CS to IMS interworking is performed. This example nevertheless does not provide any details on CS call establishment and media routing procedures, VCC error handling procedures or any specific non-SIP related communication taking place (e.g. between the MGCF and the MGW).

VCC covers a larger set of scenarios than shown here. This example does not look into cases where e.g. both users are making use of CS domain of the same operator or where the Domain Selection Function (DSF) decides to deliver the incoming terminating call via the CS domain to Theresa.

**Table 13.1** VCC Related Telephone Numbers and Addresses

	Addresses in the home network of Tobias	Addresses in the home network of Theresa
<b>Phone</b>	+33123456789 [5555::1:2:3:4]:1357	+36987654321 [5555::5:6:7:8]:1006
<b>MGCF</b>	mgcf1.home1.fr	mgcf2.home2.hu
<b>MGW</b>	[5555::36:74:58:96]	[5555::71:82:93:64]
<b>VCC-AS</b>	vccas1.home1.fr	vccas2.home2.hu
<b>P-CSCF</b>	pcscf1.home1.fr	pcscf2.home2.hu
<b>I-CSCF</b>	icscf1.home1.fr	icscf2.home2.hu
<b>S-CSCF</b>	scscf1.home1.fr	scscf2.home2.hu

**Table 13.2** VCC Related Routing Numbers and SIP Addresses

	Home network of Tobias	Home network of Theresa
<b>VDN</b>	+336574839311	+361139384756
<b>VDI</b>	vdi-vccas.home1.fr	vdi-vccas.home2.hu
<b>IMRN</b>	+3396587436	+3663478569

In addition to the basic VCC procedures, this chapter also treats a variety of additional IMS procedures:

- interworking of a CS call to IMS at the MGCF and MGW (Section 13.3.3.1);
- resolving of a PSI at the I-CSCF (Section 13.3.4);
- resolving of the S-CSCF for a unregistered user at a AS (Section 13.3.5.2);
- resolving of a tel-URL at the S-CSCF (Section 13.3.6.2);
- general behaviour of a SIP Back-to-Back User Agent (B2BUA) at the B2BUA (e.g. Section 13.3.5).

## 13.2 Configuring the Clients with Communication Continuity Configuration Parameters

In order to use the VCC functionality, the phones of Tobias and Theresa need to be configured with a set of parameters. As we have seen in Section 11.2, generic parameters can be configured to a client over the OMA Device Management (DM) mechanism. In order to provide VCC related configuration to the clients, a new OMA DM Management Object (MO) for Communication Continuity has been defined by 3GPP. Note again, that the MO is not transported via SIP, but that OMA DM is a protocol on its own, that delivers the MO to the phone.

Usually the OMA DM MO for Communication Continuity is only provided once (e.g. when the phone connects for the first time to the operator's network) to a client, as the configuration parameters for VCC are the same for every call.

We assume in this example, that both Theresa's and Tobias's phone received a Communication Continuity MO some time in the past and therefore both devices are configured with the relevant parameters.

The Communication Continuity MO consists of the following configuration parameters:

- VDI, the VCC Domain Transfer URI, i.e. the URI that is used as the destination address when the UE wants to transfer a call from CS to IMS (see Section 13.4);
- VDN, the VCC Domain Transfer Number, i.e. the telephone number (MSISDN) that is used as the called party number when the UE wants to transfer a call from IMS to CS (see Section 13.6);
- Preferred Domain, indicating the network operator's preference for which domain should be used to originate calls from;
- immediate domain transfer, indicating the network operators whether to immediately transfer any existing calls to the preferred domain, once the domain becomes available;
- DT CS-to-IM CN direction, indicating whether it is allowed for the UE to initiate a domain transfer from CS to IMS domain;
- DT IM CN-to-CS direction, indicating whether it is allowed for the UE to initiate a domain transfer from IMS domain to CS;
- DT in held\_waiting calls, indicating whether the UE is allowed to transfer an active call if it has other calls on hold or other calls are waiting. If this is allowed, only the active call will be transferred into another domain – all other calls (either held or waiting) will be released.

Tobias's phone is configured with the following parameters:

- VDI: sip:vdi-vccas.home1.fr;
- VDN: +336574839311;
- Preferred Domain: '1' – indicating that Tobias should, if both IMS and CS domains are available, set up his calls from the IMS domain;
- immediate domain transfer: '1' – indicating that domain transfer should be done immediately once the preferred domain (IMS) becomes available;
- DT CS-to-IM CN direction: '0' – indicating that the CS to IMS domain transfer is allowed;
- DT IM CN-to-CS direction: '0' – indicating that the IMS to CS domain transfer is allowed;
- DT in held\_waiting calls: '0' – indicating that Tobias can perform domain transfer, even if there are calls on hold or calls are waiting.

Theresa's phone is configured with the following parameters:

- VDI: sip:vdi-vccas.home2.hu;
- VDN: +361139384756;

- Preferred Domain: ‘1’ – indicating that Theresa should, if both IMS and CS domains are available, set up her calls from the IMS domain;
- immediate domain transfer: ‘0’ – indicating that it is not necessary to transfer all ongoing CS calls to IMS, once IMS becomes available;
- DT CS-to-IM CN direction: ‘0’ – indicating that the CS to IMS domain transfer is allowed;
- DT IM CN-to-CS direction: ‘0’ – indicating that the IMS to CS domain transfer is allowed;
- DT in held\_waiting calls: ‘1’ – indicating that Theresa’s phone should not perform domain transfer if there are any calls on hold or calls are waiting.

### 13.3 Setting up the Initial Call and Call Anchoring

#### 13.3.1 Tobias Sets up a CS Call Towards Theresa

Tobias is walking from his office in France to a nearby café and decides to call his sister Theresa. He selects her name from the contact manager in his phone, where her phone number is stored as +36987654321. The phone currently is connected to the 3GPP CS domain of Tobias’s network operator, but it is not currently connected to the IMS communication domain. Therefore the phone selects the CS domain to set up the audio call to Theresa.

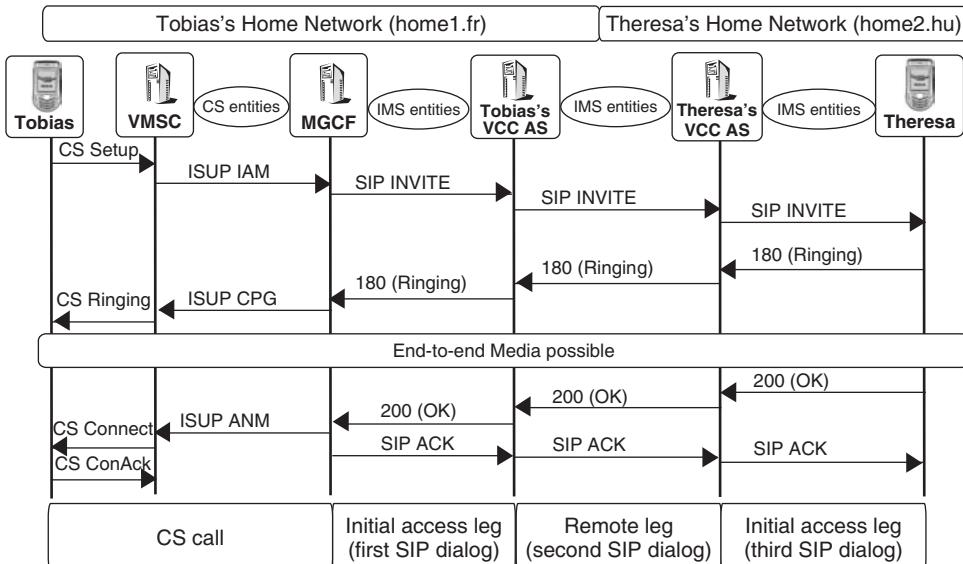
The phone sends out a CS setup message, which includes several protocol elements, out of which the following three are important for this example (see Figure 13.1):

- the Calling Party Number Information Element (IE) is set to Tobias’s phone number +33123456789;
- the Called Party Number IE is set to Theresa’s phone number, as it was dialled by Tobias, +36987654321;
- the Bearer Capability IE indicating that AMR-WB is supported as an audio codec for this call.

As said above, the detailed procedures and protocol element encoding for CS call setup are not shown here, we will only look at those parts of the CS domain that are essential to understand VCC behaviour and the related IMS procedures. For simplicity it is assumed that for this first call setup, only one codec is available to be used, i.e. AMR-WB. More complex codec and media handling will be shown during the later part of the example.

#### 13.3.2 Anchoring Decision and Routing the CS Call to the MGCF

The CS setup message ends up in the Visited Mobile Switching Centre (VMSC) to which Tobias’s phone is attached. Within the VMSC a trigger is set, that indicates that specific services have to be executed for all calls from Tobias’s phone. These CS specific triggers are set by procedures in the CS domain that are not further shown here. Due to this trigger event, the VMSC queries the CAMEL subsystem by sending a CAMEL IDP message to the gsmSCF, including the calling and called party numbers, as well as an indication that the call is an audio call. The address of the gsmSCF is part of the trigger, similar as the address of the AS is part of IMS Filter Criteria (see Section 3.13).



**Figure 13.1** Basic interworking of CS and IMS calls at MGCF

The gsmSCF is part of the VCC Application Server (see Section 3.20). How the logical functions that comprise the VCC AS are communicating amongst each other is left to the specific implementations and is not defined within 3GPP standards.

The gsmSCF communicates with the CSAF and the CAMEL service function and detects that the call from Tobias to Theresa could be subject to a VCC domain transfer on Tobias's side. Due to that, the VCC AS (gsmSCF) decides that the call needs to be anchored at the VCC AS. As anchoring can only be performed in the IMS domain, the call needs to be interworked first from CS to IMS and therefore needs to be rerouted to the MGCF. The VCC AS also must make sure that the IMS call afterwards is not directly forwarded to Theresa's network, but first is routed to Tobias's VCC AS, in order to anchor the call there. Therefore the gsmSCF and the CSAF assign a special CS phone number, the IP Multimedia Routing Number (IMRN) which will be used to deliver the call to the IMS VCC AS. How the IMRN is assigned is a network operator option and is not defined within 3GPP standards. When assigning the IMRN, the VCC AS also stores the called party number (Theresa's phone number) that was present in the CAMEL IDP message, as the IMRN now replaces Theresa's phone number until the call gets delivered to the VCC AS in the IMS domain.

After the IMRN has been assigned, the gsmSCF responds to the VMSCs CAMEL IDP request. Within the request it indicates the IMRN within the Destination Routing Address field. The IMRN then is used by the VMSC for further routing of the call. As the IMRN points to the VCC AS in the IMS domain, the VMSC now needs to deliver the call to the MGCF, in order to interwork the CS call to SIP and SDP signalling.

The VMSC now forwards the call by using a different protocol, namely SS7 ISUP (Signalling System Number 7 ISDN User Part) signaling. This is due to the fact that in the CS domain different protocols are used on the interfaces between the user to network

(UNI) interface and the so-called network-to-network interface (NNI). In IMS for both UNI and NNI SIP is used. The VMSC sends out an ISUP IAM message to the MGCF. The following fields in this IAM are of importance for the example:

- Calling Party Number IE set to Tobias's phone number: +33123456789;
- Called Party Number IE set to the IMRN (not anymore to the phone number of Theresa): +3396587436;
- the Bearer Capability IE indicating that AMR-WB is supported as a audio codec for this call.

Note that Theresa's phone number is not indicated in the protocol messages anymore and will not be delivered to the MGCF. It has been stored in the VCC AS when the anchoring decision was made and the IMRN was assigned.

### 13.3.3 Interworking the CS Call to IMS at the MGCF

#### 13.3.3.1 Interactions Between the MGCF and the MGW

After receiving the ISUP IAM, the MGCF selects the MGW which will handle the bearer connection interworking for the call from Tobias to his sister (see Figure 13.2). The MGCF performs the Reserve IMS Connection Point procedure sending a H.248/MEGACO Add Request to the MGW, indicating the following parameters:

- the 'Context Request' field to request a new context identifier;

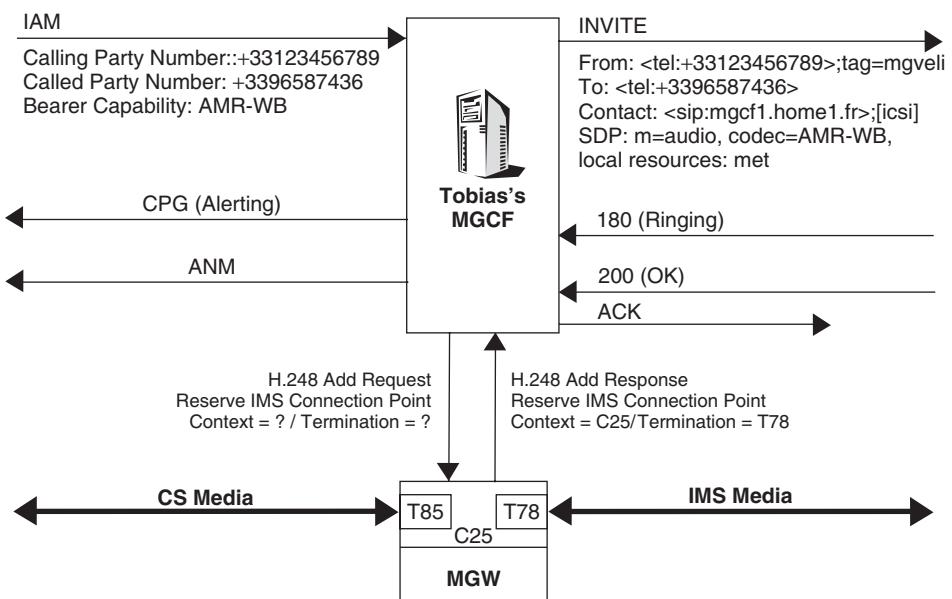


Figure 13.2 Basic dialog mapping at Tobias's VCC AS (DTF), acting as a SIP B2BUA

- the ‘IMS Terminating Request’ field to request a new IMS bearer termination;
- the ‘Local IMS Resources’ field set to the codecs that should be used on the bearer connection, which in this case is AMR-WB, as indicated in the Bearer Capability IE in the received ISUP IAM;
- the ‘Reserve Value’ field set, as there are multiple resources needed (one for the audio call and one for DTMF);
- optionally the ‘Notify Released Bearer’ field set, indicating that the MGCF wants to get informed by the MGW when the bearer gets released.

The MGW creates the related context and reserves locally the required resources. It acknowledges the Reserve IMS Connection Point procedure by sending a H.248 Add response to the MGCF, indicating:

- the ‘Context’ field set to the context identifier, in this example this is set to ‘C25’ – the context is newly created;
- the ‘IMS Termination’ field set to the reference for the IMS bearer termination that was created, in this example this is set to ‘T78’;
- the ‘Local IMS Resources’ field set to the AMR-WB codec, as this is the codec that will be used on the connection;
- the ‘Local Connection Addresses’ field set to the IP address and port number for the bearer connection towards the IMS side, in this example this is set to the IP address ‘5555::36:74:58:96’ and the port number ‘3458’.

Now the MGCF needs to make sure that the CS-side bearer also is terminated at the MGW. It therefore requests to add an additional termination to the existing context C25. It does so by performing the Prepare Bearer procedure by sending a H.248 Add request to the MGCF, including:

- the ‘Context Request’ field set to ‘C25’, the context that got created with during the Reserve IMS Connection procedure;
- the ‘Bearer Termination’ field in order to request a termination for a CS bearer;
- further fields, indicating the characteristics of the CS-side bearer as indicated in the Bearer Capability IE within the ISUP IAM

The MGW reserves the resources and creates the termination. It acknowledges the Prepare Bearer procedure by sending a H.248 Add response to the MGCF, indicating:

- the ‘Context’ field set to ‘C25’
- the ‘Bearer Termination’ field set ‘T85’, the CS-bearer termination that got now reserved and added to the context.
- the address of the reserved bearer and further characteristics of the CS-bearer, which will be sent back to the CS-side (VMS) within a Bearer Capability IE in an ISUP ACM message

Now both terminations have been created at the MGW, which means that the MGW is the endpoint for both media streams – the one towards the CS side and the other towards the IMS side.

Later on, when a 180 (Ringing) indication from Theresa's side is received, the MGCF will instruct the MGW to connect the two terminations, which means that it will let the AMR-WB media go end-to-end between the two phones.

The interactions between the MGCF and the MGW are shown on a very high level, the details of the H.248 protocol are not described any further.

### 13.3.3.2 Constructing the INVITE Request

After receiving the ISUP IAM message, the MGCF starts interworking the call from the CS side to the IMS domain and constructs a SIP INVITE request.

```
INVITE tel:+3396587436 SIP/2.0
Via: SIP/2.0/UDP mgcf1.home1.fr
Route: <sip:icscf1.home1.fr;lr>
From: <tel:+33123456789>;tag=mgveli
To: <tel:+3396587436>
P-Charging-Vector: icid-value="AyretyU0dm+6";orig-ioi=home1.fr
P-Asserted-Identity: <tel:+33123456789>
P-Asserted-Service: urn:urn-xxx:3gpp-service-
ims.icsi.mmTEL
Accept-Contact: *;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-
ims.icsi.mmTEL"
Contact: <sip:mgcf1.home1.fr>
;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-ims.icsi.mmTEL"
Call-ID: wddds2nhdg
CSeq: 2810 INVITE
```

Sections 12.2 and 12.3 give more details on how the headers in a SIP INVITE request are set. In this section the setting of these headers is only explained when looking at them from the interworking perspective at the MRFC due to the ongoing VCC scenario.

The Request URI and the To header are set for the phone number indicated in the Called Party Number IE of the ISUP IAM message, which is the IMRN as it was assigned by the VCC AS. The phone number is indicated as a telephone URI (tel-URI) and not a SIP URI.

The To header and the P-Asserted-Identity header are set to the phone number indicated in the Calling Party Number IE of the ISUP IAM message, which is Tobias's phone number. The MGCF trusts the information received from its CS domain and therefore is allowed to assert the tel-URI, i.e. it can directly put a P-Asserted-Identity header into the SIP request.

As the MGCF is the SIP endpoint (i.e. it is acting as a SIP UA) of the dialog, it puts its own address into the Contact header. By doing so it makes sure to receive all requests that are sent from the other side of the SIP dialog towards Tobias. The MGCF must receive all requests related to this SIP dialog, as it needs to interwork them towards the CS domain.

As this is the initial INVITE request, the MGCF sets the From-tag. It has to wait for the first response from the remote side, in order to be made aware of the To-tag.

Also the address in the Via header is set to the address of the MGCF, which guarantees that it will receive all responses sent for this SIP INVITE request, which the MGCF also needs to interwork towards the CS domain.

The Route header points to the next hop, to which the MGCF will send the request. As the MGCF is not capable of querying the HSS and the SLF on its own, it cannot find out to which entity the request needs to be routed based on the destination address indicated in the request URI (which is set to the IMRN tel-URI). It therefore needs to send the request first to the local I-CSCF, which then will resolve the IP-address of the final destination or at least the next hop on the route towards the final destination.

The MGCF sets the P-Charging-Vector header based, as it is creating the media connections, by setting the ICID to a new value and the originating Type 2 IOI (home1.fr).

As the audio call gets interworked from the CS domain, the MGCF also puts the IMS Multimedia Telephony Communication Service Identification (MMtel ICSI) into the following headers:

- P-Asserted-Service header, in order to invoke related services in the originating and terminating network, e.g. specific charging for multimedia telephony or routing the Telephony Application Servers (TAS) of the users;
- Accept-Contact header, to let Theresa's S-CSCF select those of Theresa's phones, which are registered with the MMtel ICSI (see Section 11.9);
- Contact header (as a parameter to the MGCF address), to indicate to Theresa's phone that the multimedia service is supported on the remote end of the call, i.e. Tobias's phone.

Again, the MGCF is allowed to put the P-Asserted-Service header as it is a trusted entity.

### 13.3.3.3 Constructing the SDP Offer

The SIP INVITE request also includes a body, containing an SDP Offer:

```
v=0
o=- 9997777888 47 IN IP6 5555::36:74:58:96
s=-
c=IN IP6 5555::36:74:58:96
t=907165566 0
m=audio 3458 RTP/AVP 97 98 a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
a=tcap:1 RTP/AVP RTP/AVPF
a=pcfg:1 t=1||2
```

The details of how to set the SDP parameters are explained in Sections 12.5 and 12.9. Here the setting of the parameters is only explained from the perspective of interworking at the MGCF and the related VCC procedures.

The connection line (c-line) indicates the IP address (**5555::36:74:58:96**) of the MGW that was returned from the MGW to the MGCF for the IMS termination ('T78') in the 'Local Connection Addresses' field that was received in the response to the IMS Connection Procedure (the first H.248 Add Request in Section 13.3.3.1). The port number (3458) in the m-line is set to the port value that was received in the same field. The indicated codec (AMR-WB) was received in the same response in the 'Local IMS Resources' field.

The media stream will be routed from Tobias's phone over the CS domain to the MGW and there it will be interworked to the IMS communication domain. We will see later on, that in this initial call, the media will be routed via Theresa's GGSN to her phone.

For simplification of this example we assume that the MGCF only proposes the AMR-WB codec for the audio session to be used. It indicates AVP Payload Type 97 in the m-line and maps the payload type in the first a-line to AMR-WB. In addition to that it indicates support for DTMF tone transport via the telephone-event mechanism (AVP payload type 98 in this example).

As the MGW has all the resources that are required for sending the AMR-WB audio codec locally available, it sets the related preconditions to met by indicating that current local resource status ('sendrecv') matches the related desired status:

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

Note that there are scenarios, where the CS network can indicate to the MGCF (by means of the ISUP COT parameter), that the required resources are not yet available in the CS network. In such cases, the MRFC would set the preconditions as not met. For this example it is assumed that the resources are available in the CS network to send the AMR-WB audio codec and therefore the preconditions are set to meet.

As the MGCF does not know whether the remote side needs to reserve resources in order to be able to send and receive the AMR-WB audio codec, it indicates that it does not have any mandatory requirement on the reserved resources at the remote side:

```
a=des:qos none remote sendrecv
```

and that it is not aware of the current status of resource reservation at the remote side:

```
a=curr:qos remote none
```

Furthermore the MGCF supports AVPF, but cannot be sure whether this is also supported by the remote side. Therefore the last two lines in the above SDP Offer example make use of the SDP Capability negotiation mechanism to allow the remote side to upgrade the call to AVPF, if supported.

The MGCF sends out this SIP INVITE request to the next hop, i.e. the I-CSCF of Tobias's home network, which is indicated in the Route header.

### 13.3.4 Forwarding the IMS call to the VCC Application Server (resolving and direct routing of PSI)

Upon receiving the SIP INVITE request from the MGCF, the I-CSCF removes its address from the Route header and tries to forward the request towards its final destination. As no further Route headers are available, the I-CSCF has to use the information provided in the Request URI to gain further routing information. The Request URI is set to a tel-URI, which can be resolved by an HSS in the local network. As we assume that there are several HSS provided in the network, the I-CSCF first has to query the SLF in order to find out, which HSS can provide routing specific information.

The I-CSCF performs a User Location Query by sending a Diameter Location Info Request (LIR) via the Dx interface to the local SLF, containing

- the ‘R’ command flag set to ‘1’, indicating that this is a Diameter request;
- the Command-Code set to ‘302’, indicating the Diameter ‘Location Info’ command
- the ‘Public-Identity’ AVP (601) set to the tel URL indicated in the SIP INVITE request URI, i.e. to tel:+3396587436, which is the IMRN that was assigned by the VCC AS;
- the ‘Origin-Host’ AVP (264) set the address of the querying I-CSCF, i.e. ‘icscf1.home1.fr’;
- the ‘Origin-Realm’ AVP (296) set to the domain name of the operator network in which the I-CSCF is located, i.e. ‘home1.fr’;
- the ‘Destination-Realm’ AVP (283) set to the home domain of the SLF/HSS, i.e. ‘home1.fr’, as this is the domain within which the user location information is queried;
- the ‘Originating-Request’ AVP (633) in order to indicate, that the request is an originating SIP request.

As the Destination-Host AVP (293) is not present, the request is routed based on the internal routing configuration of the I-CSCF to the SLF which extracts the IMRN from the ‘Public-Identity’ AVP and checks which HSS holds the related subscription data. The SLF then acts as a Diameter Redirect Agent and returns to the I-CSCF the Location Info Answer (LIA), containing:

- the ‘R’ command flag set to ‘0’, indicating that this is a Diameter answer (not a request);
- the ‘E’ command flag set to ‘1’, indicating that an error occurred;
- the Command-Code set to ‘302’, indicating the Diameter ‘Location Info’ command
- the ‘Result-Code’ AVP (268) set to ‘DIAMETER\_REDIRECT\_INDICATION’ (3006);
- the ‘Redirect-Host’ AVP (292) set to the address of the HSS.

After receiving this answer the I-CSCF performs another User Location Query request and sends the Diameter Info Request, which contains the same fields and information as the request sent towards the SLF, besides the ‘Destination-Host’ AVP (293) is now set to the HSS address.

The HSS finds out, that the tel-URL tel:+3396587436 does not belong to a user, but to a service, i.e. the IMRN is a Public Service Identity (PSI – see Section 3.5.5). The address stored for this PSI is not an S-CSCF address, but the address of the VCC AS, which will serve Tobias’s call. This is the address of the same VCC AS that was already

contacted before by the VMSC (see Section 13.3.2), which took the anchoring decision of the call and assigned the IMRN. The HSS returns a Diameter Location Info Answer (LIA) to the I-CSCF, containing:

- the ‘R’ command flag set to ‘0’, indicating that this is an Diameter answer;
- the Command-Code set to ‘302’, indicating the Diameter ‘Location Info’ command
- the ‘Result-Code’ AVP (268) set to ‘DIAMETER\_SUCCESS’ (2001), indicating that the query was successful;
- the ‘Server-Name’ AVP (602) set to the address of Tobiases VCC AS, i.e. vccas1.home1.fr;

Based on the received address in the ‘Server-Name’ AVP, the I-CSCF directly forwards the SIP INVITE request to the VCC AS.

```
INVITE tel:+3396587436 SIP/2.0
Route: <sip:vccas1.home1.fr;lr>
```

The routing example we saw here only applies in the case when the call has originated in the CS domain, as only in that case is an IMRN assigned, which allows, due to PSI routing procedures, that the VCC AS is contacted without sending the SIP INVITE request to the S-CSCF of the originating user (Tobias). If Tobias had originated the call over IMS directly, no IMRN would have been assigned and the SIP INVITE would have been delivered to the S-CSCF based on normal IMS and SIP routing procedures (see Section 12.3). In order to reach the VCC AS in that case, a Filter Criteria would have been needed to be configured at Tobias’s S-CSCF (Section 12.3.8.1). We will see later in this example (Section 13.3.6.3) how the request is routed to the VCC AS based on initial Filter Criteria.

### *13.3.5 Anchoring the Call in Tobias’s Domain*

#### **13.3.5.1 Receiving the SIP INVITE Request on the Initial Access Leg**

The Domain Transfer Function (DTF) within the VCC AS terminates the call and acts as a SIP User Agent Server (UAS) for this incoming SIP INVITE request. Acting as SIP UA give the DTF full control over the call, for example it can at any time generate additional requests on the SIP dialog, it can also terminate or transfer the call at any time. Terminating the SIP dialog locally is the only way for a network entity (in this case the DTF within the VCC AS) to gain full control over the dialog and the multimedia session.

But still the DTF is not the final destination of the call – Tobias intends to set up an audio connection with his sister Theresa. In order to contact Theresa, the DTF needs to set up another call, for which it now acts as the originating endpoint, i.e. the SIP user Agent Client.

The DTF ends up with two independent SIP dialogs, which are only related to each other by the internal logic of the VCC AS. A SIP entity acting for two SIP dialogs in this way is called a Back-to-Back User Agent (B2BUA), as from the outside it looks like it would act as two independent SIP UAs, one terminating the incoming call from Tobias and the second SIP UA originating the call towards Theresa.

The incoming call from Tobias (the CS call that got interworked at the MGCF to SIP) is called the initial access leg. The outgoing call towards Theresa, which the VCC AS now sets up, is called the remote leg. By creating these two independent SIP call legs, the DTF has anchored the call.

### 13.3.5.2 VCC AS requesting Tobiases S-CSCF Name from the SLF / HSS (Subscription Procedures over Dh and Sh Interfaces)

As the SIP INVITE request on the remote leg is set up on behalf of Tobias, the VCC AS needs to make sure, that the request is sent to the S-CSCF that is assigned to Tobias, in order to guarantee that all originating services are performed for this call. At the moment when receiving the incoming SIP INVITE request over the initial access leg, the VCC AS is not aware of the S-CSCF address that is currently assigned for Tobias, it therefore has to query the HSS that holds Tobias's subscription data, in order to find out the address of Tobias's S-CSCF.

As there are several HSS in the network, the VCC AS first needs to query the SLF in order to find out the address of the HSS that holds Tobias's subscription data. The VCC AS performs the Sh-Pull procedures by sending a Diameter User Data Request (UDR) to the SLF with the following content:

- the ‘R’ command flag set to ‘1’, indicating that this is a Diameter request;
- the Command-Code set to ‘306’, indicating the Diameter ‘User-Data’ command
- the ‘User-Identity’ AVP (700) set to Tobias’s Public User Identity, i.e. the received tel URL received in the P-Asserted-Identity header in the INVITE request: tel:+33123456789;
- the ‘Data-Reference’ AVP (703) set to the value ‘S-CSCFName’ (12) – this means that the HSS is requested to return the S-CSCF name that is assigned to the indicated public user identity (i.e. Tobias);
- the ‘Origin-Host’ AVP (264) set the address of Tobias’s VCC AS, i.e. ‘vccas1.home1.fr’;
- the ‘Origin-Realm’ AVP (296) set to the domain name of the operator network in which the VCC AS is located, i.e. ‘home1.fr’;
- the ‘Destination-Realm’ AVP (283) set to the home domain of the SLF/HSS.

As the Destination-Host AVP (293) is not present, the request is routed based on the internal routing configuration of the VCC AS to the SLF which extracts Tobias’s tel-URL from the ‘Public-Identity’ AVP and checks which HSS holds the subscription data for Tobias. The SLF then acts as a Diameter Redirect Agent and returns to the VCC AS the User Data Answer (UDA), containing:

- the ‘R’ command flag set to ‘0’, indicating that this is a Diameter answer (not a request);
- the ‘E’ command flag set to ‘1’, indicating that an error occurred;
- the Command-Code set to ‘306’, indicating the Diameter ‘User Data’ command
- the ‘Result-Code’ AVP (268) set to ‘DIAMETER\_REDIRECT\_INDICATION’ (3006);
- the ‘Redirect-Host’ AVP (292) set to the address of the HSS.

The VCC AS extracts the address of the HSS from the ‘Redirect-Host’ AVP and sends the same UDR as above to the HSS, by adding the ‘Destination Host’ AVP (293) with the address of the HSS.

The HSS now checks whether the VCC AS (as indicated in the ‘Origin-Host’ AVP) is allowed to query the related information and sees that it is allowed. It then sends back an User Data Answer (UDA) to the VCC AS, including:

- the ‘R’ command flag set to ‘0’, indicating that this is an Diameter answer;
- the Command-Code set to ‘306’, indicating the Diameter ‘User Data’ command
- the ‘Result-Code’ AVP (268) set to ‘DIAMETER\_SUCCESS’ (2001), indicating that the query was successful;
- the ‘User-Data’ AVP (702) set to the address of Tobias’s S-CSCF, i.e. ‘scscf1.home1.fr’.

### 13.3.5.3 Setting up the Remote Leg (between the two VCC Application Servers)

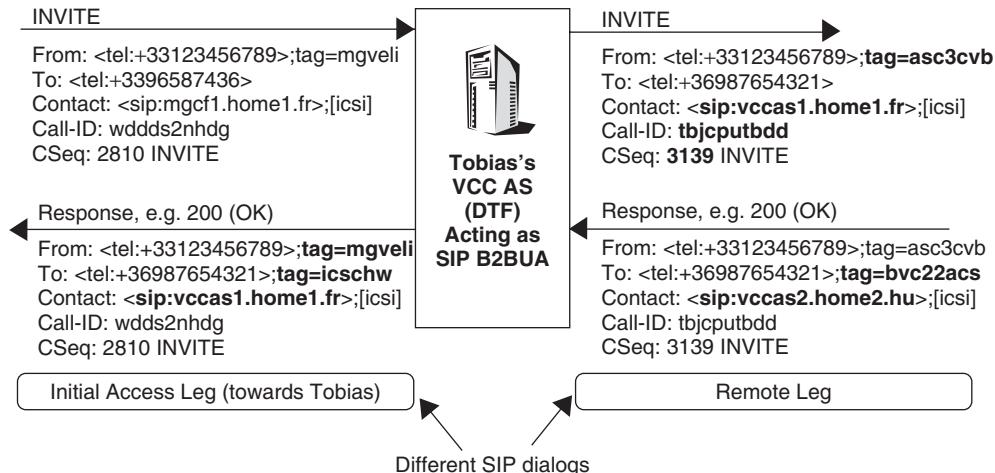
After querying the SLF/HSS the DTF establishes the remote call leg by generating a new SIP INVITE request:

```
INVITE tel:+36987654321 SIP/2.0
Via: SIP/2.0/UDP vccas1.home1.fr
Route: <sip:scscf1.home1.fr;lr>
From: <tel:+33123456789>;tag=asc3cvb
To: <tel:+36987654321>
P-Charging-Vector: icid-value="AyretyU0dm+6";orig-ioi="Type 3
home1.fr"
P-Asserted-Identity: <tel:+33123456789>
P-Asserted-Service: urn:urn-xxx:3gpp-service-
ims.icsci.mmtel
Accept-Contact: *;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-
ims.icsci.mmtel
Contact: <sip:vccas1.home1.fr>
;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-ims.icsci.mmtel"
Call-ID: tbjcpubdd
CSeq: 3139 INVITE
```

This SIP INVITE request is special, as the DTF acts on behalf of Tobias, as the served user. This means, the DTF pretends to be Tobias when generating this request. It is allowed to do so, based on the VCC procedures and the trust relationship between the IMS entities in Tobias’s home network.

In order to uniquely distinguish this SIP dialog from any other SIP dialog, the DTF acting as a B2BUA (see Figure 13.3 and Table 13.3) assigns new values (compared to the incoming SIP INVITE request) to the Call-ID, the CSeq number and the From-Tag. Only by doing so this SIP INVITE request is protocol-wise a new and independent SIP transaction.

The Request URI and the To header of this SIP INVITE request are now set again to the phone number of Theresa. This phone number was stored within the VCC AS when



**Figure 13.3** VCC Anchoring –simplified call flow

the gsmSCF and the CSAF took the anchoring decision (section 13.3.2) and replaced Theresa's phone number with the IMRN. The DTF now just used the stored number and placed it into the Request URI and the To header.

As the DTF acts on behalf of Tobias, it includes a P-Asserted-Identity header, set to the tel-URI of Tobias. The DTF is permitted to put directly a P-Asserted-Identity header into the request, as it is a trusted network entity and the SIP INVITE received on the initial access leg already contained the same value in a P-Asserted-Identity header (set by the MGCF, see Section 13.3.3).

The P-Asserted-Service header, the Accept-Contact header as well as the g.3gpp-icsi.ref feature tag in the Contact header, which both contain the MMTEL ICSI, are copied over from the SIP INVITE request that was received on the initial access leg.

The P-Charging-Vector header is set up by the DTF in the same way as by the MGCF. The DTF is allowed to add the P-Charging-Vector header to an originating request when it is acting on behalf of a user.

As the VCC AS is the SIP endpoint of the dialog, it puts its own address into the Contact header. By doing so it makes sure to receive all requests that are sent from the other side of the SIP dialog towards Tobias.

Also the address in the Via header is set to the address of the VCC AS, by which the DTF guarantees that it will receive all responses sent for this SIP INVITE request.

The Route header points to the next hop, to which the VCC AS will send the request, i.e. to the S-CSCF name received from the HSS in the User-Data AVP within the User Data Answer (UDA).

In order to establish the voice media connection between Tobias and Theresa, the SIP INVITE request includes the following SDP Offer:

```
v=0
o=- 999777888 47 IN IP6 5555::36:74:58:96
s=-
c=IN IP6 5555::36:74:58:96
```

```
t=907165566 0
m=audio 3458 RTP/AVP 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local sendrecv
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv
a=tcap:1 RTP/AVP RTP/AVPF
a=pcfg:1 t=1||2
```

This SDP Offer is simply copied over from the SIP INVITE request that was received on the initial access leg. As the VCC AS does not handle any media locally and does also not control any media resource functions, the end point for the media session is still the MGW that is indicated in the c-line of the SDP. Also the media parameters did not change.

### 13.3.5.4 Further Processing on the Initial Access Leg (towards Tobias)

Whilst setting up the remote access leg (towards Theresa), Tobias's VCC AS still handles the incoming access leg (from Tobias and the MGCF) dialog. Once it received the INVITE request from the access leg, it sent back a 100 (Trying) response towards the MGCF.

Further SIP messages on the initial access leg will be sent, based on the incoming SIP responses on the remote call leg.

#### 13.3.6 Forwarding the Call to Theresa's Domain

##### 13.3.6.1 Service Provisioning for Unregistered User (Tobias)

The SIP INVITE request is now sent from the VCC AS to the S-CSCF that serves Tobias. As Tobias is currently only active from his mobile phone, his public user identity is not registered within the IMS. Nevertheless an S-CSCF is assigned for him and this S-CSCF has downloaded his User Profile, as specific services need to be executed also for unregistered users. The incoming SIP INVITE request is now checked in the S-CSCF against all filter criteria (see Section 12.3.8). During this operation, the request might be sent to other Application Servers as well, especially the Telephony Application Server (TAS – see Section 12.3.8).

##### 13.3.6.2 Resolving a tel-URL

After evaluating all filer criteria, the S-CSCF needs to send the SIP INVITE request towards the IMS home network of Theresa. The only routing information available in the request is the content of the request URI, which contains Theresa's tel-URL. As the phone number is not known at Tobias's S-CSCF, it needs to resolve the number via an ENUM Server, in order to find the address that the request needs to be forwarded to next.

The S-CSCF contacts the ENUM server in the same way as a DNS Server (see Section 11.4.2), but needs to translate the tel-URL into a format, that can be understood by the server in order to resolve it. The S-CSCF therefore reverts the digits of the tel-URL, separates

**Table 13.3** SIP dialogs at Tobias's VCC AS (B2BUA)

	initial access leg (from Tobias's MGCF)	new access leg (from Tobias's phone)	remote leg (towards Theresa)
<b>initial Request URI</b>	+3396587436 (IMRN)	vdi-vccas.home1.fr (VDI)	+36987654321 (Theresa)
<b>To-Address</b>	+3396587436 (IMRN)	vdi-vccas.home1.fr (VDI)	+36987654321 (Theresa)
<b>From-Address</b>	+33123456789 (Tobias)	+33123456789 (Tobias)	+33123456789 (Tobias)
<b>To-Tag</b>	mgveli	idvkoan	asc3cvb
<b>From-Tag</b>	icschw	osfpm	bvc22acs
<b>local P-Asserted-Identity (set by VCC AS)</b>	+36987654321 (Theresa)	+36987654321 (Theresa)	+33123456789 (Tobias)
<b>remote P-Asserted-Identity (received from remote side)</b>	+33123456789 (Tobias)	+33123456789 (Tobias)	+36987654321 (Theresa)
<b>Call-ID</b>	wddds2nhdg	fea8x7 - p	tbicputbdd
<b>Local Contact Address (set by VCC AS)</b>	vccas1.home1.fr	vccas1.home1.fr	vccas1.home1.fr

Table 13.3 (continued)

		initial access leg (from Tobias's MGCF)	new access leg (from Tobias's phone)	remote leg (towards Theresa)
<b>Remote Contact Address</b> (received from remote side)	after anchoring	mgcf1.home1.fr	-	vccat2.home2.hu
	after first domain transfer (Tobias CS -> IMS)	-	[5555::1:2:3:4] (Tobias's phone)	vccat2.home2.hu
	after second domain transfer (Theresa IMS -> CS)	-	[5555::1:2:3:4] (Tobias's phone)	vccat2.home2.hu
<b>SDP audio codec</b>	after anchoring	AMR-WB	-	AMR-WB
	after first domain transfer (Tobias CS -> IMS)	-	AMR-WB	AMR-WB
	after second domain transfer (Theresa IMS -> CS)	-	PCMU	PCMU
<b>SDP connection address</b> (c-line) received from remote side	after anchoring	[5555::36:74:58:96] (Tobias's MGW)	[5555::5:6:7:8] (Theresa's phone)	-
	after first domain transfer (Tobias CS -> IMS)	[5555::1:2:3:4] (Tobias's phone)	[5555::5:6:7:8] (Theresa's phone)	-
	after second domain transfer (Theresa IMS -> CS)	[5555::12:3:4] (Tobias's phone)	-	[5555::71:82:93: 64] (Theresa's MGW)

Table 13.3 (continued)

		initial access leg (from Tobias's MGCF)	new access leg (from Tobias's phone)	remote leg (towards Theresa)
<b>SDP connection address (c-line) received from remote side</b>	after anchoring	[5555::36:74:58:96] (Tobias's MGW)	-	[5555::5:6:7:8] (Theresa's phone)
	after first domain transfer (Tobias CS -> IMS)	-	[5555::1:2:3:4] (Tobias's phone)	[5555::5:6:7:8] (Theresa's phone)
	after second domain transfer (Theresa IMS -> CS)	-	[5555::1:2:3:4] (Tobias's phone)	[5555::71:82:93: 64] (Theresa's MGW)
<b>SDP connection address (c-line) sent from VCC AS</b>	after anchoring	[5555::5:6:7:8] (Theresa's phone)	-	[5555::36:74:58: 96] (Tobias's MGW)
	after first domain transfer (Tobias CS -> IMS)	-	[5555::5:6:7:8] (Theresa's phone)	[5555::1:2:3:4] (Tobias's phone)
	after second domain transfer (Theresa IMS -> CS)	-	[5555::71:82:93:64] (Theresa's MGW)	[5555::1:2:3:4] (Tobias's phone)
<b>Next Hop</b>	first INVITE request	Tobias's I-CSCF	Tobias's S-CSCF	Tobias's S-CSCF
	after successful session setup	Tobias's MGCF	Tobias's S-CSCF	Tobias's S-CSCF

the digits by dots and puts the domain ‘enum.arpa’ in the end, i.e. the tel-URL tel:+369876 54321 is transformed into the domain name 1.2.3.4.5.6.7.8.9.6.3.enum.arpa. It then sends a DNS NAPTR query to the local ENUM server:

```
DNS query - NAPTR
Name: 1.2.3.4.5.6.7.8.9.6.3.enum.arpa
```

The local ENUM server does not know the IP address, under which Theresa is currently registered, but it has a list of services configured for Theresa’s home network, i.e. for the network identified by ‘7.8.9.6.3.enum.arpa’, i.e. the network operator identified by the dial string ‘987’ in Hungary (which is identified by the national code +36”). The ENUM Server now returns the list of configured services:

```
DNS response - 1.2.3.4.5.6.7.8.9.6.3.enum.arpa
IN NAPTR 100 11 "u" "E2U+sip" "" _sip._udp.home2.hu
IN NAPTR 100 10 "u" "E2T+sip" "" _sip._tcp.home2.hu
IN NAPTR 101 10 "u" "E2U+h323" "" _h323._udp.home2.hu
```

This shows that there are three services that allow the S-CSCF to reach the home domain of the indicate phone number, i.e. via:

- a SIP over UDP (‘E2U+sip’) service provided by ‘\_sip.\_udp.home2.hu’, which
  - is first in order (‘100’), which is the same order as the next entry has; as the order is processed from the lowest available value to the highest, the order indicates here that the first and second entry shall be treated first (if possible for the S-CSCF), before the third entry (order of ‘101’) should be treated;
  - has a higher preference (‘11’) than the following entry of the same order, which means that this service shall be tried first (if possible for the S-CSCF), before the other services are tried;
  - indicates that the resolution of this entry will result in a server or a list of servers which can be directly contacted (‘u’ flag);
- a SIP over TCP (‘E2T+sip’) service provided by ‘\_sip.\_tcp.home2.hu’;
- and a H323 service over UDP (‘E2U+h323’) service provided by ‘\_h323.\_udp.home2.hu’.

Based on the given order and preference, that S-CSCF must, if it supports SIP over UDP, try to contact the first entry. As SIP over UDP is supported by Tobias’s S-CSCF, it sends the following DNS SRV query to the DNS server:

```
DNS query - SRV
Name: _sip._udp.home2.hu
```

The DNS server does not know by itself which servers are configured for Theresa’s home network and therefore forwards the query to the DNS server in the domain home2.hu,

which responds with the list of I-CSCFs in Theresa's home network as well as their port numbers, on which they are prepared to receive SIP via UDP:

```
DNS response - _sip._udp.home2.hu
IN SRV 0 1 5060 icscf1.home2.hu
IN SRV 0 0 5060 icscf2.home2.hu
IN SRV 5 1 5060 icscf6.home2.hu
```

The response shows, that the home domain of Theresa is reachable via three different I-CSCFs:

- icscf1.home2.hu on port 5060, which
  - has the lowest priority number ('0'), which means it is of highest priority. It shares this priority with the second entry. The S-CSCF shall, if possible, make use of the entries with the highest priority first;
  - has the highest weight ('1') within the two entries with the same priorities. The S-CSCF shall, if possible, make use of the entries with the highest weight within the highest priority first;
- icscf2.home2.hu on port 5060;
- and icscf6.home2.hu on port 5060.

The S-CSCF therefore must contact Theresa's network over the I-CSCF in the first entry. Based on this information Tobias's S-CSCF only knows the host name of Theresa's I-CSCF, but in order to send the INVITE request towards it, the S-CSCF needs the IP address of the I-CSCF, as this is required on IP Layer in order to reach the I-CSCF. The S-CSCF therefore queries the DNS server again in order to obtain the IPv6 address of the I-CSCF in Theresa's home network:

```
DNS query - AAAA
Name: icscf1.home2.hu
```

Also this query is forwarded by the local DNS server to the DNS server in Theresa's domain. Theresa's DNS server sends the result of the query to Tobias's DNS server, which forwards it to Tobias's S-CSCF:

```
DNS response
IN AAAA 5555::cc:dd:aa:12
```

With the IPv6 address received in the DNS response, the S-CSCF can forward the SIP INVITE request to the I-CSCF in Theresa's network with the following information set:

- on SIP level optionally a Route header, indicating 'sip:icscf1.home2.hu';
- in the UDP packet (UDP is chosen based on the DNS NAPTR response) the destination port set to '5060', which was indicated in the DNS SRV response;
- in the IP packet the destination address set to '5555::cc:dd:aa:12', as it was resolved with the DNS AAAA query.

### 13.3.6.3 From Tobias's S-CSCF to Theresa's VCC AS

With this information, the S-CSCF can send the SIP INVITE request to the I-CSCF `icscf2.home2.hu`, which is the entry point into Theresa's network. The I-CSCF has to find out, which S-CSCF serves the user indicated in the request URI of the SIP INVITE request. It therefore sends a User Location Query request via the Dx Interface to the local SLF, which then responds with a DIAMETER redirect message, pointing to the HSS that holds Theresa's subscription information. The I-CSCF sends the User Location Query request to that HSS and gets back the address of the S-CSCF at which Theresa's registration is handled (see Section 12.3.3.4). The I-CSCF sends the SIP INVITE request to Theresa's S-CSCF.

Upon receiving the SIP INVITE request, the S-CSCF applies the filter criteria for the terminating case, which it has downloaded with Theresa's user profile from the HSS when Theresa was registering. Based on these filter criteria, the SIP INVITE request might be forwarded to different Application Servers, e.g. the Telephony Application Server (TAS). One of these Filter Criteria triggers for incoming SIP INVITE requests and points to the VCC AS that serves Theresa (`vccas2.home2.hu`). As the Filter Criteria matches, the SIP INVITE request is forwarded to the VCC AS.

### 13.3.7 Anchoring the call in Theresa's Domain

#### 13.3.7.1 VCC AS receives INVITE request on remote call leg

The SIP INVITE request is received by Theresa's VCC AS with the following SIP headers (again only those are shown, which are relevant for this example):

```
INVITE tel:+36987654321 SIP/2.0
Via: SIP/2.0/UDP scscf.home2.hu, SIP/2.0/UDP tas.home2.hu, SIP/2.0/UDP
scscf2.home2.hu, SIP/2.0/UDP icscf.home2.hu, SIP/2.0/UDP scscf1.home1.fr,
SIP/2.0/UDP tas.home1.fr, SIP/2.0/UDP scscf1.home1.fr, SIP/2.0/UDP
vccas1.home1.fr
Record-Route: sip:scscf2.home2.hu;lr, sip:tas.home2.hu;lr,
sip:scscf2.home2.hu;lr, sip:scscf1.home1.fr;lr, sip:tas.home1.fr;lr,
sip:scscf1.home1.fr;lr
Route: sip:vccas2.home2.hu;lr, sip:scscf2.home2.hu;lr;orig=75tiopotts
From: <tel:+33123456789>;tag=asc3cvb
To: <tel:+3612323454>
P-Charging-Vector: icid-value="AyretyU0dm+6"
P-Asserted-Identity: <tel:+33123456789>
P-Asserted-Service: urn:urn-xxx:3gpp-service-ims.icsi.mmTEL
Contact: <sip:vccas1.home1.fr>
;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-ims.icsi.mmTEL"
Call-ID: tbjcpubdd
CSeq: 3139 INVITE
```

The Via header shows which switches have received the INVITE request so far (read in reverse order, as every SIP entity puts itself on top of the Via-list):

- Tobias's VCC AS (`vccas1.home1.fr`) has initiated the remote call leg, i.e. it is the originator of the INVITE request;

- As seen in Section 13.3.5.2 Tobias's VCC AS determined the S-CSCF (scscf1.home1.fr) that serves Tobias (in the unregistered case) and sent the INVITE down to it;
- The S-CSCF then ran through the filter criteria of Tobias and found a matching Filter Criteria for Tobias's Telephony Application Server (tas.home1.fr), to which it forwarded the request;
- After providing the relevant services, the TAS sent back the INVITE request to Tobias's S-CSCF (scscf1.home1.fr);
- As no more Filter Criteria did match, Tobias's S-CSCF forwarded the INVITE towards Theresa's network, for which it resolved (see Section 13.3.6.1) the tel-URL to an I-CSCF address (icscf2.home2.hu);
- The I-CSCF resolved Theresa's S-CSCF (scscf2.home2.hu) and forwarded the INVITE there;
- Theresa's S-CSCF applied the initial Filter Criteria for the terminating case and also forwarded the request to Theresa's TAS (tas.home2.hu), which provided the relevant services and sent the INVITE back to Theresa's S-CSCF (scscf2.home2.hu).

The Record-Route header shows nearly the same list as in the Via header, with the following exceptions:

- Tobias's VCC AS (vccas1.home1.fr) is not listed here, as it is the end point of the SIP dialog (the remote access leg) and therefore its address is provided in the Contact header
- Theresa's I-CSCF (icscf2.home2.hu) is not listed here, as it is only needed for querying the SLF / HSS to find out Theresa's S-CSCF, i.e. it does not need to stay on the dialog.

The Route header shows that, based on the matching Filter Criteria in the S-CSCF, the INVITE request first is sent to Theresa's VCC AS (vccas2.home2.hu) and then must be sent back to Theresa's S-CSCF (scscf2.home2.hu). Note that the S-CSCF included an orig Parameter in its own address in the Route header, which allows it to relate an incoming SIP INVITE request to the current one, even if the dialog information (Call-ID, To- and From-Tags) should have been changed.

The other headers did not change since Tobias's VCC AS has send out the INVITE request.

The SDP included in this SIP message is the same as shown above (see Section 13.3.3.3).

### 13.3.7.2 Domain Selection

Theresa's VCC AS now must determine, how the call should be delivered to Theresa. There are three possibilities:

- via IMS – this is only possible if Theresa has registered with her S-CSCF;
- via CS – this is only possible if Theresa is currently attached via the CS domain;
- Theresa is not reachable via IMS or CS and therefore the call needs to be answered by other means (e.g. voice-mailbox in the network).

In our example, Theresa is registered via IMS and also attached via the CS domain. It is up to Theresa's network operator policy how to determine whether to forward the call

via IMS or via CS to Theresa. The domain selection decision is made by the Domain Selection Function (DSF) which is also part of Theresa's VCC AS (see Section 3.20). In this example, the DSF decides to deliver the call to Theresa via the IMS domain.

### 13.3.7.3 Establishing the Initial Access Leg Towards Theresa

Theresa's VCC AS now anchors the call by acting as a B2BUA (see Table 13.4 as well as Section 13.3.5), i.e. it terminates the incoming call, the so-called remote leg, which originates from Tobias's VCC AS, and establishes a new initial access leg towards Theresa's phone. It therefore sends a new SIP INVITE request towards the S-CSCF of Theresa:

```
INVITE tel:+36987654321 SIP/2.0
Via: SIP/2.0/UDP vccas2.home2.hu
Route: <sip:scscf2.home2.hu;lr;orig=75tiopotts> From:
<tel:+33123456789>;tag=thvas2rez To: <tel:+3612323454>
P-Charging-Vector: icid-value="AyretyU0dm+6" P-Asserted-Identity:
<tel:+36987654321> P-Asserted-Service:
urn:urn-xxx:3gpp-service-ims.icsi.mmtel Contact:
<sip:vccas2.home2.hu
;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-ims.icsi.mmtel"
Call-ID: blcpgrsws CSeq: 6412 INVITE
```

Again, this is a completely new SIP INVITE request which will establish a third SIP dialog, which is independent of the the initial access leg between the MGCF and the VCC AS on Tobias's side and the remote leg between Tobias's and Theresa's VCC ASes. Therefore Call-ID, CSeq and From-tag have new globally unique values.

The Route header points to the S-CSCF serving Theresa. As the SIP INVITE request on the remote call leg was received via this S-CSCF, the VCC AS learned its address from that request and therefore does not need to resolve it via SLF/HSS. Note that the same orig-parameter is included here as it was received within the incoming INVITE request on the remote call leg. This is necessary to allow the S-CSCF to correlate the incoming INVITE request with the one it sent to Theresa's VCC AS.

The Via and Record-Route headers have been removed, as they belong to the remote call leg.

The request URI and the To header values are set to Theresa's tel-URL, which was dialled by Tobias. If the DSF (see Section 13.3.7.2) would have decided to deliver the call via the CS domain to Theresa, the request URI would be set to a newly assigned number, the CS Domain Routing Number (CSRN) which would route the INVITE request towards Theresa's MGCF in order to interwork the call to the CS domain. The handling of the CSRN is not shown in this example.

The Contact header points to Theresa's VCC AS, as it is the endpoint for the SIP dialog. It also includes the ICSI in the feature tag, as it was set by Tobias's MGCF (see Section 13.3.3), in order to indicate Tobias's capabilities to Theresa's phone.

The From, P-Asserted-Identity, P-Asserted-Service and Accept-Contact headers did not change, they are still the same as set by Tobias's MGCF (see Section 13.3.3) and therefore are copied over from the incoming INVITE request.

Also the SDP of the original message is copied from the original INVITE request, i.e. this as well is the same as indicated by the MGCF (see Section 13.3.3).

**Table 13.4** SIP dialogs at Theresa's VCC AS (B2BUA)

	remote Leg towards Tobias)	initial access leg (towards Theresa's <b>P-CSCF</b> )	new access leg (from Theresa's <b>MGCF</b> )
<b>initial Request</b>			
<b>URI (incoming)</b>	+36987654321 (Theresa)	+36987654321 (Theresa)	+3663478569 (IMRN)
<b>To-Address</b>	+36987654321 (Theresa)	+36987654321 (Theresa)	+3663478569 (IMRN)
<b>From-Address</b>	+33123456789 (Tobias)	+33123456789 (Tobias)	+36987654321 (Theresa)
<b>To-Tag</b>	asc3cvb	thvas2ez	gwiwdbbe
<b>From-Tag</b>	byrc22aas	ake1s9ls	sneuwidtw
<b>local P-Asserted-Identity (set by VCC AS)</b>	(Theresa)	(Tobias)	(Tobias)
<b>remote P-Asserted- Identity (received from remote side)</b>	+36987654321 (Tobias)	+33123456789 (Theresa)	+33123456789 (Theresa)
<b>Call-ID</b>	tbicputbdd	blcpersws	aazginav
<b>Local Contact Address (set by VCC AS)</b>	vccas2.home2.hu	vccas2.home2.hu	vccas2.home2.hu

**Table 13.4** (continued)

		remote Leg towards Tobias)	initial access leg (towards Theresa's P-CSCF)	new access leg (from Theresa's MGCF)
<b>Remote Contact Address (received from remote side)</b>	<b>after anchoring</b>	vccas1.home1.fr	[5555::5:6:7:8]: 1006 (Theresa's phone)	-
	<b>after first domain transfer (Tobias CS -&gt; IMS)</b>	vccas1.home1.fr	[5555::5:6:7:8]: 1006 (Theresa's phone)	-
	<b>after second domain transfer (Theresa IMS -&gt; CS)</b>	vccas1.home1.fr	-	mgcf2.home2. hu
<b>SDP audio codec</b>	<b>after anchoring</b>	AMR-WB	AMR-WB	
	<b>after first domain transfer (Tobias CS -&gt; IMS)</b>	AMR-WB	AMR-WB	
	<b>after second domain transfer (Theresa IMS -&gt; CS)</b>	PCMU	PCMU	
<b>SDP connection address (c-line) sent from VCC AS</b>	<b>after anchoring</b>	[5555::5:6:7:8] (Theresa's phone)	[5555::36:74:58: 96] (Tobias's MGW)	
	<b>after first domain transfer (Tobias CS -&gt; IMS)</b>	[5555::5:6:7:8] (Theresa's phone)	[5555::1:2:3:4] (Tobias's phone)	
	<b>after second domain transfer (Theresa IMS -&gt; CS)</b>	[5555::7:1:82:93:64] (Theresa's MGW)	-	[5555::1:2:3:4] (Tobias's phone)
<b>Next Hop</b>	<b>first INVITE request</b>	Theresa's S-CSCF	Theresa's S-CSCF	Theresa's I-CSCF
	<b>after successful session setup</b>	Theresa's S-CSCF	Theresa's S-CSCF	Theresa's MGCF

### 13.3.7.4 Further Actions on the Remote Call Leg

Theresa's VCC AS sends back a 100 (Trying) on the remote call leg. Further SIP messages on the remote call leg will be sent based on the incoming messages on the initial access call leg towards Theresa.

#### 13.3.8 Delivering the Call to Theresa

Theresa's S-CSCF receives the INVITE from the VCC AS and finds out, based on the orig-parameter included in the Route header, that it is correlated to the INVITE request it has sent out before (see Section 13.3.6.3). Due to this, the S-CSCF tries to go on with the evaluation of Theresa's Filter Criteria for the terminating case. As there is no further Filter Criteria available, the S-CSCF performs the actions necessary to route the request to Theresa's phone.

The S-CSCF first needs to determine the caller preferences expressed in the Accept-Contact header (\*;g.3gpp-icsi.ref = "urn%3Aurn-xxx%3A3gpp-service-ims.icsi.mmTEL"), in order to find out, to which of the phones, from which Theresa is currently registered, the call should be delivered. For simplification we assume that Theresa currently is only registered from her mobile phone, i.e. there is only one contact address ([5555::5:6:7:8]:1006) available for her and the phone indicated, during registration, the MMTEL ICSI in the Contact header. Therefore the call should be routed to this device.

After determining this, the S-CSCF

- rewrites the request URI of the INVITE request with the selected Contact address ([5555::5:6:7:8]:1006), in order to indicate the physical address of the final destination of the INVITE request;
- adds a P-Called-Party-ID header with the URI received in the incoming INVITE (tel:+3612323454 – see Section 12.2.4.2);
- puts itself onto the recorded route, in order to stay on the dialog for all subsequent requests sent on the dialog;
- puts the address of the P-CSCF, that was indicated in the REGISTER request from Theresa's phone within the Path header, into the Route header, in order to make sure that the INVITE request first gets routed to the P-CSCF before it is sent to the final destination (the Contact address in the request URI), as the request must be sent over the IPsec Security Association established between the P-CSCF and the UE (see Section 11.7).
- sends the INVITE request out, towards the P-CSCF.

```

INVITE [5555::5:6:7:8]:1006 SIP/2.0
Via: SIP/2.0/UDP vccas2.home2.hu, SIP/2.0/UDP scscf2.home2.hu
Route: <sip:pcscf2.home2.hu;lr>
Record-Route: <sip:scscf2.home2.hu;lr>
From: <tel:+33123456789>;tag=thvas2rez
To: <tel:+3612323454>
P-Charging-Vector: icid-value="AyretyU0dm+6";term-ioi=home2.hu
P-Asserted-Identity: <tel:+36987654321>
P-Asserted-Service: urn:urn-xxx:3gpp-service-ims.icsi.mmTEL

```

```

P-Called-Party-ID: tel:+36987654321
Contact: <sip:vccas2.home2.hu>
;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-ims.icsi.mmTEL"
Call-ID: blcpgrsws
CSeq: 6412 INVITE

```

Theresa's P-CSCF now performs the normal procedures, as described in Section 12.3.3.2 and forwards the SIP INVITE request to Theresa's phone.

Note that both, Theresa's P-CSCF as well as her S-CSCF, are not aware of the call anchoring and there are no specific procedures needed in any CSCF in order to make VCC work within IMS. All VCC related actions and service logic is provided by the UEs and the VCC ASes.

### 13.3.9 Establishing the End-to-End Call

#### 13.3.9.1 Response over the initial access leg towards Theresa

Theresa's phone, upon receiving the initial INVITE request does not need to perform any VCC specific actions. It performs the normal IMS session establishment procedures, as they are described in Chapter 11. In our example, Theresa's phone needs to reserve the resources needed for the call. It can do this without sending out any further SIP response for the INVITE request, as the A-side (Tobias) indicated, that all resources are available (see Section 12.9.2).

After successful resource reservation, Theresa's phone will send back a 180 (Ringing) response. Once Theresa accepts the call, the phone will send out a 200 (OK) response. As the 180 (Ringing) response and the 200 (OK) response will be routed in the same way, we will only show the handling of the 200 (OK) response in this example.

The 200 OK Response sent from Theresa's phone looks as follows:

```

SIP/2.0 200 OK
Via: SIP/2.0/UDP pcscf2.home2.hu,
      SIP/2.0/UDP vccas2.home2.hu
Record-Route: sip:pcscf2.home2.hu;lr, sip:scscf2.home2.hu;lr
From: <tel:+33123456789>;tag=thvas2rez
To: <tel:+3612323454>;tag=ake1s91s
P-Preferred-Identity: <tel:+36987654321>
Contact: <sip:[5555::5:6:7:8]:1006><
;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-ims.icsi.mmTEL"
Call-ID: blcpgrsws
CSeq: 6412 INVITE

```

This response is sent over the intitial access leg towards Theresa within the dialog between Theresa's VCC AS and Theresa's phone. It includes:

- the P-Preferred-Identity header set to Theresa's tel-URL, which is one of Theresa's public user identities. Theresa's phone could also have put a different public user identity there, e.g. a SIP URI, but for this example we assume the phone went with the URL received in the To header. Upon receipt of this 200 (OK) response, Theresa's P-CSCF will make sure that the indicated tel-URL belongs to Theresa and replaces the

P-Preferred-Identity header with the P-Asserted-Identity header with the same value (see Section 12.2.3.2);

- the list of Via headers (for routing the response) and Record-Route headers (for routing subsequent requests, such as e.g. ACK and BYE) that were collected on the initial access leg towards Theresa, i.e. from Theresa's VCC AS to Theresa's phone;
- the CSeq, Call-ID, From-Tag values assigned by Theresa's VCC AS for the INVITE over the initial access leg towards Theresa;
- the To-Tag in the To header, that is added as per normal RFC 3261 procedures;
- the IP address of Theresa's phone in the Contact header, as the phone is the SIP endpoint (User Agent) for this dialog;

The 200 (OK) response also includes a SDP Answer;

```
v=0
o= - 1357924 1357924 IN IP6 5555::5:6:7:8
s=- 
c=IN IP6 5555::5:6:7:8
t=907165275 0
m=audio 4011 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event

a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
a=acfg:1 t=2
```

In this SDP Answer, Theresa's phone indicates:

- indicates its connection address (5555::5:6:7:8) and the port (4011) over which it wants to receive the audio media;
- that AVPF will be used, as this was indicated as an option in the received SDP Offer. Theresa's phone therefore indicates 'RTP/AVPF' in the m-line and adds the 'a = acfg:1 t = 2' attribute line for the SDP capability negotiation;
- indicates that the resources have been successfully reserved on Theresa's side ('curr:qos local' and 'des:qos local' attribute lines are matching).

The 200 (OK) response is routed along the Via headers until it reaches Theresa's VCC AS.

### 13.3.9.2 Response over the remote leg between the two VCC Application Servers

Upon receiving the 200 (OK) response from Theresa's phone, the DTF in Theresa's VCC AS acts as a B2BUA, i.e.

- it terminates the received 200 (OK) response on the initial call leg from Theresa by acting as a SIP User Agent Client;

- it sends a different 200 (OK) over the SIP dialog on the remote call leg towards Tobias's VCC AS. This SIP dialog was initiated by Tobias's VCC AS (see Section 13.3.5.3).

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf2.home2.hu,
      SIP/2.0/UDP tas.home2.hu, SIP/2.0/UDP
scscf2.home2.hu, SIP/2.0/UDP icscf2.home2.hu, SIP/2.0/UDP
scscf1.home1.fr, SIP/2.0/UDP tas.home1.fr,
      SIP/2.0/UDP scscf1.home1.fr,
SIP/2.0/UDP vccas1.home1.fr
Record-Route: sip:scscf2.home2.hu;lr, sip:tas.home2.hu;lr,
sip:scscf2.home2.hu;lr, sip:scscf1.home1.fr;lr,
sip:tas.home1.fr;lr,
sip:scscf1.home1.fr;lr
From: <tel:+33123456789>;tag=asc3cvb
To: <tel:+3612323454>;tag=bvc22acs
P-Asserted-Identity: <tel:+36987654321>
Contact: <sip:vccas2.home2.hu>|
          ;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-ims.icsi.mmTEL"
Call-ID: tbjcpusbdd
CSeq: 3139 INVITE
```

Note that the following values in particular are different from the 200 (OK) response that was sent from Theresa's phone:

- the list of Via and Record-Route entries are set to the list collected over the remote call leg;
- Call-ID, CSeq, From-Tag are those of the dialog over the remote call leg (as assigned initially by Tobias's VCC AS, see Section 13.3.5.3) and a new To-Tag is assigned;
- the Contact header is set to the address of Theresa's VCC AS, as it is the endpoint of this SIP dialog.

Nevertheless, there are several values in this 200 (OK) response that are copied over from the 200 (OK) response that was received from Theresa's phone:

- the P-Asserted-Identity, indicating Theresa's tel-URL;
- the feature tags in the Contact header, in this example indicating the IMS Multimedia Telephony ICSI;
- the SDP Answer.

The 200 (OK) response is routed along the Via header until it reaches Tobias's VCC AS.

### 13.3.9.3 Response Over the Initial Access Leg Towards Tobias

Upon receiving the 200 (OK) response on the remote leg, Tobias's VCC AS also starts performing the SIP B2BUA procedures, i.e. it terminates the received 200 (OK) over the

remote call leg and sends out a new 200 (OK) over the initial access leg towards Tobias's phone, which looks as follows:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP icscf1.home1.fr, SIP/2.0/UDP mgcf1.home1.fr
From: <tel:+33123456789>;tag= mgveli To:
<tel:+3612323454>;tag= icschw P-Asserted-Identity:
<tel:+36987654321> Contact: <vccas1.home1.fr><
;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-ims.icsi.mmtel"
Call-ID: wddds2nhdg CSeq: 2810 INVITE
```

The differences between the received and the sent 200 (OK) responses are similar to those as described above in Section 13.3.9.2. Note especially that the SDP Answer is once again just copied over from the original response.

The response is routed along the Via headers until it reaches Tobias's MGCF.

#### 13.3.9.4 Response Interworking at the MGCF and Establishing the Call on CS Side Towards Tobias

Upon receiving the 200 (OK) response from the VCC AS, Tobias's MGCF sends out an ISUP Answer Message (ANM) towards Tobias's phone and returns immediately a SIP ACK request to Tobias's VCC AS. The SIP ACK request is sent along the recorded routes of the three dialogs until it reaches Theresa's phone, the VCC Application Servers perform the SIP B2BUA actions as described in this Section for the INVITE request and the 200 (OK) response.

The ISUP ANM is delivered to Tobias's VMSC and triggers there a CS Connect message to be sent to Tobias's phone. The phone responds with a CS Connect Acknowledge message, which is terminated at the VMSC.

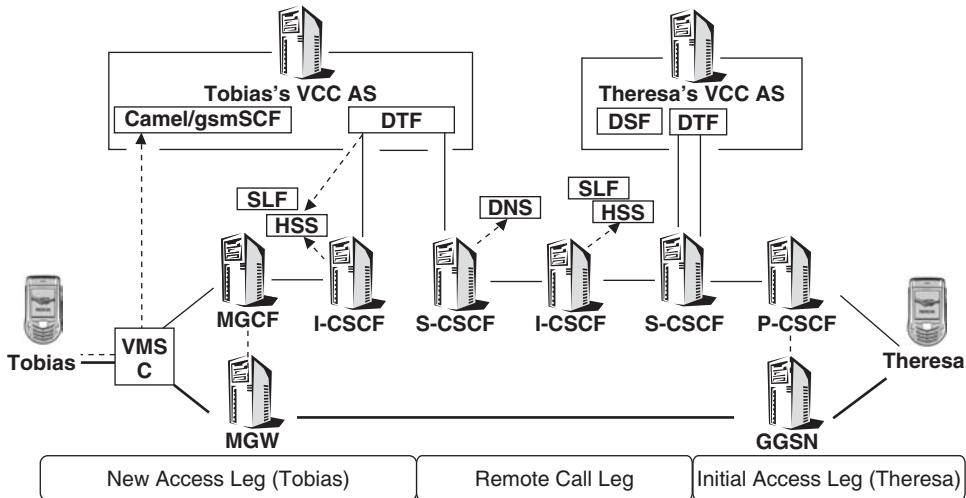
#### 13.3.10 Scenario After Anchoring

Figure 13.4 shows the existing connections between the different entities after VCC anchoring has performed.

On the signalling level, the call is split into different segments:

- the CS connection (TS 24.0008 protocol) between Tobias's phone and the VMSC;
- the SS7/ISUP connection between the VMSC and the MGCF;
- the SIP dialog over the initial access leg from Tobias's MGCF to Tobias's VCC AS;
- the SIP dialog over the remote leg from Tobias's VCC AS to Theresa's VCC AS – note that this SIP dialog is different from the one over the initial access leg (MGCF <-> VCC AS);
- the SIP dialog over the initial access leg from Theresa's VCC AS to Theresa's phone – note that this SIP dialog is again different from the other two SIP dialogs.

On a media level, the Tobias's phone sends all the media to the MGW (which is controlled by the MGCF), which sends the media to Theresa's GGSN, which then forwards it to Theresa's phone.



**Figure 13.4** VCC –connections after anchoring

What was described so far was the normal call setup between Tobias and Theresa. The calls are only anchored in the VCC Application Servers, but none of the users has changed the communication domain. For a great number of calls, the VCC domain transfer procedures will never be applied, as the users do not move in and out the different communication domains. In this example we assume, that domain transfers occur on both sides. We start with Tobias, whose phone switches over from CS to IMS.

## 13.4 Domain Transfer: CS to IMS

### 13.4.1 Tobias's Phone Invokes VCC Procedures

After the call between Tobias and his sister has been ongoing for a while, Tobias enters a café, at which his network operator provides I-WLAN access. Tobias's phone detects, that I-WLAN and therefore also IMS access are available. The phone connects to I-WLAN and afterwards performs an IMS registration to Tobias's home network (similar to the registration described in Chapter 11). This all happens whilst Tobias's CS call with his sister is still ongoing.

After successful IMS registration, the phone now must immediately perform a VCC domain transfer of the ongoing call from CS to IMS, as in the Communication Continuity MO (see Section 13.2) the following parameters are set:

- Preferred Domain: ‘1’ – indicating that Tobias’s preferred domain is IMS;
- Immediate domain transfer: ‘1’ – indicating that domain transfer should be done immediately once the preferred domain (IMS) becomes available;
- DT CS-to-IM CN direction: ‘0’ – indicating that the CS to IMS domain transfer is allowed.

The phone now sends out an SIP INVITE request which is destined not to Theresa's phone, but to the VCC Domain Transfer URI (VDI), which also was provided by the Communication Continuity MO. The VDI is a Public Service Identity (PSI – see Section 3.5.5) which points to Tobias's VCC AS.

```
INVITE sip:vdi-vccas.home1.fr SIP/2.0
Via: SIP/2.0/UDP [5555::1:2:3:4]:1357
Route: sip:pcscf1.home1.fr;lr, sip:scscf1.home1.fr;lr
From: <tel:+33123456789>;tag=osftpm
To: <sip:vdi-vccas.home1.fr>
P-Preferred-Identity: <tel:+33123456789>
P-Preferred-Service: urn:urn-xxx:3gpp-service-ims.icsi.mmtel
Accept-Contact: *;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-
ims.icsi.mmtel"
Contact: <sip:[5555::1:2:3:4]:1357>
;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-ims.icsi.mmtel"
Call-ID: fea8x7-p
CSeq: 7283 INVITE
```

Note the following fields within this new INVITE request:

- the request URI and the To header are set to the VDI, that is pointing to Tobias's VCC AS;
- the P-Preferred-Identity header indicates the tel-URL of Tobias, which is the same phone number (MSISDN) as the one used on the original call (between Tobias's UE and the MGCF and further on to Theresa). The P-CSCF will assert this value and replace the header with a P-Asserted-Identity header;
- the Via and Contact headers indicate the IP address of Tobias's phone;
- the Route header points to Tobias's P-CSCF, over which the phone has just registered and to the address of Tobias's S-CSCF, which was provided during registration within the Service-Route header (see Section 12.3.3.1).

This initial INVITE request creates a new SIP dialog, which is independent of the already existing SIP dialogs that were created during anchoring of the original call. Tobias's phone has already one connection towards the VCC AS via the MGCF, this connection is called initial access leg (from Tobias). Now the phone creates a second connection towards the VCC AS, this time using IMS only. This second connection is called new access leg (from Tobias).

The INVITE request also includes an SDP Offer:

```
v=0
o=- 56777665533 47 IN IP6 5555::1:2:3:4
s=-
c=IN IP6 5555::1:2:3:4 t=907165798 0 m=audio 3111 RTP/AVP 98 99
a=rtpmap:98 AMR-WB a=rtpmap:99 telephone-event a=curr:qos local
sendrecv a=curr:qos remote none a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv a=tcap:1 RTP/AVP RTP/AVPF a=pcfg:1
t=1||2
```

In this SDP Offer, Tobias's phone indicates:

- its own IP address (5555::1:2:3:4) in the c-line;
- port 3111 in the m-line, this is the port over which the phone wants to send and receive the RTP media packets related to the audio session
- AMR-WB as the only codec that should be used for the call – Tobias's phone could indicate a list of codecs here, but it is aware that on the CS call (which is still ongoing in parallel) uses AMR-WB, so this codec must be supported on the route and therefore it is indicated here as the only choice (this is an implementation option and not required by the standard);
- the SDP capability negotiation framework identifiers for selecting AVPF if it is supported (tcap and pcfg attribute lines);
- that the resources necessary for this call are available at Tobias's phone (local des and curr attribute lines have matching preconditions), as we assume that the I-WLAN connection does not require the device to reserve resources.

Tobias's phone now sends out the INVITE request to the next hop on the route, i.e. the P-CSCF.

From this moment on, Tobias's phone must keep listen to both media connections, i.e. one for the ongoing CS call (towards the VMSC and the MGW) and another one for the IMS call (towards the GGSN). Theresa's phone can at any time decide to send media over the newly established IMS media connection, as Tobias's phone indicated, that all resources are met locally.

#### 13.4.2 Routing to Tobias's VCC AS

The INVITE request is routed via the normal SIP and IMS routing procedures to Tobias's VCC AS. It is sent first to the P-CSCF (based on the first entry in the Route header) and then to Tobias's S-CSCF (based on the next entry in the Route header). The S-CSCF now checks whether any of Tobias's Filter Criteria do match the incoming INVITE request. The first Filter Criteria is as follows:

```
Public User Identity: sip:tobias@home1.fr / tel:+33123456789 /
Initial Filter Criteria
Priority: 0
Service Trigger Point (STP):
Method: INVITE
Request-URI: vdi-vccas.home1.fr
Session Case: Originating
Application Server: vccas1.home1.fr
```

This means that every SIP INVITE request (Method STP) that is originated (Session Case STP) from Tobias (Public User identity) and which includes the VDI in the request URI (Request-URI STP) must be sent to Tobias's VCC AS (Application Server STP).

The S-CSCF therefore sends the INVITE request to Tobias's VCC AS, based on the procedures described in Section 12.3.8.

#### *13.4.3 Tobias's VCC AS Performs the CS to IMS Domain Transfer*

##### **13.4.3.1 Receiving INVITE request from Tobias, triggering domain transfer (new access leg)**

When receiving the INVITE request, Tobias's VCC AS detects by examining the VDI, that this INVITE request was sent in order to trigger a domain transfer from CS to IMS. It also sees from the P-Asserted-Identity header, that the INVITE request was initiated by Tobias's phone.

Currently the VCC AS has only one call from Tobias anchored, which is the call with Theresa that consists of the two SIP dialogs, one over the initial access leg and one over the remote leg.

In order to transfer the existing call, which originates in Tobias's CS domain, to Tobias's IMS domain, the VCC AS will only replace the initial access leg with the new access leg. By doing so, the remote access leg (and also all other connections from Tobias's VCC AS towards Theresa's phone) are kept, they only need to be updated with the new information of the new access leg from Tobias.

##### **13.4.3.2 Sending re-INVITE on the remote leg**

In order to update the existing connections towards Theresa with the new information about the new access leg, Tobias's VCC AS acts as a B2BUA and sends out a new INVITE request over the already existing dialog on the remote call leg:

```
INVITE sip:vccas2.home2.hu SIP/2.0
Via: SIP/2.0/UDP vccas1.home1.fr
Route: sip:scscf1.home1.fr;lr, sip:tas.home1.fr;lr, sip:scscf1.home1.fr,
sip:scscf2.home2.hu;lr, sip:tas.home2.hu;lr, sip:scscf2.home2.hu
From: <tel:+33123456789>;tag=asc3cvb
To: <tel:+36987654321>;tag=bvc22acs
Call-ID: tbjcpusbdd
CSeq: 3140 INVITE
```

Note the following fields in the INVITE request:

- the request URI is set to the address of Theresa's VCC AS that was received in the Contact header of the 200 (OK) response to the initial INVITE request (see Section 13.3.9.2);
- the Route header is set to the list of entities that record routed on the initial INVITE request over the remote call leg. Tobias's VCC AS received this list within the 200 (OK) response to the initial INVITE request;
- as this INVITE request is not initiating a new dialog but is sent on an already existing dialog, the To Tag, From Tag and Call-ID identify the dialog as the one that was established before between the the VCC Application Servers, when the original call was anchored (see e.g. Section 13.3.9.2);

- as the INVITE is sent on the existing dialog over the remote leg, the CSeq must take the next sequence number, i.e. 3140;
- the SDP Offer, that was conveyed in the recently received INVITE request from Tobias's phone over the new access leg (see Section 13.4.1) is copied over to this re-INVITE request in order to let Theresa's side know the new connection addresses and the resource reservation status on Tobias's side.

This re-INVITE traverses the IMS entities indicated in the Route header and finally reaches Theresa's VCC AS, which then acts as a B2BUA again (see e.g. Section 13.3.7.3) and sends the information contained in the request towards Theresa's phone within another re-INVITE request, that is sent over Theresa's initial access leg.

Theresa's phone sees from the SDP Offer in the received re-INVITE that the media connection address has changed. As the media characteristics have not changed, Theresa does not need to reserve further resources and therefore immediately can send out a 200 (OK) response to the INVITE request. Note that Theresa's phone does not need to send and provisional response here, e.g. 180 (Ringing), as the session is already established and the re-INVITE is only sent to change the connection parameters.

At this moment it is an implementation option which of the two media connections Theresa's phone decides to use. It can immediately take into use the new media connection towards Tobias's phone, which was indicated in the SDP Offer in the re-INVITE or it can wait until the SIP ACK request is received and go on sending media packets towards the old connection address (Tobias's MGW) that was provided in the initial SIP INVITE request.

The 200 (OK) response takes the same route as the 200 (OK) response to the initial INVITE request from Theresa's phone, that is described in Section 13.3.9.

### 13.4.3.3 Establishing the SIP dialog over the new access leg

Once Tobias's VCC AS receives the 200 (OK) response it sends it out on the new access leg towards Tobias:

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP scscf1.home1.fr,
      SIP/2.0/UDP pcscf1.home1.fr, SIP/2.0/UDP [5555::1:2:3:4]:1357
Record-Route: sip:scscf1.home1.fr;lr, sip:pcscf1.home1.fr;lr
From: <tel:+33123456789>;tag=osftpm
To: <sip:vdi-vccas.home1.fr>;tag=idvkoan
Contact: <vccas1.home1.fr>
;g.3gpp-icsi.ref="urn%3Aurn-xxx%3A3gpp-service-Call-ID: tbjcpusbdd
CSeq: 7283 INVITE
```

The 200 (OK) response also includes the SDP Answer, that was indicated by Theresa's phone when sending the response. Note that this SDP Answer was not changed by any of the entities on the route, even not by any of the VCC Application Servers that were acting as SIP B2BUAs:

```
v=0
```

```
o= - 1357924 1357924 IN IP6 5555::5:6:7:8 s=- c=IN IP6 5555::5:6:7:8
```

```
t=907165934 0 m=audio 4011 RTP/AVPF 97 98 a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event a=curr:qos local sendrecv a=curr:qos
remote sendrecv a=des:qos mandatory local sendrecv a=des:qos
mandatory remote sendrecv a=acfg:1 t=2
```

Tobias's phone now sends back a SIP ACK request, acknowledging the receipt of the 200 (OK) and thereby establishing the new media session and the SIP dialog. The sending/receiving of this SIP ACK request is the latest moment in time, when both phones should switch over the media handling from the old media connection (between Tobias's MGW and Theresa's phone) to the new media connection (between Tobias's and Theresa's phones, via their GGSNs).

#### 13.4.3.4 Releasing the SIP dialog over the initial access leg

After receiving the SIP ACK request over the new access leg from Tobias, Tobias's VCC AS releases the SIP dialog over the initial access leg towards Tobias's MGCF by sending a SIP BYE request. This BYE request is only sent on the initial access leg, the other SIP dialogs at Tobias's VCC AS (the one over the new access leg towards Tobias's phone and the one over the remote leg towards Theresa's VCC AS) stay active, as those are the dialogs which handle the end-to-end connection between the two phones now.

Tobias's MGCF interworks the BYE request into a ISUP REL message, indicating the cause value 16, which stands for normal call clearing. It also instructs the MGW to release all media resources for the ongoing call

Upon receipt of ISUP REL, the VMSC sends out a CS DISC message to Tobias's phone. With this the CS call and the SIP dialog over the initial access leg have been cleared. The communication from Tobias's phone towards Theresa is now handled on Tobias's side completely over the SIP dialog on the new access leg and the related media connection via Tobias's GGSN.

#### 13.4.4 Scenario after CS to IMS Domain Transfer

Figure 13.5 shows the existing connections between the different entities after Tobias's phone has performed the domain transfer from CS to IMS domain.

On the signalling level, the call is split into the following segments:

- the new SIP dialog over the new access leg from Tobias's phone to Tobias's VCC AS;
- the SIP dialog over the remote leg from Tobias's VCC AS to Theresa's VCC AS, which has not changed since anchoring was performed;
- the SIP dialog over the initial access leg from Theresa's VCC AS to Theresa's phone, which has not changed since anchoring was performed.

The media is now sent over the GGSNs of the two parties.

After the successful domain transfer, Tobias's VMSC, MGCF and MGW are not part of the connection anymore.

All the actions described in this Section 13.3.4 are taking place without any user interaction. The domain transfer takes place seamlessly for Tobias and Theresa and all the actions described here are performed automatically by the involved phones and the network elements.

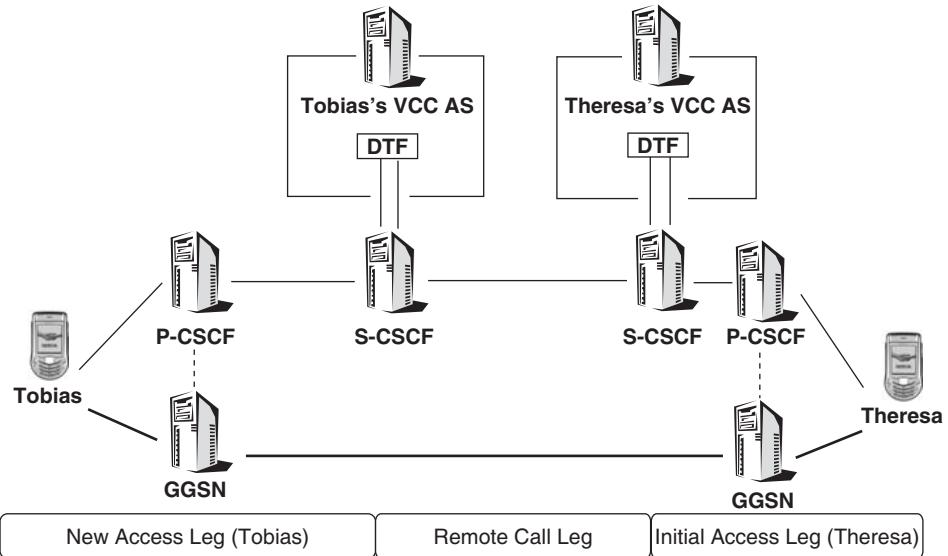


Figure 13.5 VCC –connections After CS to PS Domain Transfer (A-Side)

### 13.5 Theresa adds Video to the Call

After a while, Theresa decides that she wants to have also a video conversation with her brother. As the connection between the two phones is now end-to-end IMS, additional media types can be added by the users.

After Theresa selected to add video, the phone creates a re-INVITE request with a new SDP Offer, in order to inform the remote side (Tobias's phone) that a new media connection has to be established. In this example we assume that Theresa phone does not have the required resources available and therefore needs to reserve them.

Note that Theresa's phone could also have sent an UPDATE request on the existing SIP dialog in order to add a new media stream. This is nevertheless discouraged, as UPDATE does not allow resource reservation in parallel with the SIP signalling, as it immediately is answered with a 200 (OK). Only with the re-INVITE request is it possible to perform the update of the SIP dialog and resource reservation in parallel.

The SIP INVITE is sent by Theresa's phone over the initial access leg on Theresa's side and includes, besides others, the following information:

```

INVITE sip:vccas2.home2.hu SIP/2.0
Via: SIP/2.0/UDP [5555::5:6:7:8]:1006
Route: sip:pcscf2.home2.hu;lr, sip:scscf2.home2.hu;lr
From: <tel:+3612323454>;tag=ake1s9ls To:
<tel:+33123456789>;tag=thvas2rez Call-ID: blcpgrsws CSeq: 1212
INVITE

```

The request URI points to Theresa's VCC AS, which will, when receiving this INVITE request, send it further on the remote leg towards Tobias's phone.

The Route header indicates, that the request has to be sent to Theresa's P-CSCF and afterwards to her S-CSCF, from which it will be forwarded to the final destination, i.e. the VCC AS. Both CSCFs put their addresses into the Record-Route header of the initial INVITE request, which was sent over the initial access leg towards Theresa (see Section 13.3.8) and therefore need to stay on the route for every subsequent request.

The values in the To and From headers have been swapped, compared to the other SIP requests and responses we have seen so far. This is, as both headers are related to the SIP transaction and not to the SIP dialog. From the perspective of the SIP dialog, it is not relevant anymore, which of the two parties (Theresa or Tobias) sent and which received the initial INVITE request – they are both now in an active call. The From header indicates Theresa's tel URL, which means that the re-INVITE was sent by Theresa and the To header indicates that Tobias is the destination of the request.

Note that the values of the To and From fields as well as the tags are only swapped around, but they are not allowed to be changed during the dialog. This is due to the fact, that Call-ID, To and From tags in combination identify the SIP dialog.

The Call-ID value stays the same during the lifetime of the dialog, but here we now see a CSeq value (1212) that is lower than the one which was received with the latest request (i.e. the re-INVITE from Tobias's side, which indicated the CS to IMS domain transfer, the CSeq value was 6413 – see Section 13.4). This is due to the fact that in SIP the CSeq numbers are assigned only by the User Agent Client, which is sending the requests, i.e. they are independent on each side of the communication. This means that the ranges of CSeq numbers used within the SIP requests sent by both sides are independent of the remote side, i.e. Theresa's phone is free to generate its own initial CSeq number with the first SIP request it sends over the dialog, independent of the last CSeq number it received from the remote side.

Also note, that all SIP responses sent for this request will include the same CSeq, To and From header values as this re-INVITE, as SIP responses always take the same values of To, From, CSeq and Call-ID headers as the related SIP requests.

The INVITE request includes the following SDP:

```
v=0
o= - 1357924 1357924 IN IP6 5555::5:6:7:8
s=-_
c=IN IP6 5555::5:6:7:8
t=907165300 0
m=audio 4011 RTP/AVPF 97 98
a=rtpmap:97 AMR-WB
a=rtpmap:98 telephone-event
a=curr:qos local sendrecv
a=curr:qos remote sendrecv
a=des:qos mandatory local sendrecv
a=des:qos mandatory remote sendrecv
m=video 4100 RTP/AVPF 31 34
a=curr:qos local none
a=curr:qos remote none
a=des:qos mandatory local sendrecv
a=des:qos none remote sendrecv$'$
a=inactive
```

The new SDP Offer repeats the parameters for the already ongoing audio session, for which resources have already been reserved. A new m-line is added, indicating video. For this video stream the phone immediately indicates the usage of AVPF, as it knows from the initial INVITE exchange, that the remote side supports AVPF. Theresa's phone offers two codecs H.261 (RTP payload type 31) and H.263 (RTP payload type 34), this means the final codec selection has not been done yet and therefore Theresa's phone has to wait with the resource reservation until it receives the final codec within an SDP Answer from the remote side.

The preconditions in this SDP Offer indicate that the local resources have not been reserved yet, that resources need to be reserved at Theresa's side for sending and receiving and that Theresa's phone has now knowledge about the availability of resources on the remote side.

In this example we will not go further into details of the SIP and SDP signalling, as this has been outlined in Chapter 12.

After sending the re-INVITE request, the following actions take place:

- Theresa's and Tobias's VCC Application Servers will act as B2BUAs on the request and forward it towards Tobias. The two VCC Application Servers will not interfere with any of the SIP messages, as these messages are not related to VCC, i.e. they are just sent back and forth between the two phones;
- Tobias's phone will take the final codec selection (in this case it will select RTP payload type 34, which is H.263), start resource reservation and send back a 183 (Session Progress) response with a related SDP Answer, including besides other information the a = conf line (see Section 12.6.4.2);
- Theresa's phone will acknowledge the 183 (Session Progress) response with a SIP PRACK request and start with resource reservation;
- once Theresa's phone has finished resource reservation it will send out a SIP UPDATE request, indicating in an SDP Offer that the local preconditions are met;
- Tobias's phone, upon receiving the SIP UPDATE request, has its resources available as well and answers the UPDATE request with a 200 (OK) response, including a SDP Answer which shows now all preconditions as met;
- the video stream will immediately be added after this. There is no need for ringing the participants, as they are already in an audio session, so Tobias's phone will send immediately a 200 (OK) response for the INVITE request as well, for which Theresa's phone will send an ACK request.

Besides Theresa's request to add video to the call, all the actions described in this section were performed automatically by the phones and the network entities.

After this Theresa and Tobias can communicate via audio and video.

## 13.6 Domain Transfer: IMS to CS

### 13.6.1 Theresa's Phone Starts VCC Procedures

We assume now that Theresa is walking out of GPRS coverage and therefore will lose the connection to IMS soon. In order to save the ongoing call it starts to invoke the

procedures for VCC domain transfer from IMS to CS domain and sends out, in parallel to the still ongoing IMS call over Theresa's initial access leg, a CS SETUP message with the following content:

- the Calling Party Number Information Element (IE) set to Theresa's phone number, as it was dialed by Tobias, +36987654321;
- the Called Party Number IE set to the VCC Domain Transfer Number (VDN) that was assigned by the Communication Continuity MO (see 13.2), +361139384756;
- the Bearer Capability IE indicating that PCMU is supported as a audio codec for this call.

In this example, AMR-WB is not any longer used as a codec, which is mainly done to give a better example for codec negotiation during a VCC domain transfer.

### *13.6.2 From Theresa's Phone to Theresa's VCC AS*

The CS SETUP message is sent to Theresa's VMSC and from there on the routing and processing of the call is similar to that described in Section 13.3, i.e.:

- the VMSC contacts the gsmSCF, which is part of Theresa's VCC AS, in order to find out further routing information;
- the gsmSCF, together with the CSAF and the CAMEL service function, assigns an IMRN (+3663478569) to the call, which is sent back to the VMSC;
- the VMSC routes the call forward by sending an ISUP IAM message, indicating the IMRN as the called party number, to Theresa's MGCF;
- the MGCF interworks the ISUP IAM to a SIP INVITE request, including an SDP Offer. The SDP Offer indicates PCMU as the only audio codec for the call and also shows that the resources are already available. The MGCF routes the SIP INVITE request to the I-CSCF in Theresa's network;
- the I-CSCF queries the SLF/HSS in order to resolve the IMRN in the request URI of the SIP INVITE request, as the IMRN is a PSI the request is directly forwarded from the I-CSCF to Theresa's VCC AS.

### *13.6.3 Performing the IMS to CS Domain Transfer*

Theresa's VCC AS detects that the incoming SIP INVITE request establishes a new access leg towards Theresa and now starts performing the VCC domain transfer procedures, which are similar to those described in Section 13.4, i.e.:

- Theresa's VCC AS sends a SIP re-INVITE request over the remote call leg, indicating the SDP that was received from the MGCF, i.e. PCMU is still the codec to be used for the call;
- upon receiving the re-INVITE request over the remote call leg Tobias's VCC AS acts as a SIP B2BUA and sends a re-INVITE request over the new access leg towards Tobias's phone;

The SDP Offer that is received at Tobias's phone in the re-INVITE request looks as follows:

```
v=0
o= - 1357924 1357924 IN IP6 5555::71:82:93:64 s=- c=IN IP6
5555::71:82:93:64 t=907165900 0 m=audio 4098 RTP/AVPF 0 98
a=rtpmap:98 telephone-event a=curr:qos local sendrecv a=curr:qos
remote none a=des:qos mandatory local sendrecv a=des:qos none remote
sendrecv m=video 0 RTP/AVPF 31 0
```

The connection address (c-line) is set to Theresa's MGCF (5555::71:82:93:64), which will handle all the media after the domain transfer has been done. In the first media line, the audio codec has been changed to "0" which is the rtp payload type for PCMU. The preconditions for the audio line indicate that resources are available at Theresa's side.

The video stream, which was added by Theresa before (see Section 13.5) has now been dropped by setting the port number in the m-line for video to "0", which means that the media stream is rejected by Theresa's side. As VCC only works for voice calls, video calls (or any other type of media besides audio) cannot be transferred from IMS to CS.

Tobias's phone sends back a 200 (OK) response to the re-INVITE request once it has finished resource reservation for PCMU. The 200 (OK) also conveys an SDP answer, indicating that resources for PCMU are available now on both sides. From this moment on, Tobias's phone must be prepared to switch over the media from the old (AMR-WB) to the new (PCMU) codec. The PCMU codec is latest taken into use once the SIP ACK request is received by Tobias's phone.

The VCC domain transfer continuous similar to the procedures described in Sections 13.3 and 13.4, i.e.:

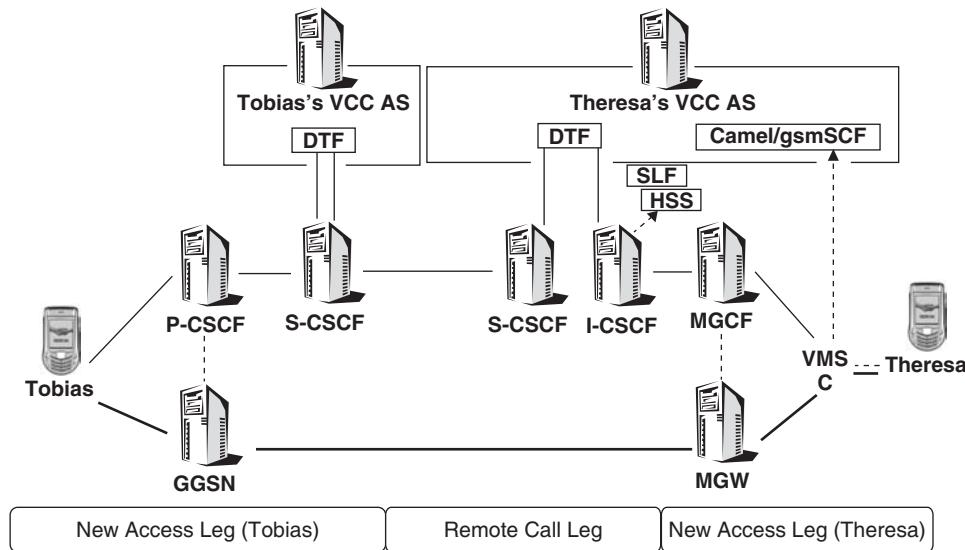
- the 200 (OK) request gets delivered to Theresa's MGCF, which acknowledges it with a SIP ACK response and interworks the 200 (OK) response into a ISUP ANM (answer message) which is sent to the VMSC;
- the VMSC sends a CS Connect message to Theresa's phone, which then takes the CS call into use and responds with a CS Connect Acknowledge message. Now the connection is active via the new access leg towards Theresa;
- upon receiving the SIP ACK request from the MGCF, Theresa's VCC AS releases Theresa's initial access leg by sending a SIP BYE request on the related SIP dialog. Based on this, Theresa's S-CSCF, P-CSCF and GGSN drop out of the end-to-end connection.

#### 13.6.4 Scenario after IMS to CS Domain Transfer

Figure 13.6 shows the connections between the different entities after the VCC domain transfer from IMS to CS has been performed.

On the signalling level, the call is split into different segments:

- the SIP dialog over the new access leg from Tobias's phone to Tobias's VCC AS;
- the SIP dialog over the remote access leg between the two VCC Application Servers;
- the SIP dialog over the new access leg from Theresa's VCC AS to Theresa's MGCF;



**Figure 13.6** VCC –connections After PS to CS Domain Transfer (B-Side)

- the SS7/ISUP connection between Theresa's MGCF and Theresa's VMSC; and
- the CS connection between Theresa's VMSC and her phone.

On a media level the connection is now handled via Tobias's GGSN and Theresa's MGW, which is controlled by Theresa's MGCF.

All the actions described in this Section 13.6 are taking place without any user interaction. The domain transfer takes place seamlessly for Tobias and Theresa and all the actions described here are performed automatically by the involved phones and the network elements.

## 13.7 Related Standards

Voice Call Continuity and basic IMS:

- 3GPP TS 23.206 Voice Call Continuity (VCC) between Circuit Switched (CS) and IP Multimedia Subsystem (IMS); Stage 2
- 3GPP TS 24.206 Voice call continuity between Circuit Switched (CS) and IP Multimedia Subsystem (IMS); Stage 3
- 3GPP TS 24.216 Communication Continuity Management Object (MO)
- 3GPP TS 24.229 Internet Protocol (IP) multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3

CS Protocols, Interworking and Media Control

3GPP TS 24.008	Mobile radio interface Layer 3 specification; Core network protocols; Stage 3
3GPP TS 29.163	Interworking between the IP Multimedia (IM) Core Network (CN) subsystem and Circuit Switched (CS) networks
3GPP TS 29.232	Media Gateway Controller (MGC) – Media Gateway (MGW) interface
ITU-T Recommendation H.248.1	Gateway control protocol: Version 2
ITU-T Recommendation H.245	Control protocol for multimedia communication

HSS and SLF communication (Cx, Dx, Sh, Dh interfaces):

3GPP TS 29.228	IP Multimedia (IM) Subsystem Cx and Dx interfaces; Signalling flows and message contents
3GPP TS 29.229	Cx and Dx interfaces based on the Diameter protocol; Protocol details
3GPP TS 29.328	IP Multimedia (IM) Subsystem Sh interface; Signalling flows and message contents
3GPP TS 29.329 RFC 3588	Sh Interface based on the Diameter protocol; Protocol details Diameter Base Protocol



# References

## Third Generation Partnership Project (3GPP)

- [3GPP TR 23.815] Charging implications of IMS architecture.
- [3GPP TS 24.930] Signalling flows for the session setup in the IMS based on SIP and SDP.
- [3GPP TS 22.101] Service principles.
- [3GPP TS 22.228] Service requirements for the IP multimedia core network subsystem.
- [3GPP TS 23.002] Network architecture.
- [3GPP TS 23.003] Technical Specification Group Core Network; Numbering, addressing and identification.
- [3GPP TS 23.060] General Packet Radio Service (GPRS); Service description; Stage 2.
- [3GPP TS 23.206] Voice Call Continuity (VCC) between Circuit Switched (CS) and IP Multimedia Subsystem (IMS); Stage 2.
- [3GPP TS 23.107] Quality of Service (QoS) concept and architecture.
- [3GPP TS 23.207] End-to-End QoS Concept and Architecture.
- [3GPP TS 23.218] IP Multimedia (IM) session handling; IM call model; Stage 2.
- [3GPP TS 23.221] Architectural requirements.
- [3GPP TS 23.228] IP Multimedia (IM) Subsystem; Stage 2.
- [3GPP TS 23.333] Multimedia Resource Function Controller (MRFC) – Multimedia Resource Function Processor (MRFP) Mp interface; Procedures descriptions.
- [3GPP TS 23.981] Interworking aspects and migration scenarios for IPv4 based IMS Implementations.
- [3GPP TS 24.008] Mobile Radio Interface Layer 3 specification; Core Network Protocols; Stage 3.
- [3GPP TS 24.147] Conferencing using the IP Multimedia (IM) Core Network (CN) subsystem; Stage 3.
- [3GPP TS 24.173] IMS Multimedia telephony service and supplementary services; Stage 3.
- [3GPP TS 24.216] Communication Continuity Management Object (MO).
- [3GPP TS 24.229] IP Multimedia Call Control based on SIP and SDP; Stage 3.
- [3GPP TS 24.341] Support of SMS over IP networks; Stage 3.
- [3GPP TS 26.114] IMS Multimedia telephony; Media handling and interaction.
- [3GPP TS 26.234] Transparent end-to-end packet-switched streaming service (PSS); Protocols and codecs.
- [3GPP TS 26.236] Packet-switched conversational multimedia applications; Transport protocols.
- [3GPP TS 29.163] Interworking between the IP Multimedia (IM) Core Network (CN) subsystem and Circuit Switched (CS) networks.
- [3GPP TS 29.198] Open Service Access (OSA); Application Programming Interface (API), Multiple parts.
- [3GPP TS 24.206] Voice call continuity between Circuit Switched (CS) and IP Multimedia Subsystem (IMS); Stage 3.
- [3GPP TS 29.212] Policy and charging control over Gx reference point
- [3GPP TS 29.213] Policy and charging control signalling flows and Quality of Service (QoS) parameter mapping.

- [3GPP TS 29.214] Policy and charging control over Rx reference point.
- [3GPP TS 29.228] IP Multimedia (IM) Subsystem Cx and Dx Interfaces; Signalling flows and message contents.
- [3GPP TS 29.229] Cx and Dx interfaces based on the Diameter protocol; Protocol details.
- [3GPP TS 29.232] Media Gateway Controller (MGC) – Media Gateway (MGW) interface
- [3GPP TS 29.328] IP Multimedia (IM) Subsystem Sh interface; Signalling flows and message contents.
- [3GPP TS 29.329] Sh interface based on the Diameter protocol.
- [3GPP TS 29.333] Multimedia Resource Function Controller (MRFC) – Multimedia Resource Function Processor (MRFP) Mp interface; Stage 3.
- [3GPP TS 32.240] Charging architecture and principles.
- [3GPP TS 32.295] Charging management; Charging Data Record (CDR) transfer.
- [3GPP TS 32.296] Online Charging System (OCS): Applications and interfaces.
- [3GPP TS 32.299] Diameter charging applications
- [3GPP TS 33.102] 3G security; Security architecture.
- [3GPP TS 33.203] 3G security; Access security for IP-based services.
- [3GPP TS 33.210] 3G security; Network Domain Security (NDS); IP network layer security, June 2003.
- [3GPP TS 33.220] 3G security; Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture, 3GPP.
- [3GPP TS 33.222] Generic Authentication Architecture (GAA); Access to Network Application Functions using Secure Hypertext Transfer Protocol (HTTPS).
- [3GPP TS 33.310] Network Domain Security; Authentication Framework (NDS/AF).

## Internet Engineering Task Force (IETF)

- [draft-ietf-behave-rfc3489bis] Rosenberg, J., Mahy, R., Matthews, P. and Wing, D., Session Traversal Utilities for (NAT) (STUN), February 2008.
- [draft-ietf-behave-turn] Rosenberg, J., Mahy, R. and Matthews, P., Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN), February 2008.
- [draft-ietf-mmusic-ice] Rosenberg, J., Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols, October 2007.
- [draft-ietf-simple-imdn-06] Burger, E. and Khatabil, H., Instant Message Disposition Notification, April 2008.
- [draft-ietf-sip-uri-list-conferencing] Camarillo, C. and Johnston, A., Conference Establishment Using Request-Contained Lists in the Session Initiation Protocol (SIP), November 2007.
- [draft-drage-sipping-service-identification] A Session Initiation Protocol (SIP) Extension for the Identification of Services.
- [draft-ietf-mmusic-sdp-capability-negotiation] SDP Capability Negotiation.
- [draft-ietf-sip-gruu-15] Obtaining and Using Globally Routable User Agent (UA) URIs (GRUU) in the Session Initiation Protocol (SIP).
- [draft-ietf-sipping-gruu-reg-event] Registration Event Package Extension for Session Initiation Protocol (SIP) Globally Routable User Agent URIs (GRUUs).
- [draft-monrad-sipping-3gpp-urn-namespace] A Uniform Resource Name (URN) Namespace for the 3rd Generation Partnership Project (3GPP).
- [RFC1305] Network Time Protocol (Version 3) Specification, Implementation and Analysis.
- [RFC1321] Rivest, R., The MD5 Message-Digest Algorithm, April 1992.
- [RFC1851] Karp, P., Metzger, P. and Simpson, W. The ESP Triple DES Transform, September 1995.

- [RFC2246] The TLS Protocol Version 1.0.
- [RFC2396] Berners-Lee, T., Fielding, R., Irvine, U.C. and Masinter, L., Uniform Resource Identifiers (URI): Generic Syntax, August 1998.
- [RFC2401] Security Architecture for the Internet Protocol.
- [RFC2403] The Use of HMAC-MD5-96 within ESP and AH.
- [RFC2404] Madson, C. and Glenn, R. The Use of HMAC-SHA-1-96 within ESP and AH, November 1998.
- [RFC2406] Kent, S. and Atkinson, R. IP Encapsulating Security Payload (ESP), November 1998.
- [RFC2408] Maughan, D., Schneider, M. and Schertler, M. Internet Security Association and Key Management Protocol (ISAKMP), November 1998.
- [RFC2409] Harkins, D. and Carrel, D. The Internet Key Exchange (IKE), November 1998.
- [RFC2451] The ESP CBC-Mode Cipher Algorithms.
- [RFC2486] Aboba, B. and Beadles, M. The Network Access Identifier, January 1999.
- [RFC2616] Fielding, R., Gettys, J., Mogul, J., Nielsen, H., Masinter, L., Leach, P. and Berners-Lee, T. Hypertext Transfer Protocol – HTTP/1.1, June 1999.
- [RFC2617] Franks, J., Hallam-Baker, P., Hostetler, J., Lawrence, S., Leach, P., Luotonen, A. and Stewart, L. HTTP Authentication: Basic and Digest Access Authentication, June 1999.
- [RFC2782] A DNS RR for specifying the location of services (DNS SRV).
- [RFC2810] Kalt, C., Internet Relay Chat: Architecture, April 2000.
- [RFC2833] RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals.
- [RFC2915] The Naming Authority Pointer (NAPTR) DNS Resource Record.
- [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and Schooler, E. SIP: Session Initiation Protocol, June 2002.
- [RFC3262] Reliability of Provisional Responses in the Session Initiation Protocol (SIP).
- [RFC3263] Session Initiation Protocol (SIP): Locating SIP Servers.
- [RFC3264] An Offer/Answer Model with SDP.
- [RFC3265] Session Initiation Protocol (SIP)-specific Event Notification.
- [RFC3310] Niemi, A., Arkko, J. and Torvinen, V. Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA), September 2002.
- [RFC3311] The Session Initiation Protocol (SIP) UPDATE Method.
- [RFC3312] Integration of Resource Management and Session Initiation Protocol.
- [RFC3315] Droms, R., Bounds, J., Volz, B., Lemon, T., Perkins, C. and Carney, M. Dynamic Host Configuration Protocol for IPv6 (DHCPv6), July 2003.
- [RFC3319] Schulzrinne, H. and Volt, B. Dynamic Host Configuration Protocol (DHCPv6) Options for Session Initiation Protocol (SIP) Servers, July 2003.
- [RFC3320] Price, R., Bormann, C., Christoffersson, J., Hannu, H., Liu, Z. and Rosenberg, J. Signalling Compression (SigComp), January 2003.
- [RFC3323] Peterson, J., A Privacy Mechanism for the Session Initiation Protocol (SIP), November 2002.
- [RFC3325] Jenning, C., Peterson, J. and Watson, M. Private Extensions to the Session Initiation Protocol (SIP) for Asserted Identity within Trusted Networks, November 2002.
- [RFC3327] Session Initiation Protocol (SIP) Extension Header Field for Registering Non-Adjacent Contacts.
- [RFC3329] Arkko, J., Torvinen, V., Camarillo, G., Niemi, A. and Haukka, T. Security Mechanism Agreement for the Session Initiation Protocol (SIP), January 2003.
- [RFC3428] Campbell, B. and Rosenberg, J. Session Initiation Protocol Extension for Instant Messaging, September 2002.
- [RFC3455] Private Header (P-Header) Extensions to the Session Initiation Protocol (SIP) for the 3rd-Generation Partnership Project (3GPP).

- [RFC3486] Compressing the Session Initiation Protocol (SIP).
- [RFC3515] The Session Initiation Protocol (SIP) Refer Method.
- [RFC3550] RTP: A Transport Protocol for Real-time Applications.
- [RFC3551] RTP Profile for Audio and Video Conferences with Minimal Control.
- [RFC3556] SDP Bandwidth Modifiers for RTCP Bandwidth.
- [RFC3588] Calhoun, P., Loughney, J., Guttman, E., Zorn, G. and Arkko, J. Diameter Base Protocol, September 2003.
- [RFC3589] Diameter Command Codes for Third Generation Partnership Project (3GPP) Release 5.
- [RFC3608] Session Initiation Protocol (SIP) Extension Header Field for Service Route Discovery during Registration.
- [RFC3646] DNS Configuration options for Dynamic Host Configuration Protocol for IPv6 (DHCPv6).
- [RFC3680] A Session Initiation Protocol (SIP) Event Package for Registrations.
- [RFC3840] Indicating User Agent Capabilities in the Session Initiation Protocol (SIP).
- [RFC3841] Caller Preferences for the Session Initiation Protocol (SIP).
- [RFC3857] Rosenberg, J., A Watcher Information Event Template-package for the Session Initiation Protocol (SIP), August 2004.
- [RFC3858] Rosenberg, J., An Extensible Markup Language (XML) Based Format for Watcher Information, August 2004.
- [RFC3966] Schulzrinne, H. and Vaha-Sipila A. 'The tel URI for Telephone Numbers', December 2004.
- [RFC4032] Update to the Session Initiation Protocol (SIP) Preconditions Framework.
- [RFC4214] Templin, F., Gleeson, T., Talwar, M. and Thaler, D., Intra-Site Automatic Tunnel Addressing Protocol (ISATAP), October 2005.
- [RFC4566] SDP: Session Description Protocol.
- [RFC4575] Rosenberg, J., Schulzrinne, H. and Levin, O. A Session Initiation Protocol (SIP) Event Package for Conference State, August 2006.
- [RFC4582] The Binary Floor Control Protocol (BFCP).
- [RFC4585] Extended RTP Profile for Real-time Transport Control Protocol (RTCP)-Based Feedback (RTP/AVPF).
- [RFC4596] Guidelines for Usage of the Session Initiation Protocol (SIP) Caller Preferences Extension.
- [RFC4662] Roach, A.B., Campbell, B. and Rosenberg, J., Session Initiation Protocol (SIP) Event Notification Extension for Resource Lists, August 2006.
- [RFC4582] Camarillo, G., Ott, J. and Drage, K., The Binary Floor Control Protocol (BFCP), November 2006.
- [RFC4745] Schulzrinne, H., Tschofenig, H., Morris, J., Cuellar, J., Polk, J. and Rosenberg, J., Common Policy: A Document Format for Expressing Privacy Preferences, February 2007.
- [RFC4825] Rosenberg, J., The Extensible Markup Language (XML) Configuration Access Protocol (XCAP), May 2007.
- [RFC4826] Rosenberg, J., Extensible Markup Language (XML) Formats for Representing Resource Lists, May 2007.
- [RFC4896] Surtees, A., West, M. and Roach, A.B, Signalling Compression (SigComp) Corrections and Clarifications, June 2007.
- [RFC4975] Campbell, B., Mahy, R. and Jennings, C., The Message Session Relay Protocol (MSRP), September 2007.
- [RFC4976] Jennings, C., Mahy, R. and Roach, A. B., Relay Extensions for the Message Session Relay Protocol (MSRP), September 2007.
- [RFC5009] Private Header (P-Header) Extension to the Session Initiation Protocol (SIP) for Authorization of Early Media.

## Open Mobile Alliance (OMA)

[OMA IM AD]	Open Mobile Alliance, Instant Messaging using SIMPLE
[OMA PoC AD]	Open Mobile Alliance, Push to Talk Over Cellular (PoC) – Architecture
[OMA POC Control Plane]	Open Mobile Alliance, OMA PoC Control Plane
[OMA PoC RD]	Open Mobile Alliance, Push to Talk Over Cellular Requirements
[OMA Presence Architecture]	Open Mobile Alliance, Presence Architecture
[OMA-TS-IM_XDM]	Open Mobile Alliance, IM XDM Specification
[OMA-TS-PoC_XDM]	Open Mobile Alliance, PoC XDM Specification

## International Telecommunication Union (ITU-T)

[ITU-T Recommendation H.245]	Control protocol for multimedia communication
[ITU-T Recommendation H.248.1]	Gateway control protocol: Version 2



# List of Abbreviations

3GPP	Third Generation Partnership Project
3GPP2	Third Generation Partnership Project 2
A RR	IPv4 Address Resource Record
AAA	Authentication, Authorization and Accounting
AAAA RR	IPv6 Address Resource Record
AAL	ATM Adaptation Layer
AAR	AA-Request
ACA	Accounting Answer
ACK	Acknowledgement
ACL	Access Control List
ACR	Accounting Request
ADSL	Asynchronous Digital Subscriber Line
AH	Authentication Header
AKA	Authentication and Key Agreement
ALG	Application Level Gateway
AMR	Adaptive Multi-Rate
AMR-WB	Adaptive Multi-Rate Wideband
ANM	ISUP Answer Message
AOR	Address Of Record
API	Application Programming Interface
APN	Access Point Name
ARIB	Association of Radio Industries and Businesses (Japan)
AS	Application Server
ASA	Abort-Session-Answer
ASR	Abort-Session-Request
ATA	Analog Terminal Adapter
ATM	Asynchronous Transfer Mode
AUC	AUthentication Centre
AUID	Application Usage ID
AUTN	Authentication token
AUTS	Synchronization token
AV	Authentication Vector

AVP	Attribute Value Pair
AVP	Audio Video Profile
AVPF	RTP Audio-Visual Profile with Feedback
B2BUA	Back to Back UA
BCF	Bearer Charging Function
BER	Bit Error Ratio
BFCP	Binary Floor Control Protocol
BGCF	Breakout Gateway Control Function
BICC	Bearer Independent Call Control
BNF	Backus-Naur Form grammar
BS	Bearer Service; Billing System
BSF	Bootstrapping Server Function
BTS	Base Transceiver Station
CA	Certificate Authority
CAMEL	Customized Applications for Mobile network Enhanced Logic
CAP	Camel Application Part
CB	Communication Barring
CCF	Charging Collection Function
CCSA	China Communications Standards Association
CD	Communication Deflection
CDF	Charging Data Function
CDMA	Code Division Multiple Access
CDR	Charging Data Record
CFB	Communication Forwarding Busy
CFNL	Communication Forwarding on Not Logged-in
CFNR	Communication Forwarding No Reply
CFNRC	Communication diversion on mobile subscriber not reachable
CFU	Communication Forwarding Unconditional
CGF	Charging Gateway Function
CGI	Cell Global ID
CK	Ciphering Key
CLIP	Calling Line Identification Presentation
CN	Core Network
CONF	Conference
CPCP	Conference Policy Control Protocol
CPIM	Common Presence and Instant Messaging
CRLF	Carriage Return Line Feed
CS	Circuit Switched
CSCN	Circuit Switched Core Network
CSCF	Call Session Control Function
CSE	CAMEL Service Environment
CSRN	CS Domain Routeing Number
CTF	Charging Trigger Function
DDDS	Dynamic Delegation Discovery System
DES	Data Encryption Standard
DHCP	Dynamic Host Configuration Protocol

DHCPv6	Dynamic Host Configuration Protocol for IPv6
DL	Downlink
DM	Device Management
DNS	Domain Name System
DNS SRV	Domain Name System Service Resource Record
DOCSIS	Data Over Cable Service Interface Specification
DOI	Domain Of Interpretation
DOS	Denial Of Service
DSF	Domain Selection Function
DSL	Digital Subscriber Line
DTF	Domain Transfer Function
DTMF	Dual-Tone MultiFrequency
EAP	Extensible Authentication Protocol
ECF	Event Charging Function
E-CSCF	Emergency-CSCF
ECT	Explicit Communication Transfer
EDGE	Enhanced Data Rates for Global Evolution
ENUM	Telephone E.164 Number Mapping
ESP	Encapsulation Security Payload
ETSI	European Telecommunications Standards Institute
FQDN	Fully Qualified Domain Name
FRS	Family Radio Service
FSM	Finite State Machine
GAA	Generic Authentication Architecture
GBA	Generic Bootstrapping Architecture
GBR	Guaranteed Bit Rate
G-CDR	GGSN-CDR
GCID	GPRS Charging IDentifier
GGSN	Gateway GPRS Support Node
GIBA	GPRS-IMS-Bundled Authentication
GPRS	General Packet Radio Service
GRUU	Globally Routable User Agent URI
GSM	Global System for Mobile Communications
HLR	Home Location Register
HOLD	Communication Hold
HSS	Home Subscriber Server
HTTP	Hyper Text Transfer Protocol
IAB	Internet Architecture Board
IAM	Initial Address Message
IANA	Internet Assigned Numbers Authority
IARI	IMS Application Reference Identification
IBCF	Interconnection Border Control Function
ICB	Incoming Communications Barring
ICE	Interactive Connectivity Establishment
ICID	IMS Charging IDentifier
I-CSCF	Interrogating-CSCF

ICSI	IMS Communication Service Identification
IE	Information Element
IESG	Internet Engineering Steering Group
IETF	Internet Engineering Task Force
IK	Integrity Key
IKE	Internet Key Exchange
IM	Instant Messaging
IMPU	IMS Public User Identity
IMRN	IP Multimedia Routeing Number
IMS	IP Multimedia Subsystem
IMS MO	IMS Management Object
IMS-GWF	IMS-Gateway Function
IMSI	International Mobile Subscriber Identifier
IMS-MGW	IP Multimedia Subsystem-Media Gateway Function
IM-SSF	IP Multimedia Service Switching Function
IOI	InterOperator Identifier
IP	Internet Protocol
IP-CAN	IP-Connectivity Access Network
IPsec	Internet Protocol security
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
IRC	Internet Relay Chat
ISAKMP	Internet Security Association and Key Management Protocol
ISB	Incoming Session Barring
ISC	IMS Service Control
ISDN	Integrated Services Digital Network
ISIM	IP Multimedia Services Identity Module
ISP	Internet Service Provider
ISUP	ISDN User Part
IV	Initialization Vector
L1	Layer 1
LCS	Location services
LIA	Location-Info-Answer
LIR	Location-Info-Request
LMR	Land Mobile Radio
LRF	Location Retrieval Function
LTE	Long Term Evolution
M3UA	SS7 MTP3-User Adaptation layer
MAA	Multimedia-Multimedia-Answer
MAC	Message Authentication Checksum
MAP	Mobile Application Part
MAR	Multimedia-Auth-Request
Mbone	Multicast backbone
MBR	Maximum Bit Rate
MCC	Mobile Country Code
MEGACO	Media Gateway Control Protocol

MGCF	Media Gateway Control Function
MGW	Media Gateway Function
MID	Media stream IDentification
MIME	Multipurpose Internet Mail Extension
MITM	Man In The Middle
MMD	Multimedia Domain
MMS	Multimedia Messaging Service
MMSC	Multimedia Messaging Service Centre
MMtel	Multimedia Telephony Communication Service
MNC	Mobile Network Code
MO	Management Object
MOBILE IP	Mobile Internet Protocol
MPTY	Multiparty conference
MPV	Music Photo Video
MRFC	Multimedia Resource Function Controller
MRFP	Media Resource Function Processor
MS	Mobile Station
MSC	Mobile Switching Centre
MSIN	Mobile Subscriber Identification Number
MSISDN	Mobile Subscriber International ISDN Number
MSRP	Message Session Relay Protocol
MTP	Message Transfer Part
MTPn	Message Transfer Part level n
MTU	Maximum Transfer Unit
MWI	Message Waiting Indication
NAF	Network Application Function
NAI	Network Access Identifier
NAPTR	Naming Authority PoinTeR
NAS	Network Access Server
NASREQ	Network Access Server REQuirements
NASS	Network Access Subsystem
NAT	Network Address Translator
NAT-PT	Network Address Translator–Protocol Translator
NDS	Network Domain Security
NGN	Next Generation Network
NNI	Network-to-Network Interface
NTP	Network Time Protocol
OCB	Outgoing Communication Barring
OCS	Online Charging System
OIP	Originating Identification Presentation
OIR	Originating Identification Restriction
OMA	Open Mobile Alliance
OMA DM	Open Mobile Alliance Device Management
OSA	Open Services Architecture
OTA	Over-the-Air
P2P	Peer to peer

PA	Presence Agent
PBX	Private Branch Exchange
PCC	Policy and Charging Control
PCEF	Policy and Charging Enforcement Point
PCMU	Pulse Code Modulation m-law
PCRF	Policy and Charging Rules Function
P-CSCF	Proxy-CSCF
PDF	Policy Decision Function
PDP	Packet Data Protocol; Policy Decision Point
PEF	Policy Enforcement Function
PEP	Policy Enforcement Point
PKI	Public Key Infrastructure
PLMN	Public Land Mobile Network
PMR	Professional Mobile Radio
PNA	Push-Notification-Answer
PNR	Push-Notification-Request
PoC	Push to talk over the Cellular service
POTS	Plain Old Telephone Service
PPA	Push-Profile-Answer
PPA	Push-Profile Answer
PPR	Push-Profile-Request
PR	Provisional Response
PRACK	Provisional Response ACKnowledgement
PRC	PROvisioning Class
PRI	PROvisioning Instance
PRID	PROvisioning instance IDentifier
PS	Packet Switched; Presence Server
PSAP	Public Safety Answering Point
PSI	Public Service Identity
PSKs	Pre-shared Secret Keys
PSTN	Public Switched Telephone Network
PUA	Presence User Agent; Profile-Update-Answer
PUR	Profile-Update-Request
QoS	Quality of Service
RAA	Re-Auth Request
RADIUS	Remote Authentication Dial In User Service
RAN	Radio Access Network
RAND	Random challenge
RAR	Re-Auth-Request
RDF	Routing Determination Function
RES	Response
RFC	Requests For Comments
RLS	Resource List Server
RNC	Radio Network Controller
ROAMOPS	Roaming operations
RR	Receiver Reports

RR	RTCP receiver bandwidth modifier
RS	RTCP sender bandwidth modifier
RSVP	Resource ReserVation setup Protocol
RTA	Registration-Termination-Answer
RTCP	RTP Control Protocol
RTP	Real-time Transport Protocol
RTP/AVP	RTP Audio and Video Profile
RTR	Registration-Termination-Request
S/MIME	Secure MIME
SA	Security Association
SAA	Server-Assignment-Answer
SAD	Security Association Database
SAR	Server-Assignment-Request
SBLP	Service-Based Local Policy
S-CDR	SGSN-CDR
SCF	Session Charging Function
SCS	Service Capability Server
S-CSCF	Serving-CSCF
SCTP	Stream Control Transmission Protocol
SDF	Service Delivery Framework
SDP	Session Description Protocol
SDU	Service Data Unit
SEG	Security Gateway
SGSN	Serving GPRS Support Node
SGW	Signalling Gateway
SHA	Secure Hash Algorithm
SigComp	Signalling Compression
SIM	Subscriber Identity Module
SIP	Session Initiation Protocol
SIPS	Secure SIP
SL	Subscriber Locator
SLA	Service-Level Agreement
SLF	Subscription Locator Function
SM	Short Message
SMG	Special Mobile Group
SMS	Short Message Service
SNA	Subscribe-Notifications-Answer
SNMP	Simple Network Management Protocol
SNR	Subscribe-Notifications-Request
SPD	Security Policy Database
SPI	Security Parameter Index
SPT	Service Point Trigger
SQN	Sequence number
SRF	Single Reservation Flow
SR	Sender Report
SRV	Service records

SS7	Signaling System No. 7
SSF	Service Switching Function
SSRC	Synchronization Source
STP	Service Trigger Point
STUN	Session Traversal Utilities for NAT
TAS	Telephony Application Server
TBCP	Talk Burst Control Protocol
TCP	Transmission Control Protocol
TCP/IP	TCP/IP stack
TD-CDMA	Time Division/Code Division Multiple Access
THIG	Topology Hiding Inter-network Gateway
TIA	Telecommunications Industry Association (North America)
TIP	Terminating Identification Presentation
TIR	Terminating Identification Restriction
TISPAN	Telecommunications and Internet converged Services and Protocols for Advanced Networking
TLS	Transport Layer Security
TTA	Telecommunications Technology Association (South Korea)
TTC	Telecommunications Technology Committee (Japan)
TTL	Time To Live
TU	Transaction User
TURN	Traversal Using Relays around NAT
UA	User Agent
UAA	User-Authorization-Answer
UAC	User Agent Client
UAR	User-Authorization-Request
UAS	User Agent Server
UDA	User-Data-Answer
UDP	User Datagram Protocol
UDR	User-Data-Request
UDVM	Universal Decompressor Virtual Machine
UE	User Equipment
UICC	Universal Integrated Circuit Card
UL	Uplink
UMTS	Universal Mobile Telecommunications System
UNI	User to Network Interface
URI	Uniform Resource Identifier
URL	Universal Resource Locator
URN	Uniform Resource Name
USIM	Universal Subscriber Identity Module
UTRAN	UMTS Terrestrial Radio Access Network
VCC	Voice Call Continuity
VCC-AS	Voice Call Continuity Application Server
VDI	VCC Domain Transfer URI
VDN	VCC Domain Transfer Number
VHE	Virtual Home Environment

VMSC	Visited MSC
VoIP	Voice over IP
WAP	Wireless Application Protocol
WB	WideBand
WCDMA	Wideband Code Division Multiple Access
WLAN	Wireless Local Area Network
XCAP	XML Configuration Access Protocol
XDM	XML Document Management
XDMC	XML Document Management Client
XDMS	XML Document Management Server
XML	Extensible Markup Language
XRES	Expected response



# Index

- ;explicit parameter 357–9
- ;require parameter 357–9
- 100 (Trying) response 347, 363, 364
- 100rel mechanism 364
- 180 (Ringing) response 374, 381
- 183 (Session in Progress) response
  - callee identity 336
  - compression negotiation 362
  - ICID exchange 384
  - IMS example session 345–6
  - media authorization 384
  - media negotiation 363
  - provisional response reliability 364
  - resource reservation 374, 379
- 200 (OK) response
  - compression negotiation 299
  - GCID/ICID correlation 386
  - PRACK requests 348
  - REGISTER requests 268–9
  - registration 262, 307
  - registration-state information 307
  - resource reservation 379–80, 381
  - SA establishment 282, 284
  - IMS Voice Call Continuity 454–6
- 3G (third generation) systems 3, 10–3, 118, 125
- 3GPP *see* Third Generation Partnership Project
- 401 (Unauthorized) response
  - authentication 280, 282
  - port setting 286
  - SA establishment 280–2
- Security-Server header 291, 292, 293
- Sip-Sec-Agree 291
- 404 (Not Found) response 335
- 420 (Bad Extension) response 412
- a-line *see* attribute line
- AAR *see* AA-Request
- AA-Request (AAR) command 71, 73, 74
- ACA *see* Accounting Answer
- Accept-Contact header
- IMS Voice Call Continuity 434
- Accept header 306
- access
  - control mechanisms 185–6
  - independence 21–2
  - IP version 4
  - networks 134–5
  - multi-access functionality 22
  - PoC user policy 164–5
  - points 422
  - registration 280–7, 300–1
  - security 121–5, 280–7
- Access Control List (ACL) 152
- Access Point Names (APNs) 69, 422
- accounting 76, 81–2
- Accounting Answer (ACA) 77, 81–2, 87
- Accounting Request (ACR) 77, 81–2, 87
- ACL *see* Access Control List
- ACR *see* Accounting Request
- ad hoc PoC groups 178, 179
- Address Of Record (AOR) 310, 312
- addresses 127–8, 386–7

- AKA *see* Authentication and Key Agreement
- ALG *see* Application Level Gateway
- Anonymous Communication Rejection (ACR) 235
- answer modes, PoC 182–6
- AOR *see* Address Of Record
- APNs *see* Access Point Names
- Application Level Gateway (ALG) 128–9
- Application Servers (ASs) 27–8
- ISC 35
  - P-Asserted-Identity header 335
  - PoC 179
  - routing information 349–50, 418–20
  - service provision 91, 93–4
  - user profiles 90
- ASs *see* Application Servers
- assignment, S-CSCF 59–61
- attribute line (a-line) 392, 412
- Attribute Value Pairs (AVPs) 60, 71, 81–3
- audio streams, RTP/AVP 367–9
- authentication
- Cx reference point 38
  - proxy authentication 126
  - re-authentication 282–4, 287, 317–9
  - registration 273–6, 278–9
  - SIP 273
  - see also* identification issues
- Authentication and Key Agreement (AKA) protocol 114, 115–6
- Authentication Vector (AV) 276–7
- authorization
- bearer authorization 62–73
  - PoC groups 163–4
  - presence 147–9
  - rulesets 163–5
- AUTN *see* network authentication token
- auto-answer mode, PoC 182–5
- AV *see* Authentication Vector
- AVPs *see* Attribute Value Pairs
- Back-to-Back User Agent (B2BUA) 350, 426, 437, 449
- Bad Extension (420) response 412
- bandwidth modifiers 369–70
- barring
- communication 172
- PoC 188, 304
- bearer authorization 62–73
- bearer loss/recovery indication 74, 84
- bearer release function 84
- bearer traffic control mechanism 61–75
- BGCF *see* Breakout Gateway Control Function
- bidding-down attacks 124
- bidirectional streams 331
- billing systems 76–8
- bootstrapping 133–4
- branch parameters, Via header 340
- Breakout Gateway Control Function (BGCF) 95
- broadband
- fixed 100
  - Internet access 22
- buddy lists 151, 154
- business aspects 152
- BYE request 387–8
- c-line *see* connection information line
- Cablelabs 11
- call anchoring, Voice Call Continuity 429–46
- direction 429–31
- call flow 332
- Call Session Control Functions (CSCFs) 23, 273, 339
- see also* Interrogating-CSCF; Proxy-CSCF; Serving-CSCF
- callee capabilities 294–7
- caller/callee identities 333–7
- Call-ID header 263
- calling line identification presentation (CLIP) 242
- call-statefull SIP proxies 347
- CAMEL *see* Customized Applications for Mobile network Enhanced Logic
- capabilities 60, 105–6
- CCCCF *see* Charging Collection Function
- CDF *see* Charging Data Function
- CDR *see* Charging Data Record
- Cell Global ID (CGI) 300
- cellular networks 175–93

- CGF *see* Charging Gateway Function  
CGI *see* Cell Global ID  
charging 18–9, 75–86  
    function addresses 386–7  
    identifiers 73, 74, 84–6  
    IMS example session 383–7  
    reference points 45  
    registration 301  
Charging Collection Function (CCF) 387  
Charging Data Function (CDF) 76–7,  
    81–2, 84, 86  
Charging Data Record (CDR) 76–7, 78,  
    81–2  
Charging Detail Records (CDRs) 202  
Charging Gateway Function (CGF) 76–8  
Charging Trigger Function (CTF) 77, 78  
chat alias 209–10  
chat conferences 198, 199  
chat IM sessions 207  
chat PoC groups 178, 180, 188  
Ciphering Key (CK) 274, 278  
Circuit Switched (CS) networks 47, 94–6  
    adaptation 108  
Circuit Switched Core Network (CS CN)  
    15  
Circuit Switched Domain Routing  
    Number (CSRN) 449  
CK *see* Ciphering Key  
class information, QoS 69, 70  
clients  
    IM 203  
    PoC 178, 182  
combinational services, IMS/CS 95–6  
Common Open Policy Service (COPS) 62  
Common Policy 153–6, 172  
communication 15, 18, 178–80  
    IM 203, 210–1  
    barring of 210–1  
        participant information 211  
communication barring (CB) 172, 235  
Communication Deflection (CD) 237  
communication diversion 172, 235, 236  
Communication Forwarding Busy (CSB)  
    237  
Communication Forwarding No Reply  
    (CFNR) 237  
    Communication Forwarding on Not  
        Logged-in (CFNL) 237  
    Communication Forwarding  
        Unconditional (CFU) 236  
    Communication Hold (HOLD) 235, 238  
    comp parameter, compression negotiation  
        297–9  
    comp=SigComp parameter 297–9, 361  
    compression 102–5, 297–9, 360–2  
    Conference service (CONF) 235, 239  
    conferencing 198, 199, 221–31  
    confidentiality 124  
    configuration 133–4  
    connection information line (c-line) 367  
    connectivity  
        IMS/CS 94–6  
        IP 16–7, 15  
        peer-to-peer 3–4, 8, 16, 197–8  
    Contact header 262, 281, 295, 298, 306  
    ‘contact’ sub-elements, registration 310  
    Content-Length header 263  
    contexts, PDP 421  
    converged networks 4  
    conversation history 166–8, 203, 211  
        deleting stored messages 213–8  
        retrieving stored messages 212–3  
COPS *see* Common Open Policy Service  
correlation, charging information 85  
Credit Control Request/Answer 82–4  
cross-certification 121  
CS *see* Circuit Switched  
CSCFs *see* Call Session Control  
    Functions  
CSCN *see* Circuit Switched Core  
    Network  
CSeq header 365  
CSeq numbers 339–40  
CTF *see* Charging Trigger Function  
customer classes, media information 88  
Customized Applications for Mobile  
    network Enhanced Logic (CAMEL)  
        27–8, 40, 108  
Cx reference point 35, 36–7  
data handling 38, 40  
data manipulation *see* group management

- data rate values 64, 66–7, 69  
databases 26–7  
de-registration 319–26  
default public user identity 303–4  
deferred delivery messaging 203, 213–5  
deferred messaging 168, 203, 213–5  
delivery report 218  
derived public user identity 53  
Device Management (DM), OMA 427  
Device Management Object (MO) 427–8  
Dh reference point 40  
DHCP *see* Dynamic Host Configuration Protocol  
DHCPv6 *see* Dynamic Host Configuration Protocol for IP Version 6  
dialogs 338–9, 347–9  
dial-out IM groups 207  
dial-out sessions, PoC 178–9, 188  
Diameter  
  AAR/RAR commands 72–4  
  ACA 77, 81–2, 87  
  ACR 77, 81–2, 87  
  AVPs 81  
  direct debiting 79–80, 82  
  discovery, P-CSCF discovery 57, 58–9  
  Domain Name System (DNS) 58–9, 258, 264, 337  
  Domain Selection Function 426  
  Domain Transfer Function (DTF) 437  
  DTMF *see* dual-tone multi-frequency tones  
  dual stack operations 132–3  
  dual-tone multi-frequency (DTMF) tones 367  
  Dx reference point 38  
  Dynamic Host Configuration Protocol (DHCP) 58–9, 258  
  Dynamic Host Configuration Protocol for IP Version 6 (DHCPv6) 258–9  
  dynamic presence 140, 142  
  early IMS security 326–9  
  ECF *see* Event Charging Function  
  EDGE *see* Enhanced Data Rates for Global Evolution radio access  
  emergency calls 300  
  emergency sessions 100–2  
    CSCF 25–6, 100, 102  
    registration 101  
    setup 101–2  
  Encapsulated/ing Security Payload (ESP) 124  
  Enhanced Data Rates for Global Evolution (EDGE) radio access 10  
  entry point, IMS 58–9  
  errors, S-CSCF assignment 61  
  ESP *see* Encapsulated/ing Security Payload  
  establishment, sessions 181–3, 338–40, 373–4, 389–415  
  European Telecommunications Standards Institute (ETSI) 10  
  event attributes, registration 310–1  
  event-based charging 76, 81–3  
  Event Charging Function (ECF) 383  
  event packages 101  
  example session, IMS 331–423  
  expected result (XRES) 274  
  Expires header 306  
  Explicit Communication Transfer (ECT) 235, 245–6  
  feature tags 295  
  filter criteria 89, 91–2, 271, 349–50  
    IMS Voice Call Continuity 452, 459–60  
  Final Delivery Report 220  
  first-party registration 263  
  fixed networks 5–7, 94, 96, 98  
  floor moderators 190  
  flow identifiers 66, 68, 130  
  flow-based charging 80, 84  
  forking issues 58, 74, 105  
  FQDN *see* Fully Qualified Domain Name  
  From header 263, 272, 309, 333  
  Fully Qualified Domain Name (FQDN) 258  
  Gateway GPRS Support Node (GGSN) 16, 19, 24, 26, 30, 32, 250, 253, 420–3

- bearer traffic control 62–3 68–9, 72–3  
charging 85, 383  
early IMS security 326–7  
P-CSCF discovery 257  
registration 257  
roaming 249–50  
session establishment 338  
*see also* General Packet Radio Service  
gateways  
control functions 29  
security domains 120  
GBA *see* Generic Bootstrapping Architecture  
GBR *see* Guaranteed Bit Rate  
GCID *see* GPRS Charging Identifier  
General Packet Radio Service (GPRS) 10, 420–3, 249, 253  
charging 383  
IP connectivity 16–7  
IP versions 4/6 134–5  
P-CSCF 23–4, 58–9, 257  
resource reservation 381–2  
roaming 19, 249–50  
SGSN 32, 421  
*see also* Gateway GPRS Support Node  
Generic Bootstrapping Architecture (GBA) 125  
GGSN *see* Gateway GPRS Support Node  
Global System for Mobile Communications (GSM) 9, 10  
Globally Routable User Agent URI (GRUU) 304–5, 415–8  
Gm reference point 33–4  
GPRS *see* General Packet Radio Service  
GPRS Charging Identifier (GCID) 85–7, 383, 385–6  
GPRS-IMS-Bundled Authentication (GIBA) 326–9  
Gq reference point 62  
group advertisements 187–8  
Group Identity, IM 218  
group management 151–74  
group sessions 179–80  
GSM *see* Global System for Mobile Communications  
Guaranteed Bit Rate (GBR) 68  
Gx reference point 44–5, 69, 71, 75, 80, 84–5  
headers  
caller/callee identities 333–7  
GCID/ICID correlation 385–6  
INVITE requests 339–41  
provisional responses 364–5  
REGISTER requests 261–2  
routing information 338, 346  
SDP offer/answer mechanism 379–81  
user identities 306  
*see also* individual headers  
hist-settings 211  
Home Location Register (HLR) 26  
home networks 21, 119  
Home Subscriber Server (HSS) 26–7  
authentication 273  
early IMS security 328–9  
reference points 35–7  
registration 48, 261–2, 265, 272  
routing 342  
user identities 53  
HSS *see* Home Subscriber Server  
Hypertext Transfer Protocol (HTTP) 42, 125–6, 275  
ICID *see* IMS Charging Identifier  
I-CSCF *see* Interrogating-Call Session Control Function  
ID attribute 310  
identification issues  
barring identities 304  
caller/callee identities 333–7  
charging identifiers 73, 74, 84–6, 301, 383–4  
flow identifiers 66, 68, 130  
GCID 85–7, 383, 385–6  
ICID 85–7, 301, 383–4  
identity modules 57  
IMSI 53, 54  
IOI 383–4  
multiple users 49–50  
private user identities 52–6, 273–4, 302–3  
Public Identification 88

- identification issues (*Continued*)  
 public service identities 55, 418–20  
 trust domains 123  
 UICC cards without ISIM 303  
 user identities 26, 49–50, 51–6,  
   273–4, 301–16  
*see also* IP Multimedia Services  
 Identity Module; public user  
 identities  
 identity minting 154  
 IKE *see* Internet Key Exchange  
 IM-SSF *see* Internet Protocol Multimedia  
   Service Switching Function  
 immediate messaging 195–7  
 implicit registration 49–50  
 IMS *see* Internet Protocol Multimedia  
   Subsystem  
 IMS Application Reference Identification  
   (IARI) 295–7, 352  
 IMS Charging Identifier (ICID) 85–7,  
   301, 383–4  
 IMS communication service identification  
   (ICSI) 234–5, 294–7, 352–60  
 IMS Management Object (IMS MO)  
   255–6  
 IMS-MGW *see* Internet Protocol  
   Multimedia Subsystems Media  
   Gateway  
 IMS Service Control (ISC) 35, 202  
 IMSI *see* International Mobile Subscriber  
   Identifier  
 Incoming Communication Barring (ICB)  
   235  
 incoming sessions 183–6  
 information  
   charging 76–8, 81, 85–6  
   media policies 88–9  
   service triggers 89–90  
 initial filter criteria 89, 91–2  
 Instant Messaging 198–9  
 instant messenger (OMA)  
   barring of communication 210–1  
   chat alias 209–10  
   client 203  
   communication 203  
   conversation history function 212–3  
   creating messaging history 212  
   Immediate Messaging 204–6  
   participant information 211  
   private message during session 209  
   server 202–3  
   session based messaging 206–8  
 instant personal alerts 186–7, 188  
 Integrated Services Digital Network  
   (ISDN) 3  
 Integrity Key (IK) 274, 278  
 integrity protection 126, 279  
 inter-domain scenarios, IP versions 4/6  
   133  
 International Mobile Subscriber Identifier  
   (IMSI) 53, 54  
 Internet Assigned Number Authority  
   (IANA) 256, 353  
 Internet Protocol (IP)  
   connectivity 16–7, 15  
   GPRS 16, 134–5  
   IMS definition 4  
   policy control 17  
   SIP registration 253  
   version 4 16, 126–35  
   version 6 16, 126–35  
 Internet Protocol Connectivity Access  
   Networks 199  
 Internet Protocol Multimedia Service  
   Switching Function (IM-SSF) 27–8  
 Internet Protocol Multimedia Routeing  
   Number (IMRN) 430  
 Internet Protocol Multimedia Subsystem  
   (IMS) 3–4  
   architecture 15–46  
   concepts 47–135  
   example session 331–423  
   reference points 33–46  
   registration example 253–329  
   Terminating Request 432  
   Termination field set 432  
 Internet Protocol Multimedia Subsystems  
   Media Gateway (IMS-MGW) 29  
 Internet Protocol security (IPsec)  
   IMS registration 255  
   SAs 280–7, 291, 292–3, 331  
 Internet Relay Chat (IRC) 197–8

- Inter-Operator Identifier (IOI) parameter 383–4
- Interrogating-Call Session Control Function (I-CSCF) 23, 436
- CS users 95
- databases 26, 27
- PSI routing 419
- reference points 34–7
- registration 48, 255, 264, 273, 255
- routing information 337, 342–4
- S-CSCF assignment 60–1
- session initiation 50
- interworking functions 29
- intra-domain scenarios, IP versions 4/6 133
- INVITE requests
- application servers 350–1
  - callee identity 335, 336
  - compression negotiation 360–1
  - conferencing 224
  - emergency calls 102
  - ICID exchange 384
  - IMS Voice Call Continuity 433–4
  - P-Asserted-Identity header 335
  - PoC session establishment models 181–3
  - public user identities 333
  - registration 255
  - re-transmission 347
  - roaming 331
  - routing 337–41
  - SDP 434–5
  - SDP offer/answer model 366
  - service provision 91–2
  - session establishment 389–90
  - session initiation 50–1
  - SIP 214, 253, 437, 438–40
  - transactions 340–1
  - VCC 111
- IOI *see* Inter-Operator Identifier
- IP *see* Internet Protocol
- IP Multimedia Services Identity Module (ISIM) 57
- AKA 115–6
- authentication 274, 278
- SIP registration 261
- user identities 302–3
- IPsec *see* Internet Protocol security
- Iq reference point 45
- IRC *see* Internet Relay Chat
- ISAKMP *see* Internet Security Association and Key Management Protocol
- ISC *see* IMS Service Control reference point
- ISDN *see* Integrated Services Digital Network
- ISIM *see* IP Multimedia Services Identity Module
- Iu interface 10
- I-WLAN 426
- Ix reference point 45
- join-in messaging sessions 206–7
- keys 115–6, 120–1, 124–5
- Large Message Mode 203, 206, 219–20
- layered design, 3GPP 21–2
- Local Connection Addresses field set 432
- local dialling plans 98–9
- Local IMS Resources field set 432
- location 35–8, 300–1
- Location Info Answer (LIA) 436–7
- LPDP *see* Local Policy Decision Point
- m-line *see* media line
- Ma reference point 35
- Management Object, Device Management 427–8
- manual answers 183–5
- Maximum Bit Rate (MBR) 68
- media
- authorization 38, 381–2
  - charging components 18–9
  - m-lines 367, 367, 377, 395
  - negotiation 181–2, 362–73
  - parameter negotiation 181–2
  - PDP contexts 373–83
  - policing 382
  - policy information 88–9
  - streams 332, 367–74

- Media Gateway Control Function (MGCF) 94–6, 431–3
- MEGACO *see* Media Gateway Control Protocol
- MESSAGE method 195, 187–8  
SIP 199–201, 212, 218–9
- Message Session Relay Protocol (MSRP) 198  
instant messaging 217–8  
Switch 199
- Message Waiting Indication (MWI) 235, 240–1
- messages  
SIP 195, 188–9, 332  
TBCP 190–1
- messaging 7, 195–220, 390  
history 212  
immediate 195–7  
session-based 197–8  
interworking 198–201  
instant, by OMA 201–2
- Mg reference point 43
- MGCF *see* Media Gateway Control Function
- MGW 431–3
- Mi reference point 42
- MID *see* Media Stream Identification
- MIME *see* Multipurpose Internet Mail Extension
- Mj reference point 43
- Mk reference point 43
- MI reference point 45
- Mm reference point 43
- MMD *see* Multimedia Domain
- MMS *see* Multimedia Messaging Service
- Mn reference point 44
- mobile networks 5–7, 96–7
- Mobile Station ISDN (MSISDN) 25, 57, 105–6, 428
- Mp reference point 43–4
- MPV *see* Music Photo Video
- Mr reference point 43
- MRFC *see* Multimedia Resource Function Controller
- MRFP *see* Multimedia Resource Function Processor
- MSISDN *see* Mobile Station ISDN
- MSRP *see* Message Session Relay Protocol
- multi-access functionality 22
- Multimedia Domain (MMD) 11, 126
- Multimedia Messaging Service (MMS) 197, 198
- Multimedia Resource Function Controller (MRFC) 27–8, 43
- Multimedia Resource Function Processor (MRFP) 27–8
- multimedia telephony 233–46  
service management 171–4  
SIP 234  
supplementary services 235–46
- multiparty conference (MPTY) 239
- multiple devices, single user identity 57–8
- multiple terminals 294  
registration 315–6
- multiple user identities, registration 49–50
- Multipurpose Internet Mail Extension (MIME) 198  
group advertisement 187  
group management 156  
participant information 189  
session-based messaging 197–8
- Music Photo Video (MPV) video stream encoding 367
- Mx reference point 45
- NAS *see* Network Access Server
- NATs *see* Network Address Translators
- NDS *see* Network Domain Security
- Network Address Translators (NATs) 126–31
- network authentication token (AUTN) 274
- Network Domain Security (NDS) 114, 118–21
- network-initiated de-registration 325–6
- network-initiated re-authentication 317–9
- network-initiated session release 389
- network interworking 20

- network nodes 55, 56  
Network Time Protocol (NTP) 367  
network-to-network interface (NNI) 431  
Next Generation Network (NGN) 11  
Not Found (404) response 335  
Notify Released Bearer field set 432  
NOTIFY requests  
  group management 156  
  network-initiated re-authentication 318  
  presence 145–6, 148–9  
  registration-state information 307–8,  
    309–10, 311  
  user-initiated de-registration 325–6  
NTP *see* Network Time Protocol
- O-Asserted-Identity header 433  
o-line, SDP 366, 371  
OCS (Online Charging System) 76–7,  
  79–80, 82–4  
offer/answer model *see* SDP offer/answer  
  model  
offline charging 18, 77–9, 81–2  
OK response *see* 200 (OK) response  
OMA *see* Open Mobile Alliance  
online charging 18, 76, 79–80, 82–4  
Open Mobile Alliance (OMA) 142, 175,  
  177–8, 188, 159–60  
  IM Immediate Messaging 204–6  
  IM session based messaging 206–8  
  instant messaging 199, 201–20  
  SIMPLE IM 199  
Open Service Architecture (OSA) 10,  
  27–8  
Originating Identification Presentation  
  (OIP) 235, 242–3  
Originating Identification Restriction  
  (OIR) 235, 243–4  
Originating Identification Services (OIR)  
  172  
OSA *see* Open Service Architecture  
Outcoming Communication Barring  
  (OCB) 235  
out-of-band provisioning mechanisms 134
- P-Access-Network-Info 300  
P-Asserted Identity headers 334–5,  
  336–7
- P-Asserted-Service header 434  
P-Associated URI header 303–4  
P-Called-Party-ID header 336  
P-Charging-Function-Address header 386  
P-Charging-Vector header 301, 385–6,  
  434  
P-CSCF *see* Proxy-Call Session Control  
  Function  
P-Preferred-Identity 333, 334  
Packet Data Protocol (PDP) 254, 255,  
  257, 421–3  
packet-switched networks 4, 6, 9  
page-mode messaging 185–7, 218–9  
Pager Mode 203  
parallel forking 58  
parallel services, IMS/CS 107  
participant information, PoC 189  
Path header 261, 262, 270–1  
payload types 367–9  
PDF *see* Policy Decision Function  
PDP *see* Packet Data Protocol; Policy  
  Decision Point  
peer-to-peer connectivity 3–4, 8, 16,  
  197–8  
PEPs *see* Policy Enforcement Points  
personal alerts 186–7, 188  
PIB *see* Policy Information Base  
PoC *see* Push to talk over Cellular  
policy control, IP 17  
Policy Decision Function (PDF)  
  bearer authorization 62–3  
  charging 85  
  P-CSCF 23–4  
  support functions 30  
Policy Decision Point (PDP)  
  bearer authorization 64, 68–9, 72–3  
  charging 85, 383  
  media contexts 367, 369, 370–3  
  session establishment 390  
  user-initiated session release 388  
  policy information, media 88–9  
  port setting 285–8  
  PRACK requests  
  compression negotiation 361–2  
  provisional response reliability 365  
  routing 348

- PRACK requests (*Continued*)  
     SDP offer/answer model 370, 379  
 pre-arranged PoC sessions 178, 179, 188  
 preconditions  
     resource reservation 374–9  
     session establishment 390  
     session set-up fallback 374  
 Preferred Domain 428, 429  
 pre-reserved resources, session  
     establishment 390  
 presence 139–49, 151, 152, 153–4  
     architecture 142–4  
     authorization 149  
     contributing to business 140–1  
     definition 141–2  
     dynamic 140, 142  
     publishing 144–5  
     services 140  
     subscribing 145–7  
     user groups 139  
     watcher information 147–9  
 presentity 143–6, 149  
 privacy 123  
 private message during IM session 209  
 private user identities 52–6, 273–4,  
     302–3  
 protocol version line (v-line) 366  
 protocols *see* individual protocols  
 provisional responses 364–5  
 Proxy-Call Session Control Function  
     (P-CSCF) 23–4, 30, 100, 101, 249  
     access information 300  
     authentication 274–5, 276, 278–9  
     bearer loss/recovery 74  
     charging-related information 384–386  
     compression negotiation 297–9, 360–2  
     discovery 47, 257–60  
     early IMS security, 327, 328  
     Gq reference point 62  
     IP versions 4/6 133–4  
     key management 124–5  
     location information 300  
     network-initiated session release 388  
     P-Asserted-Identity header 335, 336  
     port setting 285–7  
     re-authentication 282–4  
     reference points 33–5, 45  
     REGISTER requests 263–4, 270–1,  
         292–3  
     registration 48–9, 255, 253  
     roaming 249–50, 331  
     routing information 340–2, 344–7  
     session initiation 50–1  
     Sip-Sec-Agree 289–94  
     SUBSCRIBE requests 306–9  
     user identities 302  
 Proxy-Require header 291, 292, 293  
 PSIs *see* Public Service Identities  
 PSTN *see* Public Switched Telephone  
     Network  
 PUA *see* Presence User Agent  
 Public Identification 88  
 Public Safety Answering Point (PSAP)  
     100  
 Public Service Identities (PSIs) 55  
     routing of 418–20  
 Public Switched Telephone Network  
     (PSTN) 3  
 public user identities 50–6  
     default identity 303–4  
     P-Called-Party-ID header 336  
     registration 255, 261, 302–3  
     roaming 250  
 PUBLISH method 192–3  
     SIP 215  
 publishing presence 144–5  
 Push to talk over Cellular (PoC) 175–93  
     features 178–89  
     functional distribution 179  
     group management 152, 159–60,  
         162–5  
     PoC groups 162–5  
     server 178, 179–85, 187  
     service settings 192–3  
     user access policy 164–5  
     user plane 189–92  
     XDM 159, 163–5  
     XDMS 160, 163, 176, 185  
 P-Visited-Network-ID 300  
 QoS *see* Quality of Service  
 quality feedback, PoC 189, 191–2

- Quality of Service (QoS) 17  
  bearer authorization 62–73  
Packet Data Control 421  
  preconditions mechanism 376–7,  
    378–9  
UMTS 63–4, 66, 67–9
- radio access, W/TD-CDMA 9–10  
radio services 175  
  *see also* General Packet Radio Service
- RADIUS 326
- random challenge (RAND) 274
- Real-time Transport Control Protocol (RTCP) 66–7, 69, 175, 191–2
- Real-time Transport Protocol (RTP) 17, 175, 191–2, 338  
re-authentication 282–4, 287, 318–9  
Re-Auth-Request (RAR) command 71, 73, 74, 75
- Receiver Reports (RRs) 191–2
- Record-Route header  
  183 (Session in Progress) response 346  
  comp parameter 298  
  IMS Voice Call Continuity 449  
  INVITE requests 341–2  
  REGISTER requests 261  
    routing 338
- REFER, SIP 215
- reference points 33–46, 80–6, 176  
‘reginfo’ element, registration 311
- REGISTER requests  
  authentication 275  
  compression negotiation 298–9  
  early IMS security 326–9  
  IPsec SAs 292–3  
  P-CSCF 263–4, 270–1, 290–1  
  registration 255, 260–73, 253  
  SA establishment 280–2  
  Sip-Sec-Agree-related headers 291–3  
  user-initiated de-registration 325–6
- registration 48–9  
  access 280–7, 300  
  authentication 273–9  
  charging-related information 301  
  compression negotiation 297–9  
  de-registration 319–26
- early IMS security 326–9  
emergency 101  
IMS example 253–329  
location information 37, 300–1  
P-CSCF discovery 256  
re-authentication 318–9  
re-registration 317–9  
S-CSCF 48–9, 60, 253–5, 264–8, 319–22  
security 279–87  
signalling PDP contexts 256–7  
SIP 253–329  
state information 262, 285, 302, 305–16  
transport protocols 259–60  
user identities 253, 259, 301–16  
reliability, provisional responses 364–5  
request URI 335–6  
Require header 290, 291, 292  
re-registration 317–9  
Reserve Value field set 432  
resource lists 152  
resource reservation 17, 373–83, 390  
restricted/unrestricted chat PoC groups 180  
retrieval  
  Cx reference point 38  
  stored messages 212–3
- Rf reference point 80, 81–2
- Ringing (180) response 374–5, 379, 381
- RLS (Resource List Server) 156
- Ro reference point 82–4
- roaming 16, 19–20, 98, 235, 249–50, 331
- Route header  
  compression negotiation 299  
  INVITE requests 339–41  
  REGISTER requests 261, 264, 269, 271–2  
  SUBSCRIBE requests 306  
  routing 260–73, 285, 337–60, 418–20
- RRs *see* Receiver Reports; Resource Records
- RTCP *see* Real-time Transport Control Protocol
- RTP Audio and Visual Profile (RTP/AVP) 367–9

- RTP Control Protocol 369  
*RTP* *see* Real-time Transport Protocol  
 rulesets 163–5, 169  
*see also* authorization  
 Rx reference point 80, 84
- SAD *see* Security Association Database  
 SAs *see* Security Associations  
 SBLP *see* Service-Based Local Policy  
 S-CSCF *see* Serving-Call Session Control Function  
 SCTP *see* Stream Control Transmission Protocol  
 SDP offer/answer model  
   media negotiation 363, 365–71  
   PRACK requests 372, 381  
   resource reservation 373–80  
 SDP *see* Session Description Protocol  
 secret keys, AKA 115–6  
 security aspects  
   authentication 279  
   communication 18  
   early IMS security 326–9  
   NDS domains 118–21  
   services 113–26  
   SIP 289–94  
   Sip-Sec-Agree 289–94  
*see also* Internet Protocol security  
 Security Associations (SAs)  
   access security 279–89  
   IPsec 279–89, 291, 292–3, 331  
   key management 120–1  
   Sip-Sec-Agree 292–3  
 Security Client header 291, 292–3  
 Security Gateway (SEG) 30, 120–1  
 Security-Server header 293  
 Security-Verify header 293  
 SEG *see* Security Gateway  
 Sender Reports (SRs) 191–2  
 sequence number (SQN) 274  
 sequential forking 58  
 servers  
   IM 202–3  
   PoC 178, 179–85, 187  
   presence 146  
   Resource List Server 156
- XDMS 176, 185  
*see also* Application Servers; Home Subscriber Server
- service  
   control model 20–1  
   functions 27–8  
   PoC settings 192–3  
   profiles 24–5, 86  
   provision 90–4  
   settings, IM 215–7  
   triggering information 89–90  
*see also* Quality of Service  
 Service-Based Local Policy (SBLP) 62  
*see also* Internet Protocol  
 service management  
   multimedia telephony 171–4  
 Service-Route header 261, 262, 273  
 Service Trigger Point 91, 353  
 services  
   IMS examples 7–9  
   presence 139–40, 142–4  
   security 113–26  
 Serving-Call Session Control Function (S-CSCF) 24–5, 100  
   100 (Trying) response 347  
   access information 300  
   AS selection 93  
   assignment 59–61  
   authentication 273–8, 317–8  
   callee identity 335–6  
   charging-related information 385–7  
   CS users 94–5  
   de-registration 319–22  
   early IMS security 328–9  
   filter criteria 349–50, 352  
   IMS Voice Call Continuity 447  
   location information 300  
   media policy information 88–9  
   network-initiated re-authentication 317–8  
   network-initiated session release 389  
   NOTIFY requests 307–8, 309  
   P-Asserted-Identity header 335–6  
   PSI routing 418–20  
   reference points 34–5, 43  
   REGISTER requests 269–72

- registration 48–9, 255, 266–72, 255  
roaming 331  
routing 337–8, 341–4, 418–20  
SA establishment 282  
service-triggering information 89  
session initiation 50  
SUBSCRIBE requests 306–7  
user identities 52–6  
Serving GPRS Support Node (SGSN) 16, 19, 26, 32, 421  
session-based messaging 197–8  
Session Description Protocol (SDP)  
    CSCF 23  
    roaming 331  
    *see also* SDP offer/answer model  
session establishment 181–3, 338–40, 373–4, 389–415  
Session Identity, IM 218  
Session in Progress response *see* 183  
    (Session in Progress) response  
Session Initiation Protocol (SIP) 6, 10–1  
    access security 121–5  
    authentication 273  
    charging arrangements 76  
    compression 102–5  
    conferencing 221  
    CSCF 23  
    dialogs 338  
    IP policy control 17  
    MESSAGE method 195, 186–7  
    messages 195, 187–8, 332  
    presence 139  
    PUBLISH method 192–3  
    reference points 33–46  
    registration 48–9, 255, 260–74  
    resource lists 156–9  
    roaming 331  
    routing 260–73  
    session establishment 373–4  
    SIP AS 27–8  
    Sip-Sec-Agree 289–94  
    transactions 340  
    transport protocols 259–60  
    UE 289–94, 390–1  
    *see also* INVITE requests; SIP URI;  
        SUBSCRIBE requests  
session mode messaging 203, 219–20  
sessions  
    charging arrangements 81–4  
    communication networks 15  
    emergency sessions 25–6, 102–3  
    establishment 181–3, 338–40, 373–4, 389–415  
    IMS example 331–423  
    initiation procedures 50–1  
    messaging 185–7  
    multimedia components 77  
    release 387–9  
        *see also* Session Initiation Protocol  
SGSN *see* Serving GPRS Support Node  
SGW *see* Signalling Gateway  
Sh reference point 39  
Shared Group XDMS 168–70  
Shared List XDMS 168, 170  
Shared Policy XDMS 168, 170–1  
Shared Profile XDMS 168, 171  
shared secrets 278  
Si reference point 40–2  
Signalling Compression (SigComp) 103, 297–9  
Signalling Gateway (SGW) 29  
signalling PDP context establishment 257  
simultaneous PoC sessions 180–1  
SIP Security Mechanism Agreement  
    (Sip-Sec-Agree) 289–94  
SIP *see* Session Initiation Protocol  
SIP Uniform Resource Identifier (SIP URI)  
    conferencing 221  
    IMS/CS connectivity 94  
    registration 250  
    roaming 250  
    user identities 51  
Sip-Sec-Agree *see* SIP Security Mechanism Agreement  
SLF *see* Subscription Locator Function  
s-line, SDP 367  
SMS 198–9  
Spec(T), trust domains 122–3  
SQN *see* sequence number  
SRs *see* Sender Reports  
SRV RRs *see* service records

- SS7 ISUP (Signalling System Number 7 ISDN User Part) 430  
 standalone transactions 349  
 state attribute, registration 310  
 status, presence 139  
**SUBSCRIBE** requests  
     conferencing 228  
     group management 156–7  
     offline charging 76, 77  
     P-CSCF 305–9  
     presence 145–7  
     Push to talk over Cellular 189  
     registration 308–9  
     SIP 211  
**Subscription Locator Function (SLF)**  
     26–7, 342  
**Subscription/Notification procedures** 40  
**Subscription-State header** 309  
 support functions, IMS 30  
**Supported header** 364
- Talk Burst Control Protocol (TBCP)**  
     190–1  
 talk bursts 179, 189–91  
**TBCP** *see* Talk Burst Control Protocol  
**TCP** *see* Transmission Control Protocol  
**TD-CDMA** *see* Time Division/Code Division Multiple Access (TD-CDMA) 9  
**tel URI** *see* telephone Uniform Resource Identifier  
**tel URL** *see* telephone Uniform Resource Locator  
 telephone event representation 368  
 telephone networks 3–4  
 telephone Uniform Resource Locator (tel URL) 51, 52, 55, 94, 105, 441–5  
**Telephony Application Server (TAS)**  
     234–5, 353  
 temporary public user identity 54–6  
**Terminating Identification Presentation (TIP)** 173, 235, 244  
**Terminating Identification Restriction (TIR)** 173, 235, 244–5  
**THIG** *see* Topology Hiding Inter-network Gateway  
**Third Generation Partnership Project (3GPP)**
- layered design 21–2  
 messaging 198–201  
**Release** 4 10  
     Release 5 10–1, 21, 40, 62, 76, 80, 372  
     Release 6 10–1, 21–2, 40, 62, 68, 80, 119, 127  
     Release 7 12–3, 22, 62, 71–2, 97, 99, 100, 107, 113  
     Release 8 13, 22  
     Release 99 9  
     service development 21  
     XCAP 156–9  
**third generation systems** *see* 3G  
**third-party registration** 271–2  
 tightly coupled conferences 221  
**Time Division/Code Division Multiple Access (TD-CDMA)** 9  
**TISPAN** 11–2, 96, 97  
**t-line, SDP** 367  
**TLS** *see* Transport Layer Security  
**To header**  
     caller/callee identities 333  
     IMS Voice Call Continuity 433  
     NOTIFY requests 309  
     REGISTER requests 262  
     SUBSCRIBE requests 306  
     third-party registration 272  
**Topology Hiding Inter-network Gateway (THIG)** 31  
 traffic classes 67–9  
 traffic control mechanism 61–75  
 transactions, routing 338–40  
 transit networks 96–7  
 translation, addresses 127–8  
**Transmission Control Protocol (TCP)**  
     198, 259, 288  
 transport protocols, SIP 259–60  
 trigger points 89–90  
 trust domains 122–3, 335  
 Trying (100) response 347, 363, 364  
**TU** *see* transaction user  
 tunnelling mechanisms 134–5
- UA** *see* User Agent  
**UAC** *see* User Agent Client

- UAS *see* User Agent Server  
UDP *see* User Datagram Protocol  
UDVM *see* Universal Decompressor  
    Virtual Machine  
UE *see* User Equipment  
UICC *see* Universal Integrated Circuit Card  
UMTS *see* Universal Mobile Telecommunications System  
UMTS Terrestrial Radio Access Network (UTRAN) 10  
Unauthorized response *see* 401  
    (Unauthorized) response  
Uniform Resource Identifier (URI)  
    conferencing 223, 224, 225–6  
    P-Asserted-Identity header 335–6  
    P-Called-Party ID header 336  
    public user identities 333  
    registration 301–16  
    request URI 336  
    resource lists 152, 153  
    *see also* SIP(S) Uniform Resource Identifier  
Uniform Resource Name (URN) 101–2  
unit reservation, online charging 79–80, 83  
Universal Integrated Circuit Card (UICC) 57, 114–5  
Universal Mobile Telecommunications System (UMTS)  
    AKA 114, 115–6  
    IP connectivity 16  
    QoS 62, 63, 68–9  
Universal Subscriber Identity Module (USIM) 57, 303  
unregistered users 60  
unsupported media types 382  
UPDATE requests 379–81, 385–6  
URI *see* Uniform Resource Identifier  
User Agent Server (UAS) 437  
User Data Answer (UDA) 440  
User Datagram Protocol (UDP) 259, 264, 288  
User Equipment (UE) 202  
    authentication 274–5  
    authorization 68–9  
capability exchange 105–6  
compression negotiation 297–9, 360–2  
CSCF 23  
early IMS security 327–8  
GPRS 420–3  
IP connectivity 16–7  
P-CSCF discovery 58–9  
P-Preferred Identity 333, 334  
port setting 285–8  
pre-reserved resources 390  
QoS 17  
re-authentication 282–4  
registration 48–50, 253, 255  
resource reservation 373  
roaming 331  
routing 338, 340–1, 344–8  
S-CSCF 24–5  
SDP offer/answer model 363  
secure communications 18  
session initiation 50–1  
SIP 289–94, 390  
standalone transactions 349  
SUBSCRIBE requests 305–8  
user identities 51–6, 302–3, 303–4  
user-initiated de-registration 325–6  
user-initiated re-registration 317  
user-initiated session release 387–8  
user-plane  
    IM 217–8  
    PoC 189–92  
users  
    data handling procedures 35  
    identities 26, 49–50, 51–6, 272, 301–16  
    PoC access policy 164–5  
    presence 139–40  
    privacy 123  
    profiles 86–90  
USIM *see* Universal Subscriber Identity Module  
Ut interface 125  
Ut reference point 46  
UTRAN *see* UMTS Terrestrial Radio Access Network

- VCC *see* Voice Call Continuity  
Via header  
  branch parameter 340  
  comp parameter 298  
  IMS Voice Call Continuity 434, 448,  
    454  
  INVITE requests 344–5  
  REGISTER requests 261, 263, 264  
  routing information 340  
  UDP/TCP routing 288  
Visited Mobile Switching Centre (VMSC)  
  110, 429  
video streams, RTP/AVP 367–9  
video-sharing 105–7  
visited networks 21, 99–100  
v-line *see* protocol version line  
Voice Call Continuity 107–13  
  Applications Server (VCC-AS) 425–6,  
    447–9  
  call anchoring 429–46  
  Communication Continuity  
    Configuration Parameters  
      427–9  
  Domain Transfer Number 428  
  domain transfer procedure 111–2,  
    457–62, 465–8  
  Domain Transfer URI 428  
  example 425–69  
  functionality 108  
  initial call 429–57  
  related standards 468–9  
  session initiation and termination  
    108–11  
  supplementary services 113  
video 463–5  
voice over IP (VoIP) 107–8, 132  
  
watcher information event template  
  package (winfo) 147–9  
watchers, presence 143, 147–9  
WCDMA *see* Wideband Code Division  
  Multiple Access  
web pages 151  
Wideband Code Division Multiple Access  
  (WCDMA) 10  
winfo *see* watcher information event  
  template package  
Wireless Local Area Network (WLAN) 6,  
  11, 22, 107–8, 111  
WLAN *see* Wireless Local Area Network  
worst case scenarios, media policing 382  
WWW-Authenticate header 277–8  
  
XCAP *see* XML Configuration Access  
  Protocol  
XDM *see* XML Document Management  
XDMS *see* XML Document Management  
  Server  
XML Configuration Access Protocol  
  (XCAP) 153, 156–9  
XML Document Management (XDM)  
  153, 159–61  
XML Document Management Server  
  (XDMS) 159–62, 165–8, 176, 185,  
    202  
XRES *see* expected result  
  
Za/b interfaces 118