

[MS-SIPREGE]:

Session Initiation Protocol (SIP) Registration Extensions

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft [Open Specification Promise](#) or the [Community Promise](#). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.
- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

Revision Summary

Date	Revision History	Revision Class	Comments
4/4/2008	0.1		Initial version
4/25/2008	0.2		Revised and edited technical content
6/27/2008	1.0		Revised and edited technical content
8/15/2008	1.01		Revised and edited technical content
12/12/2008	2.0		Revised and edited technical content
2/13/2009	2.01		Revised and edited technical content
3/13/2009	2.02		Revised and edited technical content
7/13/2009	2.03	Major	Revised and edited the technical content
8/28/2009	2.04	Editorial	Revised and edited the technical content
11/6/2009	2.05	Minor	Revised and edited the technical content
2/19/2010	2.06	Editorial	Revised and edited the technical content
3/31/2010	2.07	Major	Updated and revised the technical content
4/30/2010	2.08	Editorial	Revised and edited the technical content
6/7/2010	2.09	Minor	Updated the technical content
6/29/2010	2.10	Editorial	Changed language and formatting in the technical content.
7/23/2010	2.10	No Change	No changes to the meaning, language, or formatting of the technical content.
9/27/2010	3.0	Major	Significantly changed the technical content.
11/15/2010	3.0	No Change	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	3.0	No Change	No changes to the meaning, language, or formatting of the technical content.
3/18/2011	3.1	Minor	Clarified the meaning of the technical content.
6/10/2011	3.1	No Change	No changes to the meaning, language, or formatting of the technical content.
1/20/2012	4.0	Major	Significantly changed the technical content.
4/11/2012	4.0	No Change	No changes to the meaning, language, or formatting of the technical content.
7/16/2012	4.0	No Change	No changes to the meaning, language, or formatting of the technical content.
10/8/2012	5.0	Major	Significantly changed the technical content.
2/11/2013	5.1	Minor	Clarified the meaning of the technical content.
7/30/2013	5.2	Minor	Clarified the meaning of the technical content.

Date	Revision History	Revision Class	Comments
11/18/2013	5.2	No Change	No changes to the meaning, language, or formatting of the technical content.
2/10/2014	5.2	No Change	No changes to the meaning, language, or formatting of the technical content.
4/30/2014	5.3	Minor	Clarified the meaning of the technical content.
7/31/2014	6.0	Major	Significantly changed the technical content.
10/30/2014	6.1	Minor	Clarified the meaning of the technical content.
3/30/2015	7.0	Major	Significantly changed the technical content.

Table of Contents

1	Introduction	8
1.1	Glossary	8
1.2	References	12
1.2.1	Normative References	12
1.2.2	Informative References	13
1.3	Overview	13
1.4	Relationship to Other Protocols	13
1.5	Prerequisites/Preconditions	13
1.6	Applicability Statement	13
1.7	Versioning and Capability Negotiation	13
1.8	Vendor-Extensible Fields	13
1.9	Standards Assignments.....	13
2	Messages.....	14
2.1	Transport.....	14
2.2	Message Syntax.....	14
2.2.1	Extensions to REGISTER Requests and Responses	14
2.2.1.1	SIP REGISTER Request Format	14
2.2.1.2	SIP REGISTER Response Format	14
2.2.1.3	ms-keep-alive Header Field Syntax	15
2.2.1.4	Presence-State Header Field Syntax	15
2.2.1.5	Supported Header Field Extensions	15
2.2.1.6	Ms-Subnet Header Field Syntax	16
2.2.1.7	Ms-Device-Info Header Field Syntax	16
2.2.1.8	P-Preferred-Registrar Header Field Syntax	17
2.2.1.9	Extensions to Server Header	17
2.2.1.10	Extensions to Contact Header	17
2.2.1.11	Deregister NOTIFY Request Format	17
2.2.1.11.1	subscription-state Header.....	17
2.2.1.11.2	Registration-Notify Event Header.....	17
2.2.1.11.3	Content-Type Header.....	17
2.2.1.11.4	Ms-Diagnostics-Public Header	18
2.2.1.12	Survivable Mode NOTIFY Request Format.....	18
2.2.1.12.1	subscription-state Header.....	18
2.2.1.12.2	Registration-Notify Event Header.....	18
2.2.1.12.3	Content-Type Header.....	18
2.2.1.12.4	Ms-Diagnostics-Public Header	18
2.2.1.13	text/registration-event Message Body.....	18
2.2.2	In-band Provisioning Messages.....	19
2.2.2.1	In-band Provisioning Request	19
2.2.2.2	In-band Provisioning Response	20
2.2.2.3	Data Model for application/vnd-microsoft-roaming-provisioning-v2+xml Documents	20
2.2.2.4	Data Model for Requests	20
2.2.2.5	Data Model for Responses	22
2.2.2.5.1	Data Model for ServerConfiguration provisionGroup	22
2.2.2.5.2	Data Model for meetingPolicy provisionGroup.....	29
2.2.2.5.3	Data Model for ucPolicy provisionGroup	32
2.2.2.5.4	Data Model for publicationGrammar provisionGroup.....	33
2.2.2.5.5	Data Model for userSetting provisionGroup	48
2.2.2.5.6	Data Model for endpointConfiguration provisionGroup	49
2.2.2.5.7	Data Model for locationPolicy provisionGroup	56
2.2.2.5.8	Data Model for mediaConfiguration provisionGroup.....	57
2.2.2.5.9	Data Model for presencePolicyV2 provisionGroup	58
2.2.2.5.10	Data Model for privacyPublicationGrammar provisionGroup.....	59

2.2.2.5.11	Data Model for persistentChatConfiguration provisionGroup	70
2.2.2.5.12	Data Model for publicProviders provisionGroup	71
3	Protocol Details	73
3.1	Basic Registration	73
3.1.1	Client Role	73
3.1.1.1	Abstract Data Model	73
3.1.1.2	Timers	74
3.1.1.3	Initialization	74
3.1.1.4	Higher-Layer Triggered Events	74
3.1.1.4.1	Constructing the Outgoing SIP REGISTER Request	74
3.1.1.5	Message Processing Events and Sequencing Rules	74
3.1.1.5.1	Processing the SIP REGISTER Response	74
3.1.1.6	Timer Events	76
3.1.1.7	Other Local Events	76
3.1.2	Server Role	76
3.1.2.1	Abstract Data Model	76
3.1.2.2	Timers	76
3.1.2.3	Initialization	77
3.1.2.4	Higher-Layer Triggered Events	77
3.1.2.5	Message Processing Events and Sequencing Rules	77
3.1.2.5.1	Processing the REGISTER Request	77
3.1.2.6	Timer Events	78
3.1.2.7	Other Local Events	79
3.2	Removing a Binding from the Registrar	79
3.2.1	Client Role	79
3.2.1.1	Abstract Data Model	79
3.2.1.2	Timers	79
3.2.1.3	Initialization	79
3.2.1.4	Higher-Layer Triggered Events	80
3.2.1.5	Message Processing Events and Sequencing Rules	80
3.2.2	Server Role	80
3.2.2.1	Abstract Data Model	80
3.2.2.2	Timers	80
3.2.2.3	Initialization	80
3.2.2.4	Higher-Layer Triggered Events	80
3.2.2.4.1	Constructing the Outgoing Deregister NOTIFY Request	80
3.2.2.5	Message Processing Events and Sequencing Rules	80
3.2.2.6	Timer Events	81
3.2.2.7	Other Local Events	81
3.3	Obtaining Provisioning Information	81
3.3.1	Client Role	81
3.3.1.1	Abstract Data Model	81
3.3.1.2	Timers	81
3.3.1.3	Initialization	81
3.3.1.4	Higher-Layer Triggered Events	81
3.3.1.5	Message Processing Events and Sequencing Rules	81
3.3.1.6	Timer Events	81
3.3.1.7	Other Local Events	82
3.3.2	Server Role	82
3.3.2.1	Abstract Data Model	82
3.3.2.2	Timers	82
3.3.2.3	Initialization	82
3.3.2.4	Higher-Layer Triggered Events	82
3.3.2.4.1	Processing the Incoming SUBSCRIBE Request	82
3.3.2.5	Message Processing Events and Sequencing Rules	82
3.3.2.6	Timer Events	82
3.3.2.7	Other Local Events	82

3.4	Automatically Updating Client to a Server Compatible Version	83
3.4.1	Client Role	83
3.4.1.1	Abstract Data Model	83
3.4.1.2	Timers	83
3.4.1.3	Initialization	83
3.4.1.4	Higher-Layer Triggered Events	83
3.4.1.5	Message Processing Events and Sequencing Rules	83
3.4.1.5.1	Construction of User-Agent header	83
3.4.1.5.2	Construction of Dynamic URL	83
3.4.1.5.3	Possible Values for lang Parameter	84
3.4.1.6	Timer Events	87
3.4.1.7	Other Local Events	87
3.4.2	Server Role	87
3.4.2.1	Abstract Data Model	87
3.4.2.2	Timers	87
3.4.2.3	Initialization	87
3.4.2.4	Higher-Layer Triggered Events	87
3.4.2.5	Message Processing Events and Sequencing Rules	87
3.4.2.5.1	Processing Incoming REGISTER Request	88
3.4.2.5.2	Sending 200 OK response (Action: Allow client)	88
3.4.2.5.3	Sending 403 Forbidden response (Action: Block client with prompt).....	88
3.4.2.5.4	Sending 403 Forbidden response (Action: Block client with static URL)...	89
3.4.2.5.5	Sending 200 OK response (Action: Allow clients with static URL)	89
3.4.2.5.6	Sending 403 Forbidden response (Action: Block clients with upgrade)....	90
3.4.2.5.7	Sending 200 OK response (Action: Allow clients with upgrade).....	90
3.4.2.5.8	Sending 403 Forbidden response (Action: Block client with dynamic URL)90	
3.4.2.5.9	Sending 200 OK response (Action: Allow client with dynamic URL)	91
3.4.2.6	Timer Events	91
3.4.2.7	Other Local Events	91
3.5	Notifying the Client of Survivable Mode	91
3.5.1	Server Role	91
3.5.1.1	Abstract Data Model	91
3.5.1.2	Timers	92
3.5.1.3	Initialization	92
3.5.1.4	Higher-Layer Triggered Events	92
3.5.1.4.1	Constructing the Survivable Mode NOTIFY Request	92
3.5.1.5	Message Processing Events and Sequencing Rules	92
3.5.1.6	Timer Events	92
3.5.1.7	Other Local Events	92
4	Protocol Examples	93
4.1	Registration	93
4.1.1	Basic Registration	93
4.1.2	Basic Unregistration	94
4.1.3	Deregistration	95
4.1.4	Survivable Mode Notify	95
4.1.5	Notify for Registrar Change	95
4.2	In-band Provisioning	96
4.2.1	Client to Server Request	96
4.2.2	Server to Client Response	97
4.2.3	Client-to-Server Delegated Provisioning Request	111
4.3	Automatically Updating Client to a Server-Compatible Version	112
4.3.1	Processing Incoming Register Request	112
4.3.2	Sending 200 OK Response (Action: Allow client)	112
4.3.3	Sending 403 Forbidden Response (Action: Block Client with Prompt).....	113
4.3.4	Sending 403 Forbidden Response (Action: Block Client with Static URL)	113
4.3.5	Sending 403 Forbidden Response (Action: Block Client with Dynamic URL)	113
4.3.6	Sending 200 OK Response (Action: Allow Client with Dynamic URL)	114

4.3.7	Sending 200 OK Response (Action: Allow Clients with Upgrade)	114
4.3.8	Sending 403 Forbidden (Action: Block Clients with Upgrade).....	115
4.3.9	Sending 200 OK Response (Action: Allow Client with Static URL)	115
5	Security	117
5.1	Security Considerations for Implementers	117
5.2	Index of Security Parameters	117
6	Appendix A: Product Behavior	118
7	Change Tracking	128
8	Index	130

1 Introduction

The Session Initiation Protocol (SIP) Registration Extensions protocol specifies proprietary extensions to Session Initiation Protocol (SIP) registration procedures. It also defines a provisioning protocol to enable SIP clients to obtain server provisioning data from SIP servers compliant to this specification. It is expected that the provisioning protocol sequence is performed during the client bootstrap process and that the data obtained is used for subsequent protocol operations attempted on the network.

Sections 1.8, 2, and 3 of this specification are normative and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in [\[RFC2119\]](#). Sections 1.5 and 1.9 are also normative but do not contain those terms. All other sections and examples in this specification are informative.

1.1 Glossary

The following terms are specific to this document:

200 OK: A response to indicate that the request has succeeded.

403 Forbidden: A response that indicates that a protocol server understood but denies a request.

access control list (ACL): A list of access control entries (ACEs) that collectively describe the security rules for authorizing access to some resource; for example, an object or set of objects.

address book: A collection of Address Book objects, each of which are contained in any number of address lists.

address book file: A file that contains a set of address book contact records.

address-of-record: A **Session Initiation Protocol (SIP) URI** that specifies a domain with a location service that can map the URI to another URI for a user, as described in [\[RFC3261\]](#).

Augmented Backus-Naur Form (ABNF): A modified version of Backus-Naur Form (BNF), commonly used by Internet specifications. ABNF notation balances compactness and simplicity with reasonable representational power. ABNF differs from standard BNF in its definitions and uses of naming rules, repetition, alternatives, order-independence, and value ranges. For more information, see [\[RFC5234\]](#).

authentication: The act of proving an identity to a server while providing key material that binds the identity to subsequent communications.

base URL: A URL that is specified for a web resource to convert all relative URLs in that resource to absolute URLs. A base URL ends with either a file name, such as `http://www.example.com/sample.htm`, or a slash, such as `http://www.example.com/subdir/`. See also absolute URL.

bot: A structured HTML comment that is processed by a front-end web server when the containing document is opened by or saved to the server. Also referred to as web bot.

Common Intermediate Format (CIF): A picture format, described in the H.263 standard, that is used to specify the horizontal and vertical resolutions of pixels in YCbCr sequences in video signals.

Content-Type header: A message header field whose value describes the type of data that is in the body of the message.

delegate: A user or resource that has permissions to act on behalf of another user or resource.

delegator: A user or resource for which another user or resource has permission to act on its behalf.

dialog: A peer-to-peer **Session Initiation Protocol (SIP)** relationship that exists between two **user agents** and persists for a period of time. A dialog is established by **SIP messages**, such as a 2xx response to an INVITE request, and is identified by a call identifier, a local tag, and a remote tag.

endpoint: A device that is connected to a computer network.

endpoint identifier (EPID): A unique identifier of a Session Initiation Protocol (SIP) **endpoint**. It is formed by combining the value of an epid parameter in a From or To header field with the **address-of-record** in the corresponding header field.

Focus Factory: A component that is responsible for creating, managing, and deleting conferences.

fully qualified domain name (FQDN): An unambiguous domain name (2) that gives an absolute location in the Domain Name System's (DNS) hierarchy tree, as defined in [\[RFC1035\]](#) section 3.1 and [\[RFC2181\]](#) section 11.

Globally Routable User Agent URI (GRUU): A **URI** that identifies a **user agent** and is globally routable. A URI possesses a GRUU property if it is useable by any **user agent client (UAC)** that is connected to the Internet, routable to a specific user agent instance, and long-lived.

header field: A component of a Session Initiation Protocol (SIP) message header, as described in [\[RFC3261\]](#).

Hypertext Transfer Protocol (HTTP): An application-level protocol for distributed, collaborative, hypermedia information systems (text, graphic images, sound, video, and other multimedia files) on the World Wide Web.

Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS): An extension of **HTTP** that securely encrypts and decrypts webpage requests.

in-band provisioning: A process in which a protocol client obtains configuration information from a protocol server.

Internet Protocol version 4 (IPv4): An Internet protocol that has 32-bit source and destination addresses. IPv4 is the predecessor of IPv6.

Internet Protocol version 6 (IPv6): A revised version of the Internet Protocol (IP) designed to address growth on the Internet. Improvements include a 128-bit IP address size, expanded routing capabilities, and support for **authentication** and privacy.

Kerberos: An **authentication** system that enables two parties to exchange private information across an otherwise open network by assigning a unique key (called a ticket) to each user that logs on to the network and then embedding these tickets into messages sent by the users. For more information, see [\[MS-KILE\]](#).

location profile: A definition of an environment where local numbers can be resolved to identifiers that either route to unique enterprise users or form unique numbers in a public telephone network, as defined by the International Telecommunications Union (ITU) recommendation.

meeting console: The abbreviated name for the Microsoft Office Communications Live Meeting Console software.

Messaging Application Programming Interface (MAPI): A messaging architecture that enables multiple applications to interact with multiple messaging systems across a variety of hardware platforms.

ms-diagnostics header: A header that is added to a **Session Initiation Protocol (SIP)** response, BYE request, or CANCEL request to convey troubleshooting information.

ms-diagnostics-public header: A header that is added to a **Session Initiation Protocol (SIP)** response, BYE request, or CANCEL request to convey troubleshooting information. Unlike the **ms-diagnostics header**, the ms-diagnostics-public header does not contain a "source" parameter.

Multipurpose Internet Mail Extensions (MIME): A set of extensions that redefines and expands support for various types of content in email messages, as described in [\[RFC2045\]](#), [\[RFC2046\]](#), and [\[RFC2047\]](#).

NOTIFY: A method that is used to notify a **Session Initiation Protocol (SIP)** client that an event requested by an earlier SUBSCRIBE method has occurred. The notification optionally provides details about the event.

NT LAN Manager (NTLM) Authentication Protocol: A protocol using a challenge-response mechanism for **authentication** in which clients are able to verify their identities without sending a password to the server. It consists of three messages, commonly referred to as Type 1 (negotiation), Type 2 (challenge) and Type 3 (authentication). For more information, see [\[MS-NLMP\]](#).

public IM connectivity: The ability of a protocol server deployment to interoperate with a public instant messaging (IM) provider.

public switched telephone network (PSTN): Public switched telephone network is the voice-oriented public switched telephone network. It is circuit-switched, as opposed to the packet-switched networks.

QoE Monitoring Server: A server that collects and processes Quality of Experience (QoE) metrics.

Real-Time Transport Protocol (RTP): A network transport protocol that provides end-to-end transport functions that are suitable for applications that transmit real-time data, such as audio and video, as described in [\[RFC3550\]](#).

REGISTER: A **Session Initiation Protocol (SIP)** method that is used by an SIP client to register the client address with an SIP server.

security association (SA): A simplex "connection" that provides security services to the traffic carried by it. See [\[RFC4301\]](#) for more information.

server: A replicating machine that sends replicated files to a partner (client). The term "server" refers to the machine acting in response to requests from partners that want to receive replicated files.

Session Initiation Protocol (SIP): An application-layer control (signaling) protocol for creating, modifying, and terminating sessions with one or more participants. **SIP** is defined in [\[RFC3261\]](#).

Simple Mail Transfer Protocol (SMTP): A member of the TCP/IP suite of protocols that is used to transport Internet messages, as described in [\[RFC5321\]](#).

SIP message: The data that is exchanged between **Session Initiation Protocol (SIP)** elements as part of the protocol. An SIP message is either a request or a response.

SIP protocol client: A network client that sends **Session Initiation Protocol (SIP)** requests and receives SIP responses. An SIP client does not necessarily interact directly with a human user. **User agent clients (UACs)** and proxies are SIP clients.

SIP registrar: A **Session Initiation Protocol (SIP)** server that accepts REGISTER requests and places the information that it receives from those requests into the location service for the domain that it handles.

SIP request: A **Session Initiation Protocol (SIP)** message that is sent from a **user agent client (UAC)** to a user agent server (UAS) to call a specific operation.

SIP response: A **Session Initiation Protocol (SIP)** message that is sent from a user agent server (UAS) to a **user agent client (UAC)** to indicate the status of a request from the UAC to the UAS.

site: A group of related pages and data within a SharePoint site collection. The structure and content of a site is based on a site definition. Also referred to as SharePoint site and web site.

SUBSCRIBE: A **Session Initiation Protocol (SIP)** method that is used to request asynchronous notification of an event or a set of events at a later time.

subscription: The result of a SUBSCRIBE request from a **Session Initiation Protocol (SIP)** element.

survivable mode: A mode that enables a protocol client to access basic voice services if some server or network resources are unavailable.

Transmission Control Protocol (TCP): A protocol used with the Internet Protocol (IP) to send data in the form of message units between computers over the Internet. TCP handles keeping track of the individual units of data (called packets) that a message is divided into for efficient routing through the Internet.

Transport Layer Security (TLS): A security protocol that supports confidentiality and integrity of messages in client and server applications communicating over open networks. **TLS** supports server and, optionally, client authentication by using X.509 certificates (as specified in [\[X509\]](#)). **TLS** is standardized in the IETF TLS working group. See [\[RFC4346\]](#).

Uniform Resource Identifier (URI): A string that identifies a resource. The URI is an addressing mechanism defined in Internet Engineering Task Force (IETF) Uniform Resource Identifier (URI): Generic Syntax [\[RFC3986\]](#).

Uniform Resource Locator (URL): A string of characters in a standardized format that identifies a document or resource on the World Wide Web. The format is as specified in [\[RFC1738\]](#).

user agent: An HTTP user agent, as specified in [\[RFC2616\]](#).

user agent client (UAC): A logical entity that creates a new request, and then uses the client transaction state machinery to send it. The role of **UAC** lasts only for the duration of that transaction. In other words, if a piece of software initiates a request, it acts as a **UAC** for the duration of that transaction. If it receives a request later, it assumes the role of a user agent server (UAS) for the processing of that transaction.

virtual directory: An HTTP URL that represents the root of a location where content can be published administratively.

Voice over IP (VoIP): The use of the Internet Protocol (IP) for transmitting voice communications. VoIP delivers digitized audio in packet form and can be used to transmit over intranets, extranets, and the Internet.

Windows Installer (.msi) file: A package file that contains the instructions and data required to install an application on a Windows-based computer. Every package contains at least one .msi file. The .msi file contains the installer database, a summary information stream, and possibly one or more transforms and internal source files.

XML document: A document object that is well formed, as described in [\[XML\]](#), and might be valid. An XML document has a logical structure that is composed of declarations, elements, comments, character references, and processing instructions. It also has a physical structure that is composed of entities, starting with the root, or document, entity.

XML schema definition (XSD): The World Wide Web Consortium (W3C) standard language that is used in defining XML schemas. Schemas are useful for enforcing structure and constraining the types of data that can be used validly within other XML documents. XML schema definition refers to the fully specified and currently recommended standard for use in authoring XML schemas.

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as defined in [\[RFC2119\]](#). All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information.

[IETF-DRAFT-OUGRUAUSIP-10] Rosenberg, J., "Obtaining and Using Globally Routable User Agent (UA) URIs (GRUU) in the Session Initiation Protocol (SIP)", draft-ietf-sip-gruu-10, July 2006, <http://tools.ietf.org/id/draft-ietf-sip-gruu-10.txt>

[MS-ABS] Microsoft Corporation, "[Address Book File Structure](#)".

[MS-AVEDGEA] Microsoft Corporation, "[Audio Video Edge Authentication Protocol](#)".

[MS-CONMGMT] Microsoft Corporation, "[Connection Management Protocol](#)".

[MS-DLX] Microsoft Corporation, "[Distribution List Expansion Protocol](#)".

[MS-E911WS] Microsoft Corporation, "[Web Service for E911 Support Protocol](#)".

[MS-OCER] Microsoft Corporation, "[Client Error Reporting Protocol](#)".

[MS-PRES] Microsoft Corporation, "[Presence Protocol](#)".

[MS-QoE] Microsoft Corporation, "[Quality of Experience Monitoring Server Protocol](#)".

[MS-RGSWS] Microsoft Corporation, "[Response Group Service Web Service Protocol](#)".

[MS-SIPAE] Microsoft Corporation, "[Session Initiation Protocol \(SIP\) Authentication Extensions](#)".

[MS-SIPRE] Microsoft Corporation, "[Session Initiation Protocol \(SIP\) Routing Extensions](#)".

[MS-SIP] Microsoft Corporation, "[Session Initiation Protocol Extensions](#)".

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.rfc-editor.org/rfc/rfc2119.txt>

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and Schooler, E., "SIP: Session Initiation Protocol", RFC 3261, June 2002, <http://www.ietf.org/rfc/rfc3261.txt>

[RFC3265] Roach, A. B., "Session Initiation Protocol (SIP)-Specific Event Notification", RFC 3265, June 2002, <http://www.ietf.org/rfc/rfc3265.txt>

[RFC950] Mogul, J., and Postel, J., "Internet Standard Subnetting Procedure", STD 5, RFC 950, August 1985, <http://www.rfc-editor.org/rfc/rfc950.txt>

1.2.2 Informative References

[RFC5234] Crocker, D., Ed., and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008, <http://www.rfc-editor.org/rfc/rfc5234.txt>

1.3 Overview

This protocol is an extension of the original **Session Initiation Protocol (SIP)**.

This protocol defines registration extensions to the SIP protocol. These extensions are described in detail in section [3.1](#).

This protocol also defines a protocol to retrieve provisioning and configuration information from a **server** compliant to this specification. This protocol is also referred to as **in-band provisioning**. These extensions are described in detail in section [3.3](#).

1.4 Relationship to Other Protocols

This protocol depends on SIP. This protocol is invoked as an extension of SIP. This protocol depends on all of the protocols on which the SIP specification depends.

In addition, this protocol depends on contacts and group **subscription** described in [\[MS-SIP\]](#) section [2.2.4](#), and self-subscription described in [\[MS-PRES\]](#) section 2.2.2.3.

1.5 Prerequisites/Preconditions

This protocol assumes that both the **SIP protocol clients** and the server support SIP. The prerequisites for this protocol are the same as the prerequisites described in [\[MS-SIP\]](#) section [1.5](#).

1.6 Applicability Statement

This protocol is applicable when both the SIP protocol clients and the server support SIP and depend upon the use of one or more of the enhancements offered by SIP extensions.

1.7 Versioning and Capability Negotiation

This protocol does not have protocol versioning. Instead, explicit capability negotiation is done as specified in this section by using the **Supported** header (section [2.2.1.5](#)) to indicate support of various features. Using the **Supported** header is the standard SIP mechanism of doing capability negotiation.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

A **user agent client (UAC)** that supports this protocol MUST use Session Initiation Protocol (SIP) over **Transmission Control Protocol (TCP)** or **Transport Layer Security (TLS)**, as specified in [\[RFC3261\]](#), for **Internet Protocol version 4 (IPv4)/Internet Protocol version 6 (IPv6)<1>** to exchange **SIP messages** with the **SIP registrar**.

2.2 Message Syntax

This protocol relies on the SIP message format, as specified in [\[RFC3261\]](#) section 7.

This protocol uses the **REGISTER** request and response, as specified in [\[RFC3261\]](#) section 10, for a UAC to register a binding. This protocol defines new option tags for the **Supported header field** and also adds new headers to be used with REGISTER requests and responses. This protocol defines a new **Event** header to be used with REGISTER requests and responses.

This protocol also uses the **NOTIFY** request specified in [\[RFC3265\]](#) section 4.4.5 to remove the binding, or deregister, from the SIP registrar. This protocol defines a new token to be used as the event-type in the **Event** header field and a new **Multipurpose Internet Mail Extensions (MIME)** type in the **Content-Type header** field, as specified in [\[RFC3261\]](#) section 20.15, in the NOTIFY request.

This protocol uses the **SUBSCRIBE** request and response specified in [\[RFC3265\]](#) section 4.4.3 to obtain in-band provisioning configuration information from the server.

2.2.1 Extensions to REGISTER Requests and Responses

2.2.1.1 SIP REGISTER Request Format

The SIP REGISTER request is constructed using the rules specified in [\[RFC3261\]](#) section 10 and the following extensions.

Zero or one **Event** header MAY be present in a REGISTER request. If an **Event** header is present, it MUST have the value "registration". This creates an implicit subscription for sending the registration state-change notifications specified in [\[RFC3265\]](#) section 7.2.1.

The UAC SHOULD add the option tags defined in section [2.2.1.5](#) to the **Supported** header field in the REGISTER request.

The UAC MAY add at most one **ms-keep-alive** header field, as specified in [\[MS-CONMGMT\]](#) section 2.2.1, in the REGISTER request.

The constructed REGISTER request MUST also conform to the rules specified in Section 3.4 of [\[MS-SIPRE\]](#).

2.2.1.2 SIP REGISTER Response Format

The SIP REGISTER response is constructed using the rules specified in [\[RFC3261\]](#) section 10 and the following extensions.

The REGISTER **200 OK** response MAY contain an **ms-keep-alive** header field, as specified in [\[MS-CONMGMT\]](#) section 2.2.1.

The REGISTER 200 response constructed by the SIP registrar SHOULD contain either:

- Zero **ms-keep-alive** header fields, as specified in [MS-CONMGMT] section 2.2.1.
- One **ms-keep-alive** header field, as specified in [MS-CONMGMT] section 2.2.1.

The registrar SHOULD add a **presence-state** header constructed using the rules specified in section [2.2.1.4](#).

The registrar MUST add a **Server** header using the rules specified in section [2.2.1.9](#).

The registrar MUST add a **Contact** header using the rules specified in [RFC3261] section 8.1.1.8.

2.2.1.3 ms-keep-alive Header Field Syntax

The syntax and the use of the **ms-keep-alive header** are specified in [\[MS-CONMGMT\]](#) section 2.2.1.

2.2.1.4 Presence-State Header Field Syntax

This protocol defines a new **Presence-State** header field.

The **Augmented Backus-Naur Form (ABNF)**, as defined in [\[RFC5234\]](#), syntax for the **Presence-State** header field [<2>](#) is as follows:

```
presence-state           = "presence-State" HCOLON presence-state-value
presence-state-value    = register-action[SEMI primary-cluster-state [SEMI user-services-
state-unavailable] ]
register-action          = "register-action" EQUAL register-action-value
register-action-value    = ""added"" / ""refreshed"" / ""fixed""
primary-cluster-state   = "primary-cluster-type" EQUAL primary-cluster-type-value SEMI
cluster-connection-type-state
primary-cluster-type-value = ""central"" / ""remote""
cluster-connection-type-state = "is-connected-to-primary" EQUAL yes-no-value
yes-no-value            = ""yes"" / ""no""
user-services-state-unavailable = "user-services-state=unavailable"
```

The ABNF syntax for the **Presence-State** header [<3>](#) field is as follows:

```
presence-state           = "presence-State" HCOLON presence-state-value
presence-state-value     = "register-action"
register-action-value    = ""added"" / ""refreshed"" / ""fixed""
```

The SIP registrar MAY add at most one **Presence-State** header field in the 200 response to the REGISTER request.

2.2.1.5 Supported Header Field Extensions

This protocol extends the **Supported** header field for the REGISTER request with the following new option tags.

adhoclist: The UAC SHOULD add this option tag in the **Supported** header field in the REGISTER request if the UAC supports batched subscribe requests, as specified in [\[MS-SIP\]](#) section [3.3.4.1](#).

com.microsoft.msrtc.presence: The UAC SHOULD add this option tag in the **Supported** header field in the REGISTER request if the UAC supports the Presence Protocol, as specified in [MS-SIP] section [3.2.4.1](#).

msrtc-event-categories: The UAC SHOULD add this option tag in the **Supported** header field in the REGISTER request if the UAC supports the Presence Protocol, as specified in [\[MS-PRES\]](#) section 2.2.2.4.1.

gruu-10: Used by the UAC for requesting the **Globally Routable User Agent URI (GRUU)** specified in [\[IETF DRAFT-OUGRUAUSIP-10\]](#) section 6. The UAC SHOULD add this option tag in the **Supported** header field in the REGISTER request.

ms-forking: This tag is deprecated and SHOULD NOT be sent in requests.

ms-userservices-state-notification: The UAC SHOULD [<4>](#) add this option tag in the **Supported** header field in the REGISTER request if the UAC supports **survivable mode** registration, which is registration without the availability of the service that handles the Presence Protocol specified in [\[MS-PRES\]](#) section 1.

ms-cluster-failover: The UAC SHOULD [<5>](#) add this option tag to the **Supported** header field in the REGISTER request if the UAC supports serial processing of contact headers in decreasing q-value order, as specified in [\[RFC3261\]](#) section 10.2.

The ABNF, as defined in [\[RFC5234\]](#), syntax listing specified in [\[RFC3261\]](#) section 25.1 is extended as follows:

```
Supported = ( "Supported" / "k" ) HCOLON [ "adhoclist" / "com.microsoft.msrtc.presence" /
"msrtc-event-categories" / "gruu-10" / "ms-userservices-state-notification" / "ms-cluster-
failover" / option-tag *(COMMA option-tag) ]
```

2.2.1.6 Ms-Subnet Header Field Syntax

This protocol defines a new **Ms-Subnet** header field. [<6>](#) The ABNF, as defined in [\[RFC5234\]](#), syntax for the **Ms-Subnet** header field is as follows:

```
Ms-Subnet = "ms-subnet" HCOLON ms-subnet-value
ms-subnet-value = IPv4address
```

The ABNF for **IPv4address** is specified in [\[RFC3261\]](#) section 25.1.

The UAC SHOULD add, at most, one **Ms-Subnet** header field in the REGISTER request to the SIP registrar. [<7>](#)

2.2.1.7 Ms-Device-Info Header Field Syntax

This protocol defines a new **Ms-Device-Info** header field [<8>](#). The ABNF syntax, as defined in [\[RFC5234\]](#), for the **Ms-Device-Info** header field is as follows:

```
Ms-Device-Info = "Ms-Device-Info" HCOLON ms-device-info-value
Ms-device-info-value = mac-info COMMA vendor-info COMMA version-info
mac-info = "MAC" EQUAL mac-addr
mac-addr = ^((([a-fA-F0-9]{2}-){5})([a-fA-F0-9]{2})$|^((([a-fA-F0-9]{2}-){7})([a-fA-F0-9]{2}))$)
vendor-info = "vendor" EQUAL pvalue
version-info = "version" EQUAL token
```

The ABNF for **pvalue** and **token** is specified in [\[RFC3261\]](#) section 25.1.

The UAC SHOULD add, at most, one **Ms-Device-Info** header field in the REGISTER request to the SIP registrar. [<9>](#)

2.2.1.8 P-Preferred-Registrar Header Field Syntax

This protocol defines a new **P-Preferred-Registrar** header field. <10> The ABNF syntax, as defined in [RFC5234], for the **P-Preferred-Registrar** header field is as follows:

```
p-preferred-registrar = "p-preferred-registrar" HCOLON preferred-registrar-value
preferred-registrar-value = addr-spec
```

The ABNF for **addr-spec** is specified in [RFC3261] section 25.1.

The SIP registrar SHOULD add, at most, one **P-Preferred-Registrar** header field in the NOTIFY to the UAC. <11>

2.2.1.9 Extensions to Server Header

The **Server** header is specified in [RFC3261] section 20.35. Implementations conformant with this specification SHOULD <12> set the value of the **Server** header to the literal string "RTC/4.0".

The ABNF, as defined in [RFC5234], for the **Server** header specified in [RFC3261] section 25.1 is restricted as follows:

```
Server = "Server" HCOLON "RTC/4.0" / "RTC/3.5" / "RTC/3.0"
```

2.2.1.10 Extensions to Contact Header

Implementations SHOULD NOT use a wildcard **Contact** in REGISTER requests for modifying all registrar bindings. This is a deviation from what is specified in [RFC3261] section 8.1.1.8, which allows wildcard **Contacts** with **Expires: 0** header to be used for removing all bindings.

2.2.1.11 Deregister NOTIFY Request Format

The deregister NOTIFY request does not define a new event package, in the context of what is specified in [RFC3265], or a new subscription event package. Instead, the following **Event** and **Content-Type** headers are defined to identify the deregister NOTIFY, in the context of what is specified in [RFC3265] section 7.1.2 and section 7.2.1.

2.2.1.11.1 subscription-state Header

The deregister NOTIFY MUST add exactly one **subscription-state** header field with **terminated** value and **expires** parameter value of zero.

2.2.1.11.2 Registration-Notify Event Header

The SIP registrar MUST send the deregister NOTIFY request with exactly one **Event** header field with "registration-notify" as the **event-type**.

This protocol defines this new **event-type** "registration-notify". This event type does not define a new event package, in the context of what is specified in [RFC3265] section 2. The presence of a **subscription-state** header field does not imply the use of any subscription event package. It is used in conjunction with the extensions defined in section 2.2.1.11.3 and section 2.2.1.11.4, primarily as the identifying features of a deregister NOTIFY.

2.2.1.11.3 Content-Type Header

The SIP registrar MUST send the deregister NOTIFY request with **text/registration-event** as the MIME type in the Content-Type header field, as specified in [\[RFC3261\]](#) section 20.15.

2.2.1.11.4 Ms-Diagnostics-Public Header

When generating this NOTIFY, the SIP registrar SHOULD add exactly one **ms-diagnostics-public header**, as specified in [\[MS-OCER\]](#) section 2.2.1.2. The **ErrorId** field values are given in section [2.2.1.13](#).

2.2.1.12 Survivable Mode NOTIFY Request Format

The survivable mode NOTIFY request<[13](#)> does not define a new event package, as specified in [\[RFC3265\]](#), or a new subscription event package. Instead the following **Event** and **Content-Type** headers are defined to identify the survivable mode NOTIFY, as specified in [\[RFC3265\]](#) section 7.1.2 and section 7.2.1.

2.2.1.12.1 subscription-state Header

The survivable mode NOTIFY MUST add exactly one **subscription-State** header field with an **active** value and an **expires** parameter value of the time remaining on the registration.

2.2.1.12.2 Registration-Notify Event Header

The SIP registrar MUST send the survivable mode NOTIFY request with exactly one **Event** header field with "registration-notify" as the **event-type**, as defined in section [2.2.1.11.2](#).

2.2.1.12.3 Content-Type Header

The SIP registrar MUST send the survivable mode NOTIFY request with "text/registration-event" as the MIME in the Content-Type header field. For more information, see [\[RFC3265\]](#) section 20.15.

2.2.1.12.4 Ms-Diagnostics-Public Header

When generating this NOTIFY, the SIP registrar SHOULD add exactly one ms-diagnostics-public header, as specified in [\[MS-OCER\]](#) section 2.2.1.2. The **ErrorId** field values are defined in section [2.2.1.13](#).

2.2.1.13 text/registration-event Message Body

The message body MUST be set to one of the following values, depending on the event that triggered the NOTIFY. White spaces, both leading and trailing, SHOULD be ignored by the UAC.

deregistered;event=unregistered: This is used by the SIP registrar if it unregisters an **endpoint** because of policy. For example, this is used when other endpoints of the same user have registered and the registrar decides to unregister the oldest endpoint. The **ms-diagnostics-public ErrorId** for this event is 4140. For more information, see [\[MS-OCER\]](#) section 7.

deregistered;event=rejected: This is used by the registrar if the user is disabled for SIP communication. For example, this can happen if the administrator decides to lock the user's account.

The **ms-diagnostics-public ErrorId** for this event is 4141. For more information, see [\[MS-OCER\]](#) section 7.

deregistered;event=deactivated: This is used by the registrar to indicate that the service is temporarily unavailable for this user. For example, the user could be in the process of being moved from one server to another server by the administrator.

The **ms-diagnostics-public ErrorId** for this event is 4142. For more information, see [MS-OCER] section 7.

deregistered;event=userservices-unavailable<14>: This is used by the registrar if it unregisters or rejects an endpoint when the presence service for the user is unavailable and the endpoint does not support survivable mode registrations.

The **ms-diagnostics-public ErrorId** for this event is 4165. For more information, see [MS-OCER] section 7.

deregistered;event=preferred-registrar-change<15>: This is used by the registrar to indicate that the registrar for the endpoint has changed. The **P-Preferred-Registrar** header, as defined in section [2.2.1.8](#), specifies the new registrar for the endpoint (5).

The **ms-diagnostics-public ErrorId** for this event is 4166 or 4167. For more information, see [MS-OCER] section 7.

registered;event=userservices-unavailable<16>: Used by the registrar to indicate that the UAC is in survivable mode. UACs SHOULD indicate support for survivable mode registration via the **ms-userservices-state-notification Supported** header described in section [2.2.1.5](#).

The **ms-diagnostics-public ErrorId** for this event is 4165.

2.2.2 In-band Provisioning Messages

A UAC requests the provisioning configuration that it is interested in by sending an in-band provisioning request. The server responds with the provisioning data for each of the groups listed in the request.

2.2.2.1 In-band Provisioning Request

The SUBSCRIBE request is constructed according to the procedures specified in [\[RFC3265\]](#) section 4.4.3. To request in-band provisioning information, the following additional rules SHOULD be followed.

- The **To-Uri** of the request is set to the user's SIP **address-of-record**.
- The **From-Uri** is the same as the **To-Uri** if the user does a self-request for provisioning [<17>](#).
- The **From-Uri** is the **delegate's URI** if the request is done by a delegate on behalf of the **delegator**. If the **From-Uri** is a delegate's URI, there MUST be a **p-session-on-behalf-of** header that equals the delegator's URI. The **To-Uri** is set as a delegator's URI. The authorization logic when delegation is used is described in [\[MS-SIPAE\].<18>](#)
- When the request is made by a server entity that does not have a SIP address-of-record, the server MUST set the host portion of the **To-Uri** and the **From-Uri** to its own **fully qualified domain name (FQDN)** and MUST NOT include the user portion for both. The server entity initiating this request SHOULD be trusted to send this request via some out-of-band mechanism. [<19>](#)
- The **Event** header is set to "vnd-microsoft-provisioning-v2" to identify this as an in-band provisioning request.

- The **Accept** header is set to "application/vnd-microsoft-roaming-provisioning-v2+xml". Implementations SHOULD reject a request with a 406 response if the **Accept** header is not set to this value.
- The **Content-Type** header is set to "application/vnd-microsoft-roaming-provisioning-v2+xml".
- An **Expires** header is added with a value of "0" because this subscription does not establish a dialog.
- No **Require** headers are present in the request.
- Optional **Supported** headers MAY be present in the request, as specified in [\[MS-SIP\]](#) section 3.1.4.2 and [\[MS-PRES\]](#) section 2.2.2.4.1 for SUBSCRIBE requests.
- The body is a valid **application/vnd-microsoft-roaming-provisioning-v2+xml** document.

For a detailed request example, see section [4.2.1](#).

2.2.2.2 In-band Provisioning Response

The 200 SUBSCRIBE response is used to send the in-band provisioning configuration body if the UAC indicates support for the **ms-piggyback-first-notify** extension, as specified in [\[MS-SIP\]](#) section 3.4.4.1. If the UAC does not indicate support for this extension, the in-band provisioning response is sent using a NOTIFY request. In either case, the 200 SUBSCRIBE or NOTIFY response SHOULD be constructed according to the procedures specified in [\[RFC3265\]](#) section 3.2.2, with the following extensions:

- The **Event** header is set to "vnd-microsoft-provisioning-v2".
- The **Content-Type** header is set to "application/vnd-microsoft-roaming-provisioning-v2+xml".
- This subscription does not establish a dialog, so the **Expires** header is added with a value of zero.
- The body is a valid **application/vnd-microsoft-roaming-provisioning-v2+xml** document, as described in section [2.2.2.4](#).

A detailed response example is given in section [4.2.2](#).

2.2.2.3 Data Model for application/vnd-microsoft-roaming-provisioning-v2+xml Documents

This is an **XML document**, and unless otherwise specified, all elements have a cardinality of 1 and are required.

2.2.2.4 Data Model for Requests

```
provisioningGroupList
|
|-- provisioningGroup
```

The following **XML schema definition (XSD)** [<20>](#) fragment defines the requirements to which a **provisioningGroupList** XML document SHOULD conform.

```
<xs:complexType name="ProvisioningGroupListType">
  <xs:sequence>
    <xs:element name="provisioningGroup" type="ProvisioningGroupType"
maxOccurs="unbounded"/>
```

```

        <xs:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="subnet" type="IPAddress" use="optional"/>
    <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>
<xs:complexType name="ProvisioningGroupType">
    <xs:sequence>
        <xs:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required"/>
    <xs:attribute name="opaque" type="xs:string" use="optional"/>
    <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>
<xs:element name="provisioningGroupList" type="ProvisioningGroupListType"/>

```

provisioningGroupList: A list of available **provisioningGroup** elements.

provisioningGroup: A set of configuration data requested by the sending entity. The **name** attribute indicates the type of configuration data being requested. The following valid values are defined for the **name** attribute:

- **ServerConfiguration:** Global Server Configuration and Provisioning Data.
- **meetingPolicy:** Global Conferencing Policy Data, which is used for multi-party conferencing.
- **ucPolicy:** Global Unified Communications Policy Data.
- **publicationGrammar:** Grammar describing the publication of presence information, including presence containers and membership information based on rules defined by server administrators.
- **userSetting:** User-specific configuration data.
- **endpointConfiguration:** Client endpoint-specific configuration data. It controls the behavior of specific features of the client. [<21>](#)
- **locationPolicy:** Location related configuration applied to users. [<22>](#)
- **mediaConfiguration:** Media-related configuration data. [<23>](#)
- **presencePolicyV2:** Presence-related configuration data. [<24>](#)
- **privacyPublicationGrammar:** Privacy-related grammar describing presence containers and membership information based on rules defined by server administrators. [<25>](#)
- **publicProviders:** Data on currently supported public providers. [<26>](#)

The following XSD schema [<27>](#) fragment defines the requirements to which a **provisioningGroupList** XML document SHOULD conform.

```

<xs:complexType name="ProvisioningGroupListType">
    <xs:sequence>
        <xs:element name="provisioningGroup" type="ProvisioningGroupType"
maxOccurs="unbounded" ms:propertyName="ProvisioningGroups"/>
        <xs:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="subnet" type="IPAddress" use="optional"
ms:propertyName="Subnet"/>
    <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>

```

```

    <xs:complexType name="ProvisioningGroupType"
ms:className="ProvisioningGroupXmlDocumentResult">
    <xs:sequence>
        <xs:any namespace="##other" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="name" type="xs:string" use="required" ms:propertyName="Name"/>
    <xs:attribute name="opaque" type="xs:string" use="optional"
ms:propertyName="Opaque"/>
    <xs:anyAttribute namespace="##other" processContents="lax"/>
</xs:complexType>
<xs:element name="provisioningGroupList" type="ProvisioningGroupListType"
ms:className="CProvisioningGroupListXmlDocumentRoot"/>

```

2.2.2.5 Data Model for Responses

```

provisionGroupList
|
|-- provisionGroup

```

The XSD schema for a response **provisionGroupList** XML document is specified in section [2.2.2.4](#).

provisionGroupList: A list of available **provisionGroup** elements.

provisionGroup: A set of configuration data owned by a particular entity or service indicated by the **name** attribute. The following valid values are defined for the **name** attribute:

- **ServerConfiguration**: Global Server Configuration and Provisioning Data.
- **meetingPolicy**: Global Conferencing Policy Data, which is used for multi-party conferencing.
- **ucPolicy**: Global Unified Communications Policy Data.
- **publicationGrammar**: Grammar describing the publication of presence information, including presence containers and membership information based on rules defined by server administrators. [<28>](#)
- **userSetting**: User-specific configuration data. [<29>](#)
- **endpointConfiguration**: Client endpoint-specific configuration data. It controls the behavior of specific features of the client. [<30>](#)
- **locationPolicy**: Location-related configuration applied to users. [<31>](#)
- **mediaConfiguration**: Media-related configuration data. [<32>](#)
- **presencePolicyV2**: Presence-related configuration data. [<33>](#)
- **privacyPublicationGrammar**: Privacy-specific grammar describing presence containers and membership information based on rules defined by server administrators. [<34>](#)
- **publicProviders**: Data on currently supported public providers. [<35>](#)

The **provisionGroup** lists an arbitrary sequence of elements applicable to that entity. The data model is given in the following sections for all the groups.

2.2.2.5.1 Data Model for ServerConfiguration provisionGroup

Unless specified otherwise, all the properties are required and have a cardinality of 1. [<36>](#)

```

provisionGroup (name='ServerConfiguration')
|
|-- absWebServiceEnabled
|-- lisInternalUrl
|-- absInternalServerUrl
|-- absExternalServerUrl
|-- abwqInternalUrl
|-- abwqExternalUrl
|-- dlxInternalUrl
|-- dlxExternalUrl
|-- dlxEnabled
|-- updatesServerInternalUrl
|-- updatesServerExternalUrl
|-- updatesServerEnabled
|-- organization
|-- helpdeskInternalUrl
|-- helpdeskExternalUrl
|-- consoleDownloadInternalUrl
|-- consoleDownloadExternalUrl
|-- ucPortRangeEnabled
|-- ucMinMediaPort
|-- ucMaxMediaPort
|-- ucMinSipDynamicPort
|-- ucMaxSipDynamicPort
|-- ucMinAudioPort
|-- ucMaxAudioPort
|-- ucMinVideoPort
|-- ucMaxVideoPort
|-- ucMinAppSharingPort
|-- ucMaxAppSharingPort
|-- ucMinFileTransferPort
|-- ucMaxFileTransferPort
|-- ucPC2PCAVEncryption
|-- ucMaxVideoRateAllowed
|-- qosEnabled
|-- enableInCallQoS
|-- inCallQoSIntervalSeconds
|-- ucDiffServVoice
|-- ucVoice802_1p
|-- ucEnforcePinLock
|-- ucMinPinLength
|-- ucPhoneTimeOut
|-- ucExchangeMWIPoll
|-- ucEnableSIPSecurityMode
|-- ucEnableUserLogging
|-- loggingLevel
|-- enableBWPolicyCheck
|-- poolUri
|-- mrasUri
|-- qosUri
|-- callParkServerUri
|-- responseGroupServiceInternalURL
|-- responseGroupServiceExternalURL
|-- responseGroupServiceInternalAgentURL
|-- responseGroupServiceExternalAgentURL
|-- botSipUriForTestCall
|-- cwaInternalUri
|-- cwaExternalUri
|-- ucLocationProfile
|-- focusFactoryUri
|-- voiceMailUri

```

This provision group example follows the behavior described in endnote [<37>](#).

```

provisionGroup (name='ServerConfiguration')
|
|-- ucMaxVideoRateAllowed

```

```

|-- absInternalServerUrl
|-- absExternalServerUrl
|-- absWebServiceEnabled
|-- ucPC2PCAVEncryption
|-- organization
|-- consoleDownloadInternalUrl
|-- consoleDownloadExternalUrl
|-- helpdeskInternalUrl
|-- helpdeskExternalUrl
|-- dlxInternalUrl
|-- dlxExternalUrl
|-- dlxEnabled
|-- ucDiffServVoice
|-- ucVoice802_1p
|-- ucEnforcePinLock
|-- ucMinPinLength
|-- ucPhoneTimeOut
|-- ucExchangeMWIPoll
|-- ucEnableSIPSecurityMode
|-- ucEnableUserLogging
|-- updatesServerInternalUrl
|-- updatesServerExternalUrl
|-- updatesServerEnabled
|-- ucPortRangeEnabled
|-- ucMinMediaPort
|-- ucMaxMediaPort
|-- ucMinSipDynamicPort
|-- ucMaxSipDynamicPort
|-- qosEnabled
|-- enableInCallQoS
|-- inCallQoSIntervalSeconds
|-- ucLocationProfile
|-- mrasUri
|-- qosUri
|-- callControlServerUri
|-- poolUri
|-- cwaInternalUri
|-- cwaExternalUri
|-- focusFactoryUri
|-- voiceMailUri

```

The following XSD schema fragment defines the requirements to which a **ServerConfiguration provisionGroup** element XML document **MUST** conform.

```

<?xml version="1.0" encoding="utf-16"?>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2006/09/sip/provisioningrouplist-
notification" targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisioningrouplist-
notification" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="provisionGroup" type="tns:provisionGroupType" />
  <xs:complexType name="provisionGroupType">
    <xs:all minOccurs="0" maxOccurs="1">
<xs:element minOccurs="0" maxOccurs="1" name="absWebServiceEnabled" type="tns:propertyType"
/>
<xs:element minOccurs="0" maxOccurs="1" name="lisInternalUrl" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="absInternalServerUrl" type="tns:propertyType"
/>
<xs:element minOccurs="0" maxOccurs="1" name="absExternalServerUrl" type="tns:propertyType"
/>
<xs:element minOccurs="0" maxOccurs="1" name="abwqInternalUrl" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="abwqExternalUrl" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="dlxInternalUrl" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="dlxExternalUrl" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="dlxEnabled" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="updatesServerInternalUrl"
type="tns:propertyType" />

```



```

<xs:element minOccurs="0" maxOccurs="1" name="updatesServerExternalUrl"
type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="updatesServerEnabled" type="tns:propertyType"
/>
<xs:element minOccurs="0" maxOccurs="1" name="ucPortRangeEnabled" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="organization" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="helpdeskInternalUrl" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="helpdeskExternalUrl" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="consoleDownloadInternalUrl"
type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="consoleDownloadExternalUrl"
type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMinMediaPort" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMaxMediaPort" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMinSipDynamicPort" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMaxSipDynamicPort" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMinAudioPort" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMaxAudioPort" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMinVideoPort" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMaxVideoPort" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMinAppSharingPort" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMaxAppSharingPort" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMinFileTransferPort" type="tns:propertyType"
/>
<xs:element minOccurs="0" maxOccurs="1" name="ucMaxFileTransferPort" type="tns:propertyType"
/>
<xs:element minOccurs="0" maxOccurs="1" name="ucPC2PCAVEncryption" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMaxVideoRateAllowed" type="tns:propertyType"
/>
<xs:element minOccurs="0" maxOccurs="1" name="qosEnabled" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="EnableInCallQoS" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="InCallQoSIntervalSeconds"
type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucDiffServVoice" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucVoice802_1p" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucEnforcePinLock" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucMinPinLength" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucPhoneTimeOut" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucExchangeMWIPoll" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucEnableSIPSecurityMode"
type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucEnableUserLogging" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="loggingLevel" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="enableBWPolicyCheck" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="poolUri" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="mrasUri" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="qosUri" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="callControlServerUri" type="tns:propertyType"
/>
<xs:element minOccurs="0" maxOccurs="1" name="callParkServerUri" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="responseGroupServiceInternalURL"
type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="responseGroupServiceExternalURL"
type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="responseGroupServiceInternalAgentURL"
type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="responseGroupServiceExternalAgentURL"
type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="botSipUriForTestCall" type="tns:propertyType"
/>
<xs:element minOccurs="0" maxOccurs="1" name="cwaInternalUri" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="cwaExternalUri" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="requireFIPSCompliantMedia"
type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="hybridConfigServiceInternalUrl"
type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="hybridConfigServiceExternalUrl"
type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="ucLocationProfile" type="tns:propertyType" />

```

```

<xs:element minOccurs="0" maxOccurs="1" name="focusFactoryUri" type="tns:propertyType" />
<xs:element minOccurs="0" maxOccurs="1" name="voiceMailUri" type="tns:propertyType" />

    </xs:all>
    <xs:attribute fixed="ServerConfiguration" name="name" use="required" />
</xs:complexType>
</xs:schema>

```

ucMaxVideoRateAllowed: Maximum video rate that the UAC endpoints are to use. The maximum video rate MUST be one of the following defined values:

- **default:** Maximum resolution of VGA-600K.
- **HD720P-1.5M:** Maximum resolution of HD720P. High definition, with a resolution of 1280x720.
- **VGA-600K:** Maximum resolution of VGA, 640x480, 25 fps.
- **CIF-250K<38>:** Maximum resolution of **Common Intermediate Format (CIF)**. CIF has a resolution of 352x288, 15 fps.

absInternalServerUri: The URI where the Address Book Service can be reached within the enterprise. This SHOULD be a **Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)** URI pointing to the **virtual directory** in which the Address Book Service places the **address book files**, as specified in [\[MS-ABS\]](#) section 2.2.

absExternalServerUri: The URI where the Address Book Service can be reached through the public internet. This SHOULD be an HTTPS URI pointing to the virtual directory where the Address Book Service places the Address Book files, as specified in [\[MS-ABS\]](#) section 2.2.

ucPC2PCAVEncryption: Specifies whether encryption is required for Computer to Computer Audio-Video Sessions. **ucPC2PCAVEncryption** MUST be one of the following defined values:

- **SupportEncryption:** Encryption is optional.
- **RequireEncryption:** Encryption is required.
- **DoNotSupportEncryption:** Encryption is not required.

organization: The name of the organization.

consoleDownloadInternalUri<39>: The URI from which the **meeting console** can be downloaded within the enterprise. This SHOULD be an HTTP URI. For example:

```
"http://r.office.fabrikam.com/r/rlidOCS?clid=1033&pl=livemeeting"
```

consoleDownloadExternalUri<40>: The URI from which the **meeting console** can be downloaded through the public internet. This SHOULD be an HTTP URI. For example:

```
"http://r.office.fabrikam.com/r/rlidOCS?clid=1033&pl=livemeeting"
```

helpdeskInternalUri<41>: The URI for the organization's help desk within the enterprise. This SHOULD be an HTTP URI.

helpdeskExternalUri<42>: The URI for the organization's help desk through the public internet. This SHOULD be an HTTP URI.

dlxInternalUrl: The URI for the Group Expansion Web Service component within the enterprise. This SHOULD be an HTTPS URI pointing to the Group Expansion Web Service, as specified in [\[MS-DLX\]](#) section 2.

dlxExternalUrl: The URI for the Group Expansion Web Service component through the public internet. This SHOULD be an HTTPS URI pointing to the Group Expansion Web Service, as specified in [\[MS-DLX\]](#) section 2.

dlxEnabled: Indicates whether the Group Expansion Web Service is available. The value MUST be "true" or "false".

ucDiffServVoice: Ignored.

ucVoice802_1p: Ignored.

ucEnforcePinLock: Ignored.

ucMinPinLength: Ignored.

ucPhoneTimeOut: Ignored.

ucEnableUserLogging: [<43>](#) Indicates whether logging needs to be enabled on the user computer to record client behavior. The value SHOULD be "true" or "false".

ucExchangeMWIPoll: Ignored.

ucEnableSIPSecurityMode: The security mode for the client. The value MUST be "High", "Medium", or "Low".

updatesServerInternalUrl: Ignored.

updatesServerExternalUrl: Ignored.

updatesServerEnabled: Ignored.

ucPortRangeEnabled: [<44>](#) Indicates whether the port ranges sent by the server need to be used by the client for media and signaling. The value SHOULD be "true" or "false".

ucMinMediaPort: [<45>](#) The minimum of the range of dynamically-allocated ports that the client can use to transmit audio and video data using **Real-Time Transport Protocol (RTP)**.

ucMaxMediaPort: [<46>](#) The maximum of the range of dynamically-allocated ports that the client can use to transmit audio and video data using RTP.

ucMinSipDynamicPort: [<47>](#) The minimum of the range of dynamically-allocated ports that the client can use to transmit signaling data using SIP.

ucMaxSipDynamicPort: [<48>](#) The maximum of the range of dynamically-allocated ports that the client can use to transmit signaling data using SIP.

qosEnabled: [<49>](#) Optional. Ignored.

ucLocationProfile: The **location profile** for voice call features. This is used by the extensions described in [\[MS-SIPRE\]](#) section 2.2.8.

mrasUri: The URI of the Media Relay Authentication Service (MRAS), as specified in [\[MS-AVEDGEA\]](#) section 2.2.3.1.2. This MUST be a SIP URI that can be used to address the MRAS Service.

qosUri: The URI of the Media **QoE Monitoring Server**, as described in [\[MS-QoE\]](#) section 1.3. This MUST be a SIP URI that can be used to address the QoE Service.

callControlServerUri: [<50>](#) Ignored.

poolUri: [<51>](#) Ignored.

cwaInternalUri: [<52>](#) The URI for the organization's phone conferencing dial-in information page, accessible internally. This SHOULD be an HTTP(S) URI.

cwaExternalUri: [<53>](#) The URI for the organization's phone conferencing dial-in information page, accessible externally. This SHOULD be an HTTP(S) URI.

focusFactoryUri: The User's Conferencing **Focus Factory**. This is required for creating conferences. This MUST be a SIP URI.

voiceMailUri: The user's voice mail URI. This is required for sending messages to voice mail. This MUST be a SIP URI that can be used to route to the user's voice mail. The syntax is described in [MS-SIPRE] section 2.2.3.

responseGroupServiceInternalURL: [<54>](#) The **Uniform Resource Locator (URL)** for the Response Group Service Web Service within the enterprise. This MUST be an HTTPS URL pointing to the Response Group Service Web Service, as specified in [MS-RGSWS].

responseGroupServiceExternalURL: [<55>](#) The URL for the Response Group Service Web Service through the public internet. This MUST be an HTTPS URL pointing to the Response Group Service Web Service, as specified in [MS-RGSWS].

responseGroupServiceInternalAgentURL: [<56>](#) The URL of a webpage containing information about the Response Group Service within the enterprise. This SHOULD be an HTTPS URL.

responseGroupServiceExternalAgentURL: [<57>](#) The URL of a webpage containing information about the Response Group Service through the public internet. This SHOULD be an HTTPS URL.

absWebServiceEnabled: Ignored.

abwqExternalURL: [<58>](#) The Address Book Web Query URL. This is accessible through the public internet. This SHOULD be an HTTPS URI pointing to the Group Expansion Web Service, as specified in [MS-DLX] section 2.

abwqInternalURL: [<59>](#) The Address Book Web Query URL. This is accessible inside the enterprise/intranet. This SHOULD be an HTTPS URI pointing to the Group Expansion Web Service, as specified in [MS-DLX] section 2.

botSipUriForTestCall: [<60>](#) The SIP URI of the test **bot** that is available to the user for a test call.

callParkServerUri: [<61>](#) The URI for the call park server that clients can connect to.

enableBWPolicyCheck: [<62>](#) Boolean value that indicates whether the system is provisioned for bandwidth management.

lisInternalUri: [<63>](#) The URL of the Location Information Service, as specified in [MS-E911WS] section 3.1, which is accessible from inside the enterprise/intranet.

loggingLevel: [<64>](#) The logging level on the client.

ucMinAppSharingPort: [<65>](#) Minimum port used by Application Sharing.

ucMaxAppSharingPort: [<66>](#) Maximum port used by Application Sharing.

ucMinAudioPort: [<67>](#) Minimum port used by Audio.

ucMaxAudioPort: [<68>](#) Maximum port used by Audio.

ucMinFileTransferPort: [<69>](#) Minimum port used for File Transfer.

ucMaxFileTransferPort: [<70>](#) Maximum port used for File Transfer. This is for P2P File Transfers only.

ucMinVideoPort: [<71>](#) Minimum port used by Video.

ucMaxVideoPort: [<72>](#) Maximum port used by Video.

requireFIPSCompliantMedia: [<73>](#) Boolean value that requires the client to be FIPS compliant in all audio video conversations if present, and has the value of "true".

hybridConfigServiceInternalUrl: [<74>](#) The internal URL of a web service used to enable media bypass, E911, call park, and call center services for hybrid voice users.

hybridConfigServiceExternalUrl: [<75>](#) The external URL of a web service used to enable media bypass, E911, call park, and call center services for hybrid voice users.

enableInCallQos: Boolean value. True implies that the client will send in-call QoS metrics, otherwise false.

inCallQoSIntervalSeconds: Then integer value in second that specifies minimum interval between 2 in-call QoS will be sent

translationClientId:

translationClientSecret:

translationType:

translationServiceUri:

translationAccessTokenUri:

translationApplicationId:

For a detailed example, see section [4.2.2](#).

2.2.2.5.2 Data Model for meetingPolicy provisionGroup

```
provisionGroup (name='meetingPolicy')
|
|-- instance
|   |-- property
|   |   |-- name
```

The following XSD schema fragment defines the requirements a **meetingPolicy provisionGroup** element XML document MUST conform to.

```
<?xml version="1.0" encoding="utf-16"?>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification" targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="provisionGroup" type="tns:provisionGroupType" />
  <xs:complexType name="provisionGroupType">
    <xs:all minOccurs="1" maxOccurs="1">
```

```

        <xs:element minOccurs="1" maxOccurs="1" name="instance" type="tns:instanceType" />
    </xs:all>
    <xs:attribute fixed="meetingPolicy" name="name" use="required" />
</xs:complexType>
<xs:complexType name="instanceType">
    <xs:sequence>
        <xs:element minOccurs="1" maxOccurs="unbounded" name="property" type="tns:propertyType"
    />
    </xs:sequence>
</xs:complexType>
<xs:complexType name="propertyType">
    <xs:simpleContent>
        <xs:extension base="xs:string" >
            <xs:attribute name="name" use="required" />
        </xs:extension>
    </xs:simpleContent>
</xs:complexType>

</xs:schema>

```

instance: A single instance is specified within the **meetingPolicy provisionGroup**, listing a set of properties applicable for the user requesting the configuration.

Property: A single policy setting. The **name** attribute specifies the property name. Values are Boolean, unless otherwise specified.

AllowIPAudio: Specifies whether IP Audio functionality is enabled on the client. For example:

```
<property name="AllowIPAudio">true</property>
```

AllowIPVideo: Specifies whether IP Video functionality is enabled on the client. For example:

```
<property name="AllowIPVideo">true</property>
```

EnableDataCollaboration: Specifies whether Data Collaboration functionality is enabled on the client. For example:

```
<property name=" EnableDataCollaboration "> true </property>
```

Name: [<76>](#) Optional, ignored.

ColorDepth: [<77>](#) Optional, ignored.

AllowACP: [<78>](#) Optional, ignored.

RetainPPTForExternalMeeting: [<79>](#) Retain PowerPoint slides for external meeting.

MeetingSize: [<80>](#) Maximum number of attendees allowed in meetings.

AllowAppSharingForExternalMeeting: [<81>](#) Allow application sharing in external meetings.

AllowPresenterToRecord: [<82>](#) Allow meeting presenter to record meeting.

EnableAppDesktopSharing: [<83>](#) Specifies whether the client is allowed to share the desktop. For example:

```
<property name=" EnableAppDesktopSharing"> true </property>
```

EnablePSTNConferencing:[<84>](#) Enable conferencing audio call over a **public switched telephone network (PSTN)** network.

TrustedConferencingPinRequired:[<85>](#) Require that trusted conferencing PIN be specified to join the conference.

AllowAnnotations: Boolean value that allows participants to make annotations on top of PowerPoint content, as well as all whiteboarding content.

AllowAnonymousParticipants: Boolean value that allows anonymous participants to join the meeting.

AllowExternalUserControl: Boolean value that allows anonymous and federated users to control an application sharing session.

AllowExternalUserRecording: Boolean value that allows external users to record meetings organized by this user. It works if **AllowRecording** is "true".

AllowExternalUsersToSaveContent: Boolean value that allows anonymous and federated users to save content shared in a data sharing session.

AllowFileTransfer: Boolean value that allows multi-party file transfer in a meeting.

AllowParticipantControl: Boolean value that allows or prevents the user from sharing control with anyone when they are screen sharing. This applies regardless of who started/organized the conference.

AllowRecording: Boolean value that allows recording on meetings organized by this user. If "true", the **AllowExternalUserRecording** policy applies to external users.

EnableP2PFileTransfer: Enable P2P file transfer between two users during a meeting.

MaxConferenceVideoResolution: Set the maximum resolution for video conferencing. Allowed values are "CIF" or "VGA".

AllowedAppDesktopSharingLevel: Enables application and desktop sharing during web conferencing for users having this policy.

AllowUserToScheduleMeetingsWithAppSharing: Enables users to schedule meetings with Application sharing. Allowed values are "None", "SingleApplication", or "Desktop".

AppSharingBitRate: Set the bit rate for Application sharing.

AudioBitRate: Set the bit rate for Audio.

FileTransferBitRate: Set the bit rate for File Transfer.

RequirePasscode:[<86>](#) Controls whether a pass code is required to join dial-in conferences.

VideoBitRate:[<87>](#) Set the bit rate for Video.

AllowPolls:[<88>](#) Enables the polling feature within a meeting. The value MUST be "true" or "false".

EnableP2PRecording:[<89>](#) Enables recording of P2P conversations. The value MUST be "true" or "false".

EnableP2PVideo:[<90>](#) Enables P2P video conversations. The value MUST be "true" or "false". For a detailed example, see section [4.2.2](#).

EnableMultiViewJoin:[<91>](#) Enables multiple video channels in conferencing. The value MUST be "true" or "false". For a detailed example, see section [4.2.2](#).

TotalReceiveVideoBitRate:[<92>](#) Sets the total video bit rate to be used for conferencing video. For a detailed example, see section [4.2.2](#).

DisablePowerPointAnnotations:[<93>](#) Specifies whether the user has turned off PowerPoint annotations.

AllowSharedNotes:[<94>](#) Enables users to take shared notes during a meeting. The value MUST be "true" or "false". For a detailed example, see section [4.2.2](#).

AllowQna:[<95>](#) Enables the Question and Answer feature within a meeting. The value MUST be "true" or "false". This setting applies to the user who organizes the conference: if set to False, no conference created by a user affected by this policy will allow the use of the Questions and Answers Manager. However, the user can make use of the Questions and Answers Manager in other conferences where polls are allowed. The default value is true

AllowOfficeContent:[<96>](#) Enables the Office content (such as slide presentations) sharing feature within a meeting. The value MUST be "true" or "false". When set to False, prevents users from using Office content in their conferences.

EnableOnlineMeetingPromptForLyncResources: Boolean value. When set to True, users will be prompted any time they schedule a meeting in Outlook that includes invitees (such as a meeting room) that would benefit from having the meeting held online. The default value is false.

2.2.2.5.3 Data Model for ucPolicy provisionGroup

```
provisionGroup (name='ucPolicy')
|
|-- instance
|  |-- property
|  |-- element
```

The following XSD schema fragment defines the requirements to which a **ucPolicy provisionGroup** element XML document MUST conform.

```
<?xml version="1.0" encoding="utf-16"?>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification" targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="provisionGroup" type="tns:provisionGroupType" />
  <xs:complexType name="provisionGroupType">
    <xs:all minOccurs="1" maxOccurs="1">
      <xs:element minOccurs="1" maxOccurs="1" name="instance" type="tns:instanceType" />
    </xs:all>
    <xs:attribute fixed="ucPolicy" name="name" use="required" />
  </xs:complexType>
  <xs:complexType name="instanceType">
    <xs:sequence>
      <xs:element minOccurs="1" maxOccurs="unbounded" name="property" type="tns:propertyType"
/>
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="propertyType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="name" use="required" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
</xs:schema>
```



```
</xs:simpleContent>
</xs:complexType>
</xs:schema>
```

instance: A single instance is specified within the **ucPolicy provisionGroup**, listing a set of properties applicable for the user requesting the configuration.

property: Lists a single policy setting. The **name** attribute specifies the property name. Values are Boolean, specified in CDATA notation, unless otherwise specified. Some properties list a sequence of **element** sub-elements.

AllowSimultaneousRinging: Whether simultaneous ringing, as described in [\[MS-SIPRE\]](#) section 3.9.5.1.2, support is enabled. For example:

```
<property name=" AllowSimultaneousRinging ">true</property>
```

Name: Optional, ignored.

PhoneRouteUsages: [<97>](#) Optional, ignored.

AllowCallForwarding: Turns off call forwarding to a phone number or contact. The user only has the option to forward to voice mail.

enableBWPolicyOverride: [<98>](#) Boolean value that enables the user endpoint to bypass the BandWidth (BW) policy check, and only report BW utilization for an established session.

EnableCallPark: Boolean value that allows specified clients and devices to park calls.

EnableCallTransfer: Boolean value that enables the call transfer user interface on client.

EnableDelegation: Boolean value that enables all client user interface entry points for call delegation.

EnableMaliciousCallTrace: Boolean value that enables the malicious call trace functionality.

EnableTeamCall: Boolean value that enables all client user interface entry points for the team call-group.

UcEnabled: [<99>](#) Ignored.

VoiceDeploymentMode: [<100>](#) String value that indicates to the client the type of public switched telephone network (PSTN)connectivity they are provisioned for. The value MUST be one of the following

- **OnPrem**
- **Online**
- **OnlineBasic**
- **OnPremOnlineHybrid**

A detailed example is given in section [4.2.2](#).

2.2.2.5.4 Data Model for publicationGrammar provisionGroup

```
provisionGroup (name='publicationGrammar')
|
|-- containerManifestList
```

```

|   |- minSupportedClientVersion
|   |- majorVersion
|   |- minorVersion
|   |-- containerManifest
|   |   |-- containers
|   |       |-- container
|   |           |- id
|   |           |-- visible
|   |           |-- avoidRedundantMembers
|   |           |-- blocked
|   |       |-- members
|   |           |-- member
|   |               |- type
|   |               |- role
|   |               |- uri
|   |               |-- allowedContainers
|   |                   |- type
|   |                   |-- container
|   |                       |- id
|   |               |-- defaultContainer
|   |                   |- id
|   |               |-- occurrenceConstraint
|   |                   |- type
|   |               |-- sourceNetworks
|   |                   |-- sourceNetwork
|   |                       |- type
|   |               |-- resolutionRules
|   |                   |-- resolutionRule
|   |                       |- type
|   |                       |- role
|   |       |-- blockRules
|   |           |-- blockRule
|   |               |- type
|   |               |-- containers
|   |                   |- type
|   |                   |-- container
|   |                       |- id
|   |               |-- sourceNetworks
|   |                   |-- sourceNetwork
|   |                       |- type
|   |               |-- minAvailability
|   |               |-- maxAvailability
|   |       |-- rolePrecedence
|   |           |-- rolePrecedenceEntry
|   |               |- type

|-- categoryPublicationManifest
|- minSupportedClientVersion
|- majorVersion
|- minorVersion
|-- publicationList
|   |-- publicationRule
|       |- ruleType
|       |- categoryName
|       |- containerId
|       |- preferredSelfConsumption
|       |-- categoryData
|       |-- instanceId
|           |- type
|           |- value
|       |-- expireType
|           |- type
|           |- expireTime

```

The following XSD schema fragment defines the requirements to which a **publicationGrammar provisionGroup** element XML document SHOULD [<101>](#) conform.

```
<?xml version="1.0" encoding="utf-8"?>

<xs:schema xmlns="http://schemas.microsoft.com/2006/09/sip/provisiongroup-list-notification"
  targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisiongroup-list-
notification" elementFormDefault="qualified"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:ms="urn:microsoft-cpp-xml-serializer">

  <xs:complexType name="ProvisionGroupType">
    <xs:all minOccurs="0" maxOccurs="1">
      <xs:any namespace="##any" processContents="lax" minOccurs="0"
maxOccurs="unbounded">
        <xs:annotation>
          <xs:appinfo>
            </xs:appinfo>
          </xs:annotation>
        </xs:any>
      </xs:all>
      <xs:attribute name="name" fixed="publicationGrammar" use="required"/>
      <xs:anyAttribute namespace="##other" processContents="lax"/>
    </xs:complexType>
    <xs:element name="provisionGroup" type="ProvisionGroupType" />
  </xs:schema>
```

The **publicationGrammar provisionGroup** element contains the **containerManifestList** and **categoryPublicationManifest** XML documents, as shown in the example in section [4.2.2](#).

The following XSD schema fragment defines the requirements to which the **containerManifestList** XML document SHOULD [<102>](#) conform.

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema
  targetNamespace="http://schemas.microsoft.com/2008/09/sip/ContainerManifest"
  xmlns:tns="http://schemas.microsoft.com/2008/09/sip/ContainerManifest"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"
  attributeFormDefault="unqualified"
  elementFormDefault="qualified">

  <!--
  Client should not proceed with the parsing if its namespace doesn't
  match to the namespace of the xml.
  Change in namespace is an indication that format is not compatible.
  -->

  <xs:import namespace="http://schemas.microsoft.com/2006/09/sip/commontypes"
    schemaLocation="commontypes.xsd"/>

  <xs:simpleType name="sourceNetworkTypeEnum">
    <xs:restriction base="xs:token">
      <xs:enumeration value="sameEnterprise"/>
      <xs:enumeration value="federated"/>
      <xs:enumeration value="publicCloud"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="sourceNetworkTypeEnumEx">
    <xs:union memberTypes="tns:sourceNetworkTypeEnum xs:token" />
  </xs:simpleType>

  <xs:complexType name="sourceNetworkType">
```

```

<xs:sequence>
  <xs:sequence minOccurs="0" maxOccurs="1">
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <xs:element ref="ct:delimiter"/>
      <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:element ref="ct:end"/>
  </xs:sequence>
  <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>
</xs:sequence>
<xs:attribute name="type" type="tns:sourceNetworkTypeEnumEx" use="required" />
<xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="sourceNetworksType">
  <xs:sequence>
    <xs:element name="sourceNetwork" type="tns:sourceNetworkType" minOccurs="1"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!--
Container attributes.
UI will show all visible containers in the access level management view.
-->
<xs:complexType name="containerType">
  <xs:sequence>
    <xs:element name="visible" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="avoidRedundantMembers" type="xs:boolean" minOccurs="0"
maxOccurs="1"/>
    <xs:element name="blocked" type="xs:boolean" default="false" minOccurs="0"
maxOccurs="1"/>

    <xs:sequence minOccurs="0" maxOccurs="1">
      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:element ref="ct:end"/>
    </xs:sequence>
    <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>

  </xs:sequence>
  <xs:attribute name="id" type="xs:unsignedInt" />
  <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="containerIdType">
  <xs:sequence>

    <xs:sequence minOccurs="0" maxOccurs="1">
      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:element ref="ct:end"/>
    </xs:sequence>
    <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>

  </xs:sequence>
  <xs:attribute name="id" type="xs:unsignedInt" />
  <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="containersType">
  <xs:sequence>

```

```

    <xs:element name="container" type="tns:containerType" minOccurs="1"
maxOccurs="unbounded"/>

    <xs:sequence minOccurs="0" maxOccurs="1">
      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:element ref="ct:end"/>
    </xs:sequence>
    <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>
  </xs:sequence>
  <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:simpleType name="allowedContainersTypeEnum">
  <xs:restriction base="xs:token">
    <xs:enumeration value="visibleContainers"/>
    <xs:enumeration value="custom"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="allowedContainersTypeEnumEx">
  <xs:union memberTypes="tns:allowedContainersTypeEnum xs:token" />
</xs:simpleType>

<xs:complexType name="allowedContainersType">
  <xs:sequence>
    <xs:element name="container" type="tns:containerIdType" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:attribute name="type" type="tns:allowedContainersTypeEnumEx"/>
  <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:simpleType name="occurrenceTypeEnum">
  <xs:restriction base="xs:token">
    <xs:enumeration value="zero"/>
    <xs:enumeration value="one"/>
    <xs:enumeration value="zeroOrOne"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="occurrenceTypeEnumEx">
  <xs:union memberTypes="tns:occurrenceTypeEnum xs:token" />
</xs:simpleType>

<xs:complexType name="occurrenceConstraintType">
  <xs:sequence>
    <xs:sequence minOccurs="0" maxOccurs="1">
      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:element ref="ct:end"/>
    </xs:sequence>
    <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="type" type="tns:occurrenceTypeEnumEx" use="optional" />
  <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:simpleType name="roleTypeEnum">
  <xs:restriction base="xs:token">
    <xs:enumeration value="delegate"/>
    <xs:enumeration value="buddy"/>
  </xs:restriction>
</xs:simpleType>

```

```

    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="roleTypeEnumEx">
  <xs:union memberTypes="tns:roleTypeEnum xs:token" />
</xs:simpleType>

<!--
Order of precedence for which role applies if
entity maps to multiple (for example, buddy and delegate)
-->
<xs:complexType name="rolePrecedenceEntryType">
  <xs:sequence>
    <xs:sequence minOccurs="0" maxOccurs="1">
      <xs:element ref="ct:delimiter"/>
      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:sequence>
    <xs:element ref="ct:end"/>
  </xs:sequence>
  <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>
</xs:sequence>

  <xs:attribute name="type" type="tns:roleTypeEnumEx" use="required"/>
  <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="rolePrecedenceType">
  <xs:sequence>
    <xs:element name="rolePrecedenceEntry" type="tns:rolePrecedenceEntryType"
minOccurs="1" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!--
Defines resolution action to be (implictly) taken if
memberType constraints have been violated
-->
<xs:simpleType name="resolutionTypeEnum">
  <xs:restriction base="xs:token">
    <xs:enumeration value="removeFromContainer"/>
    <xs:enumeration value="moveToDefaultContainer"/>
    <xs:enumeration value="forceRole"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="resolutionTypeEnumEx">
  <xs:union memberTypes="tns:resolutionTypeEnum xs:token" />
</xs:simpleType>

<xs:complexType name="resolutionRuleType">
  <xs:sequence>
    <xs:sequence minOccurs="0" maxOccurs="1">
      <xs:element ref="ct:delimiter"/>
      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:sequence>
    <xs:element ref="ct:end"/>
  </xs:sequence>
  <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>
</xs:sequence>

  <xs:attribute name="type" type="tns:resolutionTypeEnumEx" use="required"/>
  <xs:attribute name="role" type="tns:roleTypeEnumEx" use="optional"/>
  <xs:anyAttribute processContents="lax"/>

```

```

</xs:complexType>

<xs:complexType name="resolutionRulesType">
  <xs:sequence>
    <xs:element name="resolutionRule" type="tns:resolutionRuleType" minOccurs="1"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="memberTypeEnum">
  <xs:restriction base="xs:token">
    <xs:enumeration value="user"/>
    <xs:enumeration value="domain"/>
    <xs:enumeration value="sameEnterprise"/>
    <xs:enumeration value="federated"/>
    <xs:enumeration value="publicCloud"/>
    <xs:enumeration value="everyone"/>
    <xs:enumeration value="singleIdentityUser"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="memberTypeEnumEx">
  <xs:union memberTypes="tns:memberTypeEnum xs:token" />
</xs:simpleType>

<xs:complexType name="memberType">
  <xs:sequence>

    <!--
    Allowed containers for the member.
    -->
    <xs:element name="allowedContainers" type="tns:allowedContainersType"
      minOccurs="1" maxOccurs="1"/>

    <!--
    Default container for the member.
    -->
    <xs:element name="defaultContainer" type="tns:containerIdType"
      minOccurs="0" maxOccurs="1"/>

    <!--
    Occurrence constraint on the member. There can be only one such constraint.
    Constraint check will be applied on all visible containers.
    If constraint fails, member will be deleted from all containers and placed in
    the default container.
    -->
    <xs:element name="occurrenceConstraint" type="tns:occurrenceConstraintType"
      minOccurs="0" maxOccurs="1"/>

    <xs:sequence minOccurs="0" maxOccurs="1">
      <!-- W14 extension -->
      <xs:element ref="ct:delimiter"/>
      <xs:element name="sourceNetworks" type="tns:sourceNetworksType" minOccurs="0"
maxOccurs="1"/>
      <xs:element name="resolutionRules" type="tns:resolutionRulesType" minOccurs="0"
maxOccurs="1"/>

      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:element ref="ct:end"/>
    </xs:sequence>
    <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>

  </xs:sequence>
  <xs:attribute name="type" type="tns:memberTypeEnumEx" use="optional" />

```

```

<!--
If role is present, client should identify all the members of that role and
enforce occurrence constraint on those members.
-->
<xs:attribute name="role" type="tns:roleTypeEnumEx" use="optional" />

<xs:attribute name="uri" type="xs:anyURI" use="optional" />
<xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="membersType">
  <xs:sequence>
    <xs:element name="member" type="tns:memberType" minOccurs="1"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="unboundedType">
  <xs:union memberTypes="xs:unsignedInt">
    <xs:simpleType>
      <xs:restriction base="xs:token">
        <xs:enumeration value="unbounded"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>

<xs:simpleType name="blockTypeEnum">
  <xs:restriction base="xs:token">
    <xs:enumeration value="invites"/>
    <xs:enumeration value="subscriberPrompt"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="blockTypeEnumEx">
  <xs:union memberTypes="tns:blockTypeEnum xs:token" />
</xs:simpleType>

<!--
Rule defining the blocking actions expected to be performed by clients.
-->
<xs:complexType name="blockRuleType">
  <xs:sequence>

    <!--
    Containers to which this rule apply.
    -->
    <xs:element name="containers" type="tns:allowedContainersType" minOccurs="0"
maxOccurs="1"/>

    <!--
    Source networks to which this rule apply.
    -->
    <xs:element name="sourceNetworks" type="tns:sourceNetworksType" minOccurs="0"
maxOccurs="1"/>

    <!--
    Availability range to which this rule apply.
    -->
    <xs:element name="maxAvailability" type="tns:unboundedType" default="unbounded"
minOccurs="0" maxOccurs="1"/>
    <xs:element name="minAvailability" type="tns:unboundedType" default="0"
minOccurs="0" maxOccurs="1"/>

    <xs:sequence minOccurs="0" maxOccurs="1">
      <xs:element ref="ct:delimiter"/>
      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>

```



```

        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:element ref="ct:end"/>
</xs:sequence>
    <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>

</xs:sequence>
<xs:attribute name="type" type="tns:blockTypeEnumEx" use="required" />

    <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="blockRulesType">
    <xs:sequence>
        <xs:element name="blockRule" type="tns:blockRuleType" minOccurs="1"
maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>

<xs:complexType name="containerManifestType">
    <xs:sequence>
        <xs:element name="containers" type="tns:containersType" minOccurs="0"
maxOccurs="1"/>
        <xs:element name="members" type="tns:membersType" minOccurs="0" maxOccurs="1"/>

        <xs:sequence minOccurs="0" maxOccurs="1">
            <!-- W14 extension -->
            <xs:element ref="ct:delimiter"/>
            <xs:element name="blockRules" type="tns:blockRulesType" minOccurs="0"
maxOccurs="1"/>
            <xs:element name="rolePrecedence" type="tns:rolePrecedenceType" minOccurs="0"
maxOccurs="1"/>

            <xs:sequence minOccurs="0" maxOccurs="unbounded">
                <xs:element ref="ct:delimiter"/>
                <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:element ref="ct:end"/>
        </xs:sequence>
        <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>

    </xs:sequence>
    <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="containerManifestListType">
    <xs:sequence>
        <xs:element name="containerManifest" type="tns:containerManifestType" minOccurs="1"
maxOccurs="1"/>

        <xs:sequence minOccurs="0" maxOccurs="1">
            <xs:sequence minOccurs="0" maxOccurs="unbounded">
                <xs:element ref="ct:delimiter"/>
                <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:element ref="ct:end"/>
        </xs:sequence>
        <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>

    </xs:sequence>

    <!--
Client should not proceed with the parsing if its version is less than
the version in the xml. Version mismatch is an indication that
format is not compatible.

```

```

    Missing minSupportClientVersion will be treated as 0.0.0.0, that is, always
    compatible.
    -->
    <xs:attribute name="minSupportedClientVersion" type="xs:token" use="optional"/>
    <xs:attribute name="majorVersion" type="xs:unsignedInt" use="optional"/>
    <xs:attribute name="minorVersion" type="xs:unsignedInt" use="optional"/>

    <xs:anyAttribute processContents="lax"/>
  </xs:complexType>

  <xs:element name="containerManifestList" type="tns:containerManifestListType">
  </xs:element>
</xs:schema>

```

The following XSD schema fragment defines the requirements to which the **categoryPublicationManifest** XML document SHOULD [103](#) conform.

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- edited with XMLSpy v2007 rel. 3 (http://www.altova.com) by Adarsh Khare (Microsoft) -->
<!-- Adarsh Khare (Microsoft) -->
<xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema"
  xmlns:tns="http://schemas.microsoft.com/2008/09/sip/categoryPublicationManifest"
  targetNamespace="http://schemas.microsoft.com/2008/09/sip/categoryPublicationManifest"
  elementFormDefault="qualified" attributeFormDefault="unqualified">

  <xs:simpleType name="instanceIdValuesEnum">
    <xs:annotation>
      <xs:documentation>List the possible value of instanceId type
attribute</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:token">
      <xs:enumeration value="constant"/>
      <xs:enumeration value="custom"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="instanceIdValuesEnumEx">
    <xs:union memberTypes="tns:instanceIdValuesEnum xs:token" />
  </xs:simpleType>

  <xs:simpleType name="categoryDataValuesEnum">
    <xs:annotation>
      <xs:documentation>List the the possible value to define for
categoryData</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:token">
      <xs:enumeration value="xslt"/>
      <xs:enumeration value="xml"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:simpleType name="categoryDataValuesEnumEx">
    <xs:union memberTypes="tns:categoryDataValuesEnum xs:token" />
  </xs:simpleType>

  <xs:simpleType name="expiryTypeValuesEnum">
    <xs:annotation>
      <xs:documentation>List the possible values of expiryTypeValues</xs:documentation>
    </xs:annotation>
    <xs:restriction base="xs:token">
      <xs:enumeration value="endpoint"/>
      <xs:enumeration value="static"/>
      <xs:enumeration value="user"/>
      <xs:enumeration value="time"/>
    </xs:restriction>
  </xs:simpleType>

```

```

        <xs:enumeration value="custom"/>
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="expiryTypeValuesEnumEx">
    <xs:union memberTypes="tns:expiryTypeValuesEnum xs:token" />
</xs:simpleType>

<xs:complexType name="instanceIdType">
    <xs:attribute name="type" type="tns:instanceIdValuesEnumEx" use="required"/>
    <xs:attribute name="value" type="xs:unsignedInt" use="optional"/>
</xs:complexType>

<xs:complexType name="categoryDataType">
    <xs:sequence minOccurs="0">
        <xs:any namespace="##any" processContents="lax"/>
    </xs:sequence>
    <xs:attribute name="type" type="tns:categoryDataValuesEnumEx" use="required"/>
    <xs:attribute name="categoryName" type="xs:string"/>
    <!--Specify the CategoryName attribute, if category name being published is different
from original category name, for example. When UserInformation is published it could also be
translated in one or more contactCard publications.-->
</xs:complexType>

<xs:complexType name="publicationType">
    <xs:sequence>
        <xs:element name="instanceId" type="tns:instanceIdType" minOccurs="1" maxOccurs="1"/>
        <xs:element name="categoryData" type="tns:categoryDataType" minOccurs="0"
maxOccurs="1"/>
    </xs:sequence>

    <xs:attribute name="categoryName" type="xs:string"/>
    <xs:attribute name="containerId" type="xs:unsignedShort"/>
    <xs:attribute name="expiryType" type="tns:expiryTypeValuesEnumEx" use="optional">
        <xs:annotation>
            <xs:documentation>If expiryType is defined as Time, expiryTime must be
defined</xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="allowAutoVersionFix" type="xs:boolean" use="optional"/>
    <xs:attribute name="expiryTime" type="xs:duration" use="optional">
        <xs:annotation>
            <xs:documentation>expiryTime is only meaningful for expiryType
Time</xs:documentation>
        </xs:annotation>
    </xs:attribute>
    <xs:attribute name="publishAtStartup" type="xs:boolean" use="optional"/>

    <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="publicationListType">
    <xs:sequence>
        <xs:element name="publication" type="tns:publicationType" maxOccurs="unbounded"/>
    </xs:sequence>

    <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="categoryPublicationManifestType">
    <xs:sequence>
        <xs:element name="publicationList" type="tns:publicationListType" minOccurs="1"
maxOccurs="1"/>
    </xs:sequence>

    <!--
    Client should not proceed with the parsing if its major version doesn't

```

```

match to the version in the xml.
Change in major version indicates that format is not compatible.
-->
<xs:attribute name="majorVersion" type="xs:unsignedInt" use="required" />

<!--
Change in minor version without any change in the major version indicates that
format is backward compatible.
-->
<xs:attribute name="minorVersion" type="xs:unsignedInt" use="required" />
<xs:anyAttribute processContents="lax"/>

</xs:complexType>

<xs:element name="categoryPublicationManifest" type="tns:categoryPublicationManifestType"/>

</xs:schema>

```

The following XSD schema fragment defines the schema for **commontypes** referred by the **containerManifestList** XSD schema.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  targetNamespace="http://schemas.microsoft.com/2006/09/sip/commontypes"
  xmlns:tns="http://schemas.microsoft.com/2006/09/sip/commontypes"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  >

  <xs:element name="delimiter">
    <xs:complexType />
  </xs:element>

  <xs:element name="end">
    <xs:complexType />
  </xs:element>

  <xs:complexType name="extensionType">
    <xs:sequence>
      <xs:any processContents="lax" minOccurs="1"
        maxOccurs="unbounded" namespace="##other"/>
    </xs:sequence>
  </xs:complexType>

  <xs:element name="extension" type="tns:extensionType"/>

</xs:schema>

```

The following XSD schema fragment defines the schema for **commontypesForNewPublicationManifest** referred by the **categoryPublicationManifest** XSD schema.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  xmlns:ms="urn:microsoft-cpp-xml-serializer"
  targetNamespace="http://schemas.microsoft.com/2006/09/sip/commontypes"
  xmlns:tns="http://schemas.microsoft.com/2006/09/sip/commontypes"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  >

```

```

<xs:element name="delimiter">
  <xs:complexType />
</xs:element>

<xs:element name="end">
  <xs:complexType />
</xs:element>

<xs:complexType name="extensionType">
  <xs:all>
    <xs:any processContents="lax" minOccurs="1"
      maxOccurs="unbounded" namespace="##other"/>
  </xs:all>
</xs:complexType>

<xs:element name="extension" type="tns:extensionType"/>

</xs:schema>

```

containerManifestList: A list of all **containerManifests** available.

containerManifest: Manifest that provides the list of containers and members. The UAC SHOULD create containers and its members based on the constraints defined in the manifest.

majorVersion (optional): Ignored.

minorVersion (optional): Ignored.

minSupportedClientVersion (optional): The minimum version of the UAC that consumes the container manifest.

containers: The wrapper for the list of **container** elements.

container: Each **container** element describes a container.

- **id (required):** Unsigned short integer that uniquely identifies a container.
- **Visible (optional):** [<104>](#) **Container** attribute that, when "true", designates the container as a member of the set of containers identified with allowed containers **type** set to "visibleContainers". UAC also interprets containers that set the **Visible** attribute to "true" as containers that are displayed in its access level view.
- **avoidRedundantMembers (optional):** This string indicates if a container can have redundant membership. For example, redundancy occurs when a domain, as well as a user who is in that domain, are associated with a container. The value MUST be "true" or "false". The UAC SHOULD clean up redundant members from containers if the setting is "true".
- **blocked (optional):** Ignored.

members: The wrapper for the list of member elements.

member: Describes a container member.

type (optional): This string value specifies the type of the member. The following values are defined:

- **user:** A SIP URI.
- **domain:** A SIP domain.
- **sameEnterprise:** All users from the same enterprise.
- **federated:** All federated users.

- **publicCloud:** All public cloud users.
- **everyone:** All users.
- **singleIdentityUser:** User who has just one SIP uri for that particular identity. PIC contacts that have more than one sip uri that belong to the identity should not be processed using this rule.

role (optional): This string value specifies the role of the member. Note that container members matching to a member by **role** attribute take precedence over the corresponding match to member by **type** attribute. The following values are defined:

- **delegate:** A user entity currently configured as a delegate for the local user.
- **buddy:**[<105>](#) A user entity currently a member of the local user's buddy list.

allowedContainers: List of containers to which the member MAY be added.

- **type (required):** The **type** attribute is a string that describes the access control element. The following values are defined:
 - **custom:** The list of containers is specified within the **allowedContainers** element.

visibleContainers:[<106>](#) The list of containers for the member are the subset of child elements of the **containers** element with the **Visible** attribute set to "true".

- **container (optional):** This element specifies the container to which the member MAY be added. There MAY be more than one element. The list of containers is specified when the **type** is "custom".
 - **id (required):** This unsigned short integer uniquely identifies a container.
- **defaultContainer (optional):** The default container to which the member MAY be added if the constraint is not satisfied for the member.
 - **id (required):** An unsigned integer that gives the container identifier.

occurrenceConstraint (optional): The mandatory membership occurrence rate the related type or role member is present in.

- **type (required):** A string that gives the number of occurrences mandated for the given type or role. The value MUST be "zero", "one" or "zeroOrOne".
 - When "Zero", a given container MUST have zero members of the related type or role. This value SHOULD be ignored by a UAC.
 - When "One", a given container MUST have one member of the related type or role.
 - When "zeroOrOne", a given container MAY have no members of the related type or role or it MAY have one member of a related type or role. It MUST NOT have more than one member of a related type or role.

sourceNetworks (optional): A wrapper for the optional list of source networks to which the member belongs to satisfy the constraint.

sourceNetwork: Describes a source network.

- **type (required):** A string that describes the source network. The following values are defined:
 - **sameEnterprise:** All users from the same enterprise.
 - **federated:** All federated users.

- **publicCloud:** All public cloud users.

resolutionRules (optional): A wrapper for the optional list of resolution rules that are applied if the constraint is not satisfied for the member.

resolutionRule: Describes a resolution rule.

- **type (required):** The **type** attribute is a string that describes the resolution rule. The following values are defined:
 - **removeFromContainer:** The member is removed from the container if the constraint is not satisfied for the member.
 - **moveToDefaultContainer:** The member is moved to the default container if the constraint is not satisfied for the member.
 - **forceRole:** The member is forced into the role specified by the **role** element.
- **role (optional):** A string that describes the role the member is forced into when the constraint is not satisfied for the member. This attribute is set when the **type** attribute has the value "forceRole". Possible values for the **role** attribute are "delegate" and "buddy".

blockRules (optional): The wrapper for the list of **blockRule** elements.

blockRule: Describes a **blockRule** element.

- **type (required):** The **type** attribute is a string that describes the actions that are blocked. The following values are defined:
 - **invites:** Invites are blocked when the rule matches.
 - **subscriberPrompt:** Subscriber prompts are blocked when the rule matches.
- **containers (optional):** The actions from the members of specified containers are blocked.
- **sourceNetworks (optional):** The actions from the members of specified source networks are blocked.
- **maxAvailability (optional):** Defines the upper bound of the availability range of the publisher for which this rule applies.
- **minAvailability (optional):** Defines the lower bound of the availability range of the publisher for which this rule applies.

rolePrecedence (optional): The wrapper for the list of **rolePrecedenceEntry** elements that define the order of precedence for roles when a member maps to multiple roles.

rolePrecedenceEntry: Describes a role.

- **type (required):** A string that specifies a role. The possible values are "delegate" and "buddy".

categoryPublicationManifest: Manifest that describes the list of presence publication rules.

minSupportedClientVersion (optional): The minimum version of the UAC that consumes the category publication manifest.

majorVersion (optional): Ignored.

minorVersion (optional): Ignored.

publicationList: A wrapper for **publicationRules**.

publicationRule: The rules for the publication of a category in a given container.

- **ruleType (required):** This string value specifies the type of the rule. The following values are defined:
 - **bootstrap:** Rule that seeds initial presence data.
 - **transformation:** Rule that modifies the presence data before publication according to XSL transformation defined in the **categoryData** element.
 - **cleanup:** Rule that un-publishes existing presence data.
- **categoryName (required):** The name of the target category to which the publication rule applies.
- **containerID (required):** The identifier of the target container to which the publication rule applies.
- **instance:** Ignored.
- **preferredSelfConsumption (optional):** Indicates that the **containerId** specified in the rule is a preferred container for consuming presence data of the category specified in the same rule.
- **categoryData:** The XSL if the **ruleType** is "transformation". Empty content indicates that no transformation MUST be applied. Contains the XML to be published if the **ruleType** is "bootstrap".
- **expireType (optional):** Specifies the lifetime of the publication.
- **type (required):** A string that specifies the rules for expiring the publication. The following values are defined:
 - **endpoint:** The publication lives as long as the publishing endpoint stays registered.
 - **user:** The publication lives as long as the publishing user stays registered from at least one endpoint.
 - **static:** The publication never expires.
 - **time:** The publication expires if it is not refreshed within a certain period of time.
- **expireTime (optional):** Specifies the lifetime of the publication when **expireType** is set to "time". This value is required when **type** is "time".

instanceId (optional): This element allows different instances of the same category to be published to the same container. For example, a publisher might use two different endpoints to publish the same category to the same container, yet these publications will be treated by the server as distinct publications if they use different **instanceIds**.

For a detailed example, see section 4.2.2.

2.2.2.5.5 Data Model for userSetting provisionGroup

The **userSetting provisionGroup** is specified as follows.

```
provisionGroup (name='userSetting')
|
|-- disableFileTransfer
|-- ucUserLocationProfile
|-- commonAreaPhone
```


The following XSD schema fragment defines the requirements to which a **userSetting provisionGroup** element XML document SHOULD [<107>](#) conform.

```
<?xml version="1.0" encoding="utf-16"?>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification" targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="provisionGroup" type="tns:provisionGroupType" />
  <xs:complexType name="provisionGroupType">
    <xs:all>
      <xs:element minOccurs="1" maxOccurs="1" name="disableFileTransfer"
type="tns:booleanType" />
      <xs:element minOccurs="1" maxOccurs="1" name="ucUserLocationProfile"
type="tns:stringType" />
      <xs:element minOccurs="0" maxOccurs="1" name="commonAreaPhone" type="tns:booleanType"
/>
    </xs:all>
    <xs:attribute fixed="userSetting" name="name" use="required" />
  </xs:complexType>
  <xs:simpleType name="stringType">
    <xs:restriction base="xs:string" />
  </xs:simpleType>
  <xs:simpleType name="booleanType">
    <xs:restriction base="xs:boolean" />
  </xs:simpleType>
</xs:schema>
```

disableFileTransfer: [<108>](#) Specifies whether file transfer is disabled. The value SHOULD be "true" or "false".

ucUserLocationProfile: Supplies the location profile for voice call features. This is used by the extensions described in [\[MS-SIPRE\]](#) section 2.2.8.

commonAreaPhone: Specifies whether the client is a Common Area phone. The value MUST be "true" or "false".

For a detailed example, see section [4.2.2](#).

2.2.2.5.6 Data Model for endpointConfiguration provisionGroup

The **endpointConfiguration provisionGroup** is as follows. [<109>](#)

```
provisionGroup (name='endpointConfiguration')
|
|-- ShowRecentContacts
|-- ShowManagePrivacyRelationships
|-- MaxPhotoSizeKB
|-- DisableMusicOnHold
|-- AttendantSafeTransfer
|-- MusicOnHoldAudioFile
|-- CustomLinkInErrorMessage
|-- CustomStateUrl
|-- DisablePoorDeviceWarnings
|-- DisablePoorNetworkWarnings
|-- BlockConversationFromFederatedContacts
|-- CalendarStatePublicationInterval
|-- EnableCallLogAutoArchiving
|-- EnableAppearOffline
|-- EnableConversationWindowTabs
|-- EnableEventLogging
|-- EnableFullScreenVideoPreviewDisabled
|-- EnableSQMData
|-- EnableTracing
|-- EnableURL
```

```

|-- EnableIMAutoArchiving
|-- DisableEmailComparisonCheck
|-- DisableCalendarPresence
|-- DisableEmoticons
|-- DisableFederatedPromptDisplayName
|-- DisableFreeBusyInfo
|-- DisableHandsetOnLockedMachine
|-- DisableHtmlIm
|-- DisableInkIM
|-- DisableRTFIM
|-- DisableSavingIM
|-- DisableMeetingSubjectAndLocation
|-- DisableOneNote12Integration
|-- DisableOnlineContextualSearch
|-- DisablePhonePresence
|-- DisablePICPromptDisplayName
|-- DisablePresenceNote
|-- PhotoUsage
|-- AbsUsage
|-- AllowUnencryptedFileTransfer
|-- AutoDiscoveryRetryInterval
|-- ConferenceIMIdleTimeout
|-- DGRefreshInterval
|-- ExcludedContactFolders
|-- IMWarning
|-- MapiPollInterval
|-- MaximumNumberOfContacts
|-- NotificationForNewSubscribers
|-- PlayAbbrDialTone
|-- SearchPrefixFlags
|-- TabURL
|-- WebServicePollInterval
|-- DisableFeedsTab
|-- EnableEnterpriseCustomizedHelp
|-- CustomizedHelpUrl
|-- DisableContactCardOrganizationTab
|-- EnableHotdesking
|-- HotdeskingTimeout
|-- SPSearchInternalUrl
|-- SPSearchExternalUrl
|-- SPSearchCenterInternalUrl
|-- SPSearchCenterExternalUrl
|-- EnableExchangeDelegateSyncUp
|-- EnableContactSync
|-- ShowSharepointPhotoEditLink
|-- EnableVOIPCallDefault
|-- MaximumDGsAllowedInContactList
|-- ImLatencySpinnerDelay
|-- ImLatencyErrorThreshold
|-- EnableMediaRedirection
|-- P2PAppSharingEncryption
|-- HelpEnvironment
|-- RateMyCallAllowCustomUserFeedback
|-- RateMyCallDisplayPercentage
|-- imLatencySpinnerDelay
|-- imLatencyErrorThreshold
|-- EnableHighPerformanceP2PAppSharing
|-- EnableHighPerformanceConferencingAppSharing
|-- TracingLevel
|-- EnableServerConversationHistory
|-- EnableSkypeUI
|-- CallViaWorkEnabled
|-- UseAdminCallbackNumber
|-- AdminCallbackNumber

```

The following XSD schema fragment defines the requirements to which an **endpointConfiguration provisionGroup** element XML document SHOULD conform.

```

<?xml version="1.0" encoding="utf-16"?>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2006/09/sip/provisioningrouplist-
notification" targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisioningrouplist-
notification" xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="propertyEntryType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="name" type="xs:string" use="required" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="propertyEntryListType">
    <xs:sequence>
      <xs:element name="property" type="tns:propertyEntryType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="propertyEntryList" type="tns:propertyEntryListType" />

</xs:schema>

```

ShowRecentContacts: Ignored.

AbsUsage: Specifies whether the user MAY use an Address Book Service (as specified in [\[MS-ABS\]](#)) web query, download a file, or either one for accessing an **address book**. Values include "WebSearchAndFileDownload", "WebSearchOnly", or "FileDownloadOnly".

AllowUnencryptedFileTransfer: Allows unencrypted files to be transferred.

MusicOnHoldAudioFile: Optional, ignored.

DisableMusicOnHold: Optional, ignored.

AttendantSafeTransfer: Optional, ignored.

AutoDiscoveryRetryInterval: How frequently to retry autodiscovery if it fails.

BlockConversationFromFederatedContacts: This policy prevents external contacts from inviting users to IM or Audio-Video conversations unless they are in the user's High Presence level. High Presence means that they are in the ALLOW **access control list (ACL)** of the user. In addition, it blocks messages that are not part of a session from reaching the user.

CalendarStatePublicationInterval: How frequently data loaded from the calendar is published, in seconds.

ConferenceIMIdleTimeout: Amount of time a user MAY sit in an IM-only conference without sending or receiving an IM. When the timeout occurs, the user automatically leaves the IM conference.

CustomizedHelpUrl: Determines the root URL to be used when passing help parameters to the customized Help system of the enterprise.

CustomLinkInErrorMessage: Custom link pointing users to an enterprise-hosted extranet site from the error messages.

CustomStateUrl: URL to an XML file containing custom Presence states.

DGRefreshInterval: Timing period, in seconds, for refreshing the Distribution Group.

DisableCalendarPresence: Disables loading Free/Busy data from **Messaging Application Programming Interface (MAPI)** or web services. Also disables publication of free/busy data.

DisableContactCardOrganizationTab: Removes the contact card organization tab from the UI.

DisableEmailComparisonCheck: Disables the e-mail comparison check; normally, if the e-mail client profile's **email-id** does not match the **Simple Mail Transfer Protocol (SMTP)** address received from the e-mail server<110>, any personal information management (PIM) integration<111> is disabled. This policy is disabled by default.

DisableEmoticons: Prevents emoticons being shown in instant messages.

DisableFederatedPromptDisplayName: The display name of federated, non-**public IM connectivity**, contacts in the notification dialog.

DisableFeedsTab: Indicates whether the feed environment is shown in the UI. If this setting is enabled, the feed environment and the feed tab are removed from the UI.

DisableFreeBusyInfo: Indicates whether free/busy information is published. If this setting is enabled, free/busy information is not published.

DisableHandsetOnLockedMachine: When the PC is locked, this setting does not allow calls to or from the handset.

DisableHtmlIm: Disables HTML instant messages.

DisableInkIM: Ignored.

DisableMeetingSubjectAndLocation: Indicates whether the meeting subject and location information is published.

DisableOneNote12Integration: Specifies whether planning and note-taking software integration with the instant-messaging client is enabled.<112>

DisableOnlineContextualSearch: Disables online searches for related conversations. This disables both the automatic population of previous logs, and the ability to start a contextual search for conversation related to emails. This only affects users who are using an e-mail client<113> in online mode.

DisablePhonePresence: When this policy is enabled, phone call states are not published as part of the presence information.

DisablePICPromptDisplayName: The display name of public IM connectivity contacts in the notification dialog.

DisablePoorDeviceWarnings: Disables poor device warnings that appear during the first run, in the tuning wizard, in the main UI, and in the conversation window at the endpoint.

DisablePoorNetworkWarnings: Disables poor network warnings that appear during the conversation at the endpoint.

DisablePreCallNetworkDiagnostics: Disables poor network warnings that appear during the conversation at the endpoint.

DisablePresenceNote: Disables the loading of the OOF (out of office) message from MAPI or web services. Also disables the publication of OOF messages.

DisableRTFIM: Indicates that rich text MUST NOT be allowed for IM.

DisableSavingIM: Prevents users from saving instant messages.

EnableAppearOffline: Controls the appear offline entry points on the client. The value MUST be "true" or "false".

EnableCallLogAutoArchiving: When enabled, the client always archives call logs. When disabled, the client never archives call logs.

EnableConversationWindowTabs: Enables the web browser in the conversation window.

EnableEnterpriseCustomizedHelp: Determines whether the enterprise wants the default online help system or uses a completely customized help system.

EnableEventLogging: Turns on UCCP event logging for the client.

EnableExchangeDelegateSyncUp: When this is enabled, the delegate's information is retrieved from the calendar, with author and editor rights on the boss's calendar.

EnableFullScreenVideoPreviewDisabled: Enables full screen video with the correct aspect ratio and disables video preview for all client video calls.

EnableHotdesking: Specifies whether hotdesking is enabled for a common area phone. Only phones with the common area phone device type make use of this setting.

EnableIMAutoArchiving: When this policy is enabled, the client always archives IM conversations. When it is disabled, the client never archives IM conversations. When this policy is not set, the user controls archiving of IM conversations.

EnableSQMData: Enables collection of software quality metrics.

EnableTracing: Turns on tracing for the client; primarily for use by customer support and the OC team to assist customer problem solving.

EnableURL: Allows hyperlinks in instant messages.

ExcludedContactFolders: This policy is used for excluding contact folders from being imported into the client search results.

HotdeskingTimeout: The common area phone operating in hotdesking mode uses this setting to determine when to log the user out and revert to common area mode if the user has been inactive for the **hotdeskingTimeout** period. This setting is only made use of by phones with the common area device type and the **enableHotdesking** setting set to "true".

IMWarning: Allows the administrator to configure the initial text that appears in the instant messaging area when a conversation window is opened.

MapiPollInterval: The frequency, in minutes, of loading calendar data from the MAPI provider.

MaximumNumberOfContacts: Maximum number of contacts users are allowed to have.

MaxPhotoSizeKB: Maximum size of an individual photo that a client MAY download.

NotificationForNewSubscribers: Notifications are shown unless the user has selected otherwise in the options dialog.

PhotoUsage: Whether client endpoints display photos of each type. Valid values are "NoPhoto", "PhotosFromADOnly", and "AllPhotos".

PlayAbbrDialTone: Changes the length of the dial tone from a 30-second dial tone to a fading, 3-second dial tone.

SearchPrefixFlags: A DWORD value whose bits represent the decision of which address book attribute to index into the prefix search tree. These bits are as follows (from low to high):

- **Bit 0:** Primary e-mail.

- **Bit 1:** Alias.
- **Bit 2:** All emails.
- **Bit 3:** Company.
- **Bit 4:** Display name.
- **Bit 5:** First name.
- **Bit 6:** Last name.

ShowManagePrivacyRelationships: Controls whether the "view by" menu in the contact list shows or hides "manage privacy relationships" (previously known as access level management view). Valid values are "true" and "false".

SPSearchCenterExternalUrl: The **site** search center external URL is used to enable the site link in the expert search results. This URL represents the search center outside the enterprise network of the search service specified in the **ExpertSearchExternalURL**. If this setting is blank, there is no link to the site server at the bottom of the search results.

SPSearchCenterInternalUrl: The site search center internal URL is used to enable the site link in the expert search results. This URL represents the search center within the enterprise network of the search service specified in the **ExpertSearchInternalURL**. If this setting is blank, there is no link to the site server at the bottom of the search results.

SPSearchExternalUrl: The site external URL is used to enable expert search capabilities if the client is connected to the server outside of the enterprise network. If the URL is blank or invalid, none of the expert search capabilities and UI are enabled.

SPSearchInternalUrl: The site internal URL is used to enable expert search capabilities if the client is connected to the server inside of the enterprise network. If the URL is blank or invalid, none of the expert search capabilities and UI are enabled.

TabURL: URL for the XML file from which the tab definitions are loaded.

WebServicePollInterval: The frequency, in minutes, of loading calendar data from a web services provider.

EnableContactSync: Determines whether the client synchronizes contacts into the e-mail server contacts store. [<114>](#) The value MUST be "true" or "false".

ShowSharepointPhotoEditLink: Determines whether the client shows the site link to edit their photo. The value MUST be "true" or "false".

EnableVOIPCallDefault: Determines whether the client chooses **Voice over IP (VoIP)** call as the default calling method instead of a PSTN call when click to call is used. The value MUST be "true" or "false".

MaximumDGsAllowedInContactList: Determines the number of distribution groups allowed in the contact list of the client. The type for this is unsigned integer.

EnableMediaRedirection: Enables an enterprise-grade audio/video experience. [<115>](#)

HelpEnvironment: Specifies the help documentation that the client shows to the user.

P2PAppSharingEncryption: An integer value that specifies the encryption mode for application sharing sessions between clients.

- **0 :** Supported

- **1** : Enforced
- **2** : Not Supported

TracingLevel: <116> Controls the tracing level on the client. This with **EnableTracing** controls tracing, if **EnableTracing** is not set then this attribute controls the level of tracing and the user can change the tracing level, if **EnableTracing** is set then the user cannot change the tracing level and the tracing level is determined by the value of this attribute. This string attribute MUST have one of the following values.

- **Off**
- **Light**
- **Full**

RateMyCallAllowCustomUserFeedback: bool value indicates whether the call rating dialog should include a free form text field, where user can provide custom feedback.

RateMyCallDisplayPercentage: An integer value in percentage value between 0 and 100 indicates the percentage of calls for which we request call rating information from the user at the end of call. 0 implies that we will never request call rating information. 100 implies that we call request call rating information at the end of every call. 10 implies that we will request call rating information for 1 call out of every 10 calls. Default value is 10.

ImLatencySpinnerDelay: Amount of time in milliseconds to wait before showing the spinner in the client when IM delivery is delayed

ImLatencyErrorThreshold: An integer value in millisecond. If IM latency is above this threshold the client will submit a CER

EnableServerConversationHistory: Boolean value that indicates Lync Server can take care of conversation history for particular user

P2PAppSharingEncryption:

EnableHighPerformanceP2PAppSharing: Boolean value to enable high frame rate application sharing in a 2 party conversation. To enable the best experience, the setting should be enabled on both the sharer and viewer. When this setting is enabled, the actual frame rate is determined by the AppSharingBitRate setting. If the AppSharingBitRate is less than or equal to 2000, then the frame rate is calculated as 2.5 frames per second. If the AppSharingBitRate is between 2000 and 4000, the frame rate is calculated as 4 frames per second. If the AppSharingBitRate is between 4000 and 8000, then the frame rate is calculated as 6 frames per second. If the AppSharingBitRate is between 8000 and 12000, then the frame rate is calculated as 8 frames per second. If the AppSharingBitRate is greater than 12000, then the frame rate is calculated as 10 frames per second. The default is false, and when false, the default frame rate is 2.5 frames per second

EnableHighPerformanceConferencingAppSharing: Boolean value to enable high frame rate application sharing in a multi-party (conference) conversation. Note that when enabled, this may have a performance impact on the application sharing conference server, possibly resulting in degraded performance if the server is heavily loaded. To enable the best experience, the setting should be enabled on both the sharer and viewer. When this setting is enabled, the actual frame rate is determined by the AppSharingBitRate setting. If the AppSharingBitRate is less than or equal to 2000, then the frame rate is calculated as 2.5 frames per second. If the AppSharingBitRate is between 2000 and 4000, the frame rate is calculated as 4 frames per second. If the AppSharingBitRate is between 4000 and 8000, then the frame rate is calculated as 6 frames per second. If the AppSharingBitRate is between 8000 and 12000, then the frame rate is calculated as 8 frames per second. If the AppSharingBitRate is greater than 12000, then the

frame rate is calculated as 10 frames per second. The default is false and when false, the default frame rate is 2.5 frames per second

EnableSkypeUI: Boolean value that controls whether the Lync client should display a Lync-style or Skype-style user interface. If set to 'true', the client should display a Skype-style interface; if 'false' the client should display a Lync-style interface; if the setting is absent, the client may choose internally.

CallViaWorkEnabled: Boolean value to enable/disable a user for CallViaWork functionality.

AdminCallbackNumber: IT Administrator specified callback phone number for a specific user

UseAdminCallbackNumber: Boolean value when set to true forces the user to use AdminCallbackNumber as their callback phone number. AdminCallbackNumber must be configured when set to true. When set to false, user has the ability to specify their own callback phone number from UI in addition to IT Administrator specified AdminCallbackNumber

For a detailed example, see section [4.2.2](#).

2.2.2.5.7 Data Model for locationPolicy provisionGroup

The **locationPolicy provisionGroup** follows the behavior described in product behavior note [<117>](#).

```
provisionGroup (name='locationPolicy')
|
|-- propertyEntryList
|   |-- property
|   |   |-- name
```

The following XSD schema fragment defines the requirements to which a **locationPolicy provisionGroup** element XML document SHOULD conform.

```
<?xml version="1.0" encoding="utf-16"?>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification" targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification" xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="propertyEntryType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="name" type="xs:string" use="required" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:complexType name="propertyEntryListType">
    <xs:sequence>
      <xs:element name="property" type="tns:propertyEntryType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
  <xs:element name="propertyEntryList" type="tns:propertyEntryListType" />

</xs:schema>
```

EnhancedEmergencyServicesEnabled: If "true", the client acquires location information and sends PIDF-LO with an emergency call. The value MUST be "true" or "false".

LocationPolicyTagID: If this element is present, the client MUST insert this value into the PIDF-LO of every emergency call. [\[MS-SIPRE\]](#) section 3.17.5.2 specifies the usage of this field in detail.

LocationRequired: Triggers the client to request alternate location sources from the user if the **IsInternalUri** (section [2.2.2.5.1](#)) is not populated or the Location Information Service does not return a location. Valid values are "yes", "no", or "disclaimer".

- "yes" – triggers the client to show an alert indicator and text about entering a location.
- "no" – do nothing.
- "disclaimer"– show an alert indicator and text about entering a location, and if the user clicks through then show the **EmergencyServiceDisclaimer** (section 2.2.2.5.7).

UseLocationForE911Only: If "true", the location is sent only in a 911 call. The value MUST be "true" or "false".

PstnUsage: Ignored.

EmergencyDialString: The dial string or URI that triggers the client to send PIDF-LO and initiates the E911-related UI capability.

EmergencyDialMask: Dial strings or URIs that, if entered by a user, are translated into the emergency dial string. If there are multiple values, they are separated by a semicolon (;). For example, a European employee, while visiting the United States, might dial 112 in an emergency and this property helps route the call correctly.

NotificationUri: URIs that the emergency notification will be sent to. The notification URI MAY be one SIP URI or list of comma separated SIP URIs. Group and DL URIs are not supported.

ConferenceUri: URI of the person or group that is conferenced-in in the emergency call.

ConferenceMode: Toggle setting of one-way or two-way audio for an emergency conference. Valid values are "oneway" or "twoway".

EmergencyServiceDisclaimer: Text entered by the administrator for the client to display an emergency services disclaimer if it could not fetch a location from the Location Information Service, as specified in [\[MS-E911WS\]](#) section 3.1.4.1.

LocationRefreshInterval: The interval in hours for Location Information Service requests from the client.

For a detailed example, see section [4.2.2](#).

2.2.2.5.8 Data Model for mediaConfiguration provisionGroup

```
provisionGroup (name='mediaConfiguration')
```

```
|
|-- propertyEntryList
|   |-- property
|   |   |-- name
```

The following XSD schema fragment defines the requirements to which a **mediaConfiguration provisionGroup** element XML document SHOULD conform. [<118>](#)

```
<?xml version="1.0" encoding="utf-16"?>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification" targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification" xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="propertyEntryType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
```

```

        <xs:attribute name="name" type="xs:string" use="required" />
    </xs:extension>
</xs:simpleContent>
</xs:complexType>
<xs:complexType name="propertyEntryListType">
    <xs:sequence>
        <xs:element name="property" type="tns:propertyEntryType" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="propertyEntryList" type="tns:propertyEntryListType" />
</xs:schema>

```

bypassEnabled: Specifies whether bypass is enabled for this deployment. The value MUST be "true" or "false".

internalBypassMode: If bypass is enabled for this deployment, specifies whether bypass is enabled for internal users.

externalBypassMode: If bypass is enabled for this deployment, specifies whether bypass is enabled for outside users.

bypassId: If bypass is enabled for this deployment, specifies the Bypass identifier to be used if the deployment is configured for "alwaysBypass", or the default Bypass identifier to use if no Bypass identifier is returned as a result of region or site matching.

For a detailed example, see section [4.2.2](#).

2.2.2.5.9 Data Model for presencePolicyV2 provisionGroup

The **presencePolicyV2 provisionGroup** is specified as follows. [<119>](#)

```

provisionGroup (name='presencePolicyV2')
|
|-- propertyEntryList
|   |-- property
|   |   |-- name

```

The following XSD schema fragment defines the requirements to which a **presencePolicyV2 provisionGroup** element XML document MUST conform.

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification" targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification" elementFormDefault="qualified" xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:element name="provisionGroup" type="tns:provisionGroupType" />
    <xs:complexType name="provisionGroupType">
        <xs:all>
            <xs:element minOccurs="1" maxOccurs="1" name="propertyEntryList"
type="tns:propertyEntryListType" />
        </xs:all>
        <xs:attribute fixed="presencePolicyV2" name="name" use="required" />
    </xs:complexType>
    <xs:complexType name="propertyEntryType">
        <xs:simpleContent>
            <xs:extension base="xs:string">
                <xs:attribute name="name" type="xs:string" use="required" />
            </xs:extension>
        </xs:simpleContent>
    </xs:complexType>

    <xs:complexType name="propertyEntryListType">

```

```

    <xs:sequence>
      <xs:element name="property" type="tns:propertyEntryType" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:schema>

```

propertyEntryList: A single **propertyEntryList** element is specified within the **presencePolicyV2 provisionGroup**, and lists a set of presence-related settings.

property: A single presence setting. The name attribute specifies the property name. Values are specified in CDATA notation.

The following properties MUST be present.

EnablePrivacyMode: Indicates whether privacy mode is enabled on the server. The value MUST be "true" or "false".

AutoInitiateContacts: Specifies whether the first-run automatic population of the contact list needs to be enabled on the client. The value MUST be "true" or "false".

PublishLocationDataDefault: Specifies whether clients need to publish location data by default. The value MUST be "true" or "false".

DisplayPublishedPhotoDefault: Specifies whether the user's published photo is displayed by the client by default. The value MUST be "true" or "false".

PersonalNoteHistoryDepth: The maximum number of child note elements stored in the single NoteHistory instance that MAY be published in a container at any given time. The value is an unsigned integer.

SubscribeToCollapsedDG: Specifies whether the client needs to subscribe to presence information of distribution group members when collapsed in the UI.

For a detailed example, see section [4.2.2](#).

2.2.2.5.10 Data Model for **privacyPublicationGrammar provisionGroup**

```

provisionGroup (name='privacyPublicationGrammar')
|
|-- containerManifestList
|  |- minSupportedClientVersion
|  |- majorVersion
|  |- minorVersion
|  |-- containerManifest
|  |  |-- containers
|  |  |  |-- container
|  |  |  |  |- id
|  |  |  |  |-- visible
|  |  |  |  |  |-- avoidRedundantMembers
|  |  |  |  |  |-- blocked
|  |  |-- members
|  |  |  |-- member
|  |  |  |  |- type
|  |  |  |  |- role
|  |  |  |  |- uri
|  |  |  |  |-- allowedContainers
|  |  |  |  |  |- type
|  |  |  |  |  |-- container
|  |  |  |  |  |  |- id
|  |  |  |  |-- defaultContainer
|  |  |  |  |  |- id
|  |  |  |  |-- occurrenceConstraint
|  |  |  |  |  |- type

```

```

|-- sourceNetworks
    |-- sourceNetwork
        |- type
|-- resolutionRules
    |-- resolutionRule
        |- type
        |- role
|-- blockRules
    |-- blockRule
        |- type
        |-- containers
            |- type
            |-- container
                |- id
        |-- sourceNetworks
            |-- sourceNetwork
                |- type
        |-- minAvailability
        |-- maxAvailability
|-- rolePrecedence
    |-- rolePrecedenceEntry
        |- type

```

The following XSD schema fragment defines the requirements to which a **privacyPublicationGrammar provisionGroup** element XML document SHOULD [<120>](#) conform.

```

<?xml version="1.0" encoding="utf-8"?>

<xs:schema xmlns="http://schemas.microsoft.com/2006/09/sip/provisiongroupulist-notification"
    targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisiongroupulist-
notification" elementFormDefault="qualified"
    xmlns:xs="http://www.w3.org/2001/XMLSchema"
    xmlns:ms="urn:microsoft-cpp-xml-serializer">

    <xs:complexType name="ProvisionGroupType">
        <xs:all>
            <xs:any namespace="##any" processContents="lax" minOccurs="0"
maxOccurs="unbounded">
                <xs:annotation>
                    <xs:appinfo>
                        </xs:appinfo>
                    </xs:annotation>
                </xs:any>
            </xs:all>
            <xs:attribute name="name" fixed="privacyPublicationGrammar" use="required"/>
            <xs:anyAttribute namespace="##other" processContents="lax"/>
        </xs:complexType>
        <xs:element name="provisionGroup" type="ProvisionGroupType" />
    </xs:schema>

```

The **privacyPublicationGrammar provisionGroup** element contains the **containerManifestList** XML document, as shown in the example in section [4.2.2](#).

The following XSD schema fragment defines the requirements to which the **containerManifestList** XML document SHOULD [<121>](#) conform.

```

<?xml version="1.0" encoding="utf-8"?>
<xs:schema
    targetNamespace="http://schemas.microsoft.com/2008/09/sip/ContainerManifest"
    xmlns:tns="http://schemas.microsoft.com/2008/09/sip/ContainerManifest"

```

```

xmlns:xs="http://www.w3.org/2001/XMLSchema"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"
attributeFormDefault="unqualified"
elementFormDefault="qualified">

<!--
Client should not proceed with the parsing if its namespace doesn't
match to the namespace of the xml.
Change in namespace is an indication that format is not compatible.
-->

<xs:import namespace="http://schemas.microsoft.com/2006/09/sip/commontypes"
  schemaLocation="commontypes.xsd"/>

<xs:simpleType name="sourceNetworkTypeEnum">
  <xs:restriction base="xs:token">
    <xs:enumeration value="sameEnterprise"/>
    <xs:enumeration value="federated"/>
    <xs:enumeration value="publicCloud"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="sourceNetworkTypeEnumEx">
  <xs:union memberTypes="tns:sourceNetworkTypeEnum xs:token" />
</xs:simpleType>

<xs:complexType name="sourceNetworkType">
  <xs:sequence>
    <xs:sequence minOccurs="0" maxOccurs="1">
      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:element ref="ct:end"/>
    </xs:sequence>
    <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="type" type="tns:sourceNetworkTypeEnumEx" use="required" />
  <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="sourceNetworksType">
  <xs:sequence>
    <xs:element name="sourceNetwork" type="tns:sourceNetworkType" minOccurs="1"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!--
Container attributes.
UI will show all visible containers in the access level management view.
-->
<xs:complexType name="containerType">
  <xs:sequence>
    <xs:element name="visible" type="xs:boolean" minOccurs="0" maxOccurs="1"/>
    <xs:element name="avoidRedundantMembers" type="xs:boolean" minOccurs="0"
maxOccurs="1"/>
    <xs:element name="blocked" type="xs:boolean" default="false" minOccurs="0"
maxOccurs="1"/>

    <xs:sequence minOccurs="0" maxOccurs="1">
      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:element ref="ct:end"/>
    </xs:sequence>
  </xs:sequence>
</xs:complexType>

```

```

        <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>

    </xs:sequence>
    <xs:attribute name="id" type="xs:unsignedInt" />
    <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="containerIdType">
    <xs:sequence>

        <xs:sequence minOccurs="0" maxOccurs="1">
            <xs:sequence minOccurs="0" maxOccurs="unbounded">
                <xs:element ref="ct:delimiter"/>
                <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:element ref="ct:end"/>
        </xs:sequence>
        <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>

    </xs:sequence>
    <xs:attribute name="id" type="xs:unsignedInt" />
    <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="containersType">
    <xs:sequence>
        <xs:element name="container" type="tns:containerType" minOccurs="1"
maxOccurs="unbounded"/>

        <xs:sequence minOccurs="0" maxOccurs="1">
            <xs:sequence minOccurs="0" maxOccurs="unbounded">
                <xs:element ref="ct:delimiter"/>
                <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:element ref="ct:end"/>
        </xs:sequence>
        <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>
    <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:simpleType name="allowedContainersTypeEnum">
    <xs:restriction base="xs:token">
        <xs:enumeration value="visibleContainers"/>
        <xs:enumeration value="custom"/>
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="allowedContainersTypeEnumEx">
    <xs:union memberTypes="tns:allowedContainersTypeEnum xs:token" />
</xs:simpleType>

<xs:complexType name="allowedContainersType">
    <xs:sequence>
        <xs:element name="container" type="tns:containerIdType" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:attribute name="type" type="tns:allowedContainersTypeEnumEx"/>
    <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:simpleType name="occurrenceTypeEnum">
    <xs:restriction base="xs:token">
        <xs:enumeration value="zero"/>
        <xs:enumeration value="one"/>
        <xs:enumeration value="zeroOrOne"/>
    </xs:restriction>

```

```

</xs:simpleType>

<xs:simpleType name="occurrenceTypeEnumEx">
  <xs:union memberTypes="tns:occurrenceTypeEnum xs:token" />
</xs:simpleType>

<xs:complexType name="occurrenceConstraintType">
  <xs:sequence>
    <xs:sequence minOccurs="0" maxOccurs="1">
      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:element ref="ct:end"/>
    </xs:sequence>
    <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>
  </xs:sequence>
  <xs:attribute name="type" type="tns:occurrenceTypeEnumEx" use="optional" />
  <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:simpleType name="roleTypeEnum">
  <xs:restriction base="xs:token">
    <xs:enumeration value="delegate"/>
    <xs:enumeration value="buddy"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="roleTypeEnumEx">
  <xs:union memberTypes="tns:roleTypeEnum xs:token" />
</xs:simpleType>

<!--
Order of precedence for which role applies if
entity maps to multiple (for example, buddy and delegate)
-->
<xs:complexType name="rolePrecedenceEntryType">
  <xs:sequence>
    <xs:sequence minOccurs="0" maxOccurs="1">
      <xs:element ref="ct:delimiter"/>
      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:element ref="ct:end"/>
    </xs:sequence>
    <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>
  </xs:sequence>

  <xs:attribute name="type" type="tns:roleTypeEnumEx" use="required"/>
  <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="rolePrecedenceType">
  <xs:sequence>
    <xs:element name="rolePrecedenceEntry" type="tns:rolePrecedenceEntryType"
minOccurs="1" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<!--
Defines resolution action to be (implicitly) taken if
memberType constraints have been violated
-->
<xs:simpleType name="resolutionTypeEnum">
  <xs:restriction base="xs:token">

```

```

        <xs:enumeration value="removeFromContainer"/>
        <xs:enumeration value="moveToDefaultContainer"/>
        <xs:enumeration value="forceRole"/>
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="resolutionTypeEnumEx">
    <xs:union memberTypes="tns:resolutionTypeEnum xs:token" />
</xs:simpleType>

<xs:complexType name="resolutionRuleType">
    <xs:sequence>
        <xs:sequence minOccurs="0" maxOccurs="1">
            <xs:element ref="ct:delimiter"/>
            <xs:sequence minOccurs="0" maxOccurs="unbounded">
                <xs:element ref="ct:delimiter"/>
                <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
            </xs:sequence>
            <xs:element ref="ct:end"/>
        </xs:sequence>
        <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>
    </xs:sequence>

    <xs:attribute name="type" type="tns:resolutionTypeEnumEx" use="required"/>
    <xs:attribute name="role" type="tns:roleTypeEnumEx" use="optional"/>
    <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="resolutionRulesType">
    <xs:sequence>
        <xs:element name="resolutionRule" type="tns:resolutionRuleType" minOccurs="1"
maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>

<xs:simpleType name="memberTypeEnum">
    <xs:restriction base="xs:token">
        <xs:enumeration value="user"/>
        <xs:enumeration value="domain"/>
        <xs:enumeration value="sameEnterprise"/>
        <xs:enumeration value="federated"/>
        <xs:enumeration value="publicCloud"/>
        <xs:enumeration value="everyone"/>
        <xs:enumeration value="singleIdentityUser"/>
    </xs:restriction>
</xs:simpleType>

<xs:simpleType name="memberTypeEnumEx">
    <xs:union memberTypes="tns:memberTypeEnum xs:token" />
</xs:simpleType>

<xs:complexType name="memberType">
    <xs:sequence>

        <!--
        Allowed containers for the member.
        -->
        <xs:element name="allowedContainers" type="tns:allowedContainersType"
            minOccurs="1" maxOccurs="1"/>

        <!--
        Default container for the member.
        -->
        <xs:element name="defaultContainer" type="tns:containerIdType"
            minOccurs="0" maxOccurs="1"/>

        <!--
        Occurrence constraint on the member. There can be only one such constraint.

```


Constraint check will be applied on all visible containers.
 If constraint fails, member will be deleted from all containers and placed in the default container.

```

-->
<xs:element name="occurrenceConstraint" type="tns:occurrenceConstraintType"
  minOccurs="0" maxOccurs="1"/>

<xs:sequence minOccurs="0" maxOccurs="1">
  <!-- W14 extension -->
  <xs:element ref="ct:delimiter"/>
  <xs:element name="sourceNetworks" type="tns:sourceNetworksType" minOccurs="0"
maxOccurs="1"/>
  <xs:element name="resolutionRules" type="tns:resolutionRulesType" minOccurs="0"
maxOccurs="1"/>

  <xs:sequence minOccurs="0" maxOccurs="unbounded">
    <xs:element ref="ct:delimiter"/>
    <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
  </xs:sequence>
  <xs:element ref="ct:end"/>
</xs:sequence>
<xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>

</xs:sequence>
<xs:attribute name="type" type="tns:memberTypeEnumEx" use="optional" />

<!--
If role is present, client should identify all the members of that role and
enforce occurrence constraint on those members.
-->
<xs:attribute name="role" type="tns:roleTypeEnumEx" use="optional" />

<xs:attribute name="uri" type="xs:anyURI" use="optional" />
<xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="membersType">
  <xs:sequence>
    <xs:element name="member" type="tns:memberType" minOccurs="1"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:simpleType name="unboundedType">
  <xs:union memberTypes="xs:unsignedInt">
    <xs:simpleType>
      <xs:restriction base="xs:token">
        <xs:enumeration value="unbounded"/>
      </xs:restriction>
    </xs:simpleType>
  </xs:union>
</xs:simpleType>

<xs:simpleType name="blockTypeEnum">
  <xs:restriction base="xs:token">
    <xs:enumeration value="invites"/>
    <xs:enumeration value="subscriberPrompt"/>
  </xs:restriction>
</xs:simpleType>

<xs:simpleType name="blockTypeEnumEx">
  <xs:union memberTypes="tns:blockTypeEnum xs:token" />
</xs:simpleType>

<!--
Rule defining the blocking actions expected to be performed by clients.
-->
<xs:complexType name="blockRuleType">

```

```

<xs:sequence>

  <!--
  Containers to which this rule apply.
  -->
  <xs:element name="containers" type="tns:allowedContainersType" minOccurs="0"
maxOccurs="1"/>

  <!--
  Source networks to which this rule apply.
  -->
  <xs:element name="sourceNetworks" type="tns:sourceNetworksType" minOccurs="0"
maxOccurs="1"/>

  <!--
  Availability range to which this rule apply.
  -->
  <xs:element name="maxAvailability" type="tns:unboundedType" default="unbounded"
minOccurs="0" maxOccurs="1"/>
  <xs:element name="minAvailability" type="tns:unboundedType" default="0"
minOccurs="0" maxOccurs="1"/>

  <xs:sequence minOccurs="0" maxOccurs="1">
    <xs:element ref="ct:delimiter"/>
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <xs:element ref="ct:delimiter"/>
      <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
    </xs:sequence>
    <xs:element ref="ct:end"/>
  </xs:sequence>
  <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>

</xs:sequence>
<xs:attribute name="type" type="tns:blockTypeEnumEx" use="required" />

<xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="blockRulesType">
  <xs:sequence>
    <xs:element name="blockRule" type="tns:blockRuleType" minOccurs="1"
maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>

<xs:complexType name="containerManifestType">
  <xs:sequence>
    <xs:element name="containers" type="tns:containersType" minOccurs="0"
maxOccurs="1"/>
    <xs:element name="members" type="tns:membersType" minOccurs="0" maxOccurs="1"/>

    <xs:sequence minOccurs="0" maxOccurs="1">
      <!-- W14 extension -->
      <xs:element ref="ct:delimiter"/>
      <xs:element name="blockRules" type="tns:blockRulesType" minOccurs="0"
maxOccurs="1"/>
      <xs:element name="rolePrecedence" type="tns:rolePrecedenceType" minOccurs="0"
maxOccurs="1"/>

      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
    </xs:sequence>
  </xs:complexType>
  <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>

```

```

    </xs:sequence>
    <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:complexType name="containerManifestListType">
  <xs:sequence>
    <xs:element name="containerManifest" type="tns:containerManifestType" minOccurs="1"
maxOccurs="1"/>

    <xs:sequence minOccurs="0" maxOccurs="1">
      <xs:sequence minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="ct:delimiter"/>
        <xs:any namespace="##targetNamespace" processContents="lax" minOccurs="0"
maxOccurs="unbounded"/>
      </xs:sequence>
      <xs:element ref="ct:end"/>
    </xs:sequence>
    <xs:element ref="ct:extension" minOccurs="0" maxOccurs="1"/>
  </xs:sequence>

  <!--
  Client should not proceed with the parsing if its version is less than
  the version in the xml. Version mismatch is an indication that
  format is not compatible.
  Missing minSupportClientVersion will be treated as 0.0.0.0, that is, always
  compatible.
  -->
  <xs:attribute name="minSupportedClientVersion" type="xs:token" use="optional"/>
  <xs:attribute name="majorVersion" type="xs:unsignedInt" use="optional"/>
  <xs:attribute name="minorVersion" type="xs:unsignedInt" use="optional"/>

  <xs:anyAttribute processContents="lax"/>
</xs:complexType>

<xs:element name="containerManifestList" type="tns:containerManifestListType">
</xs:element>
</xs:schema>

```

The following XSD schema fragment defines the schema for **commontypes** referred by the **containerManifestList** XSD schema.

```

<?xml version="1.0" encoding="UTF-8"?>
<xs:schema
  targetNamespace="http://schemas.microsoft.com/2006/09/sip/commontypes"
  xmlns:tns="http://schemas.microsoft.com/2006/09/sip/commontypes"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
  elementFormDefault="qualified"
  attributeFormDefault="unqualified"
  >

  <xs:element name="delimiter">
    <xs:complexType />
  </xs:element>

  <xs:element name="end">
    <xs:complexType />
  </xs:element>

  <xs:complexType name="extensionType">
    <xs:sequence>
      <xs:any processContents="lax" minOccurs="1"
        maxOccurs="unbounded" namespace="##other"/>
    </xs:sequence>
  </xs:complexType>

```

```
<xs:element name="extension" type="tns:extensionType"/>
</xs:schema>
```

containerManifestList: A list of all **containerManifests** available.

containerManifest: A manifest that provides the list of containers and members. The UAC SHOULD create containers and its members based on the constraints defined in the manifest.

majorVersion (optional): Ignored.

minorVersion (optional): Ignored.

minSupportedClientVersion (optional): The minimum version of the UAC that can use the container manifest.

containers: The wrapper for the list of **container** elements.

container: Each **container** element describes a container.

- **id (required):** This unsigned short integer uniquely identifies a container.
- **Visible (optional):** Container attribute which, when "true", designates the container as a member of the set of containers identified with allowed containers **type** equal to "visibleContainers". The UAC also interprets this attribute as those containers that are displayed in its access level view.
- **avoidRedundantMembers (optional):** This **string** indicates whether a container MAY have redundant membership. For example, redundancy occurs when a domain, as well as a user who is in that domain, are associated with a container. The value MUST be "true" or "false". The UAC SHOULD clean up redundant members from containers if the setting is "true".
- **blocked (optional):** Ignored.

members: The wrapper for the list of **member** elements.

member: Describes a container member.

type (optional): This string value specifies the type of the member. The following values are defined:

- **user:** A SIP URI.
- **domain:** A SIP domain.
- **sameEnterprise:** All users from the same enterprise.
- **federated:** All federated users.
- **publicCloud:** All public cloud users.
- **everyone:** All users.
- **singleIdentityUser:** User who has just one SIP uri for that particular identity. PIC contacts that have more than one SIP uri that belong to the identity should not be processed using this rule.

role (optional): This string value specifies the role of the member. Note that container members matching to a member by **role** attribute take precedence over the corresponding match to member by **type** attribute. The following values are defined:

- **delegate:** A user entity currently configured as a delegate for the local user.

- **buddy:** A user entity currently a member of the local user's buddy list.
allowedContainers: The list of containers to which the member MAY be added.
 - **type (required):** A string that describes the access control element. The following values are defined:
 - **custom:** The list of containers is specified within the **allowedContainers** element.
 - **visibleContainers:** The list of containers for the member are the same as the subset of child elements of the containers element with the **Visible** element set to "true".
 - **container (optional):** The container to which the member MAY be added. There MAY be more than one element. The list of containers is specified when the **type** is "custom."
 - **id (required):** This unsigned short integer uniquely identifies a container.
defaultContainer (optional): The default container to which the member MAY be added if the constraint is not satisfied for the member.
 - **id (required):** An unsigned integer that is the container identifier.
occurrenceConstraint (optional): The mandatory membership occurrence rate the related type or role the member is present in.
 - **type (required):** A string that gives the number of occurrences mandated for the given type or role in a given container. The value MUST be "zero", "one", or "zeroOrOne".
 - When "Zero", a given container MUST have zero members of the related type or role. This value SHOULD be ignored by a UAC.
 - When "One", a given container MUST have one member of the related type or role.
 - When "zeroOrOne", a given container MAY have no members of the related type or role or it MAY have one member of a related type or role. It MUST NOT have more than one member of a related type or role.
- sourceNetworks (optional):** A wrapper for the optional list of source networks to which the member belongs to satisfy the constraint.
- sourceNetwork:** Describes a source network.
- **type (required):** A string that describes the source network. The following values are defined:
 - **sameEnterprise:** All users from the same enterprise.
 - **federated:** All federated users.
 - **publicCloud:** All public cloud users.
- resolutionRules (optional):** A wrapper for the optional list of resolution rules that are applied if the constraint is not satisfied for the member.
- resolutionRule:** Describes a resolution rule.
- **type (required):** The **type** attribute is a string that describes the resolution rule. The following values are defined:
 - **removeFromContainer:** The member is removed from the container if the constraint is not satisfied for the member.

- **moveToDefaultContainer:** The member is moved to the default container if the constraint is not satisfied for the member.
- **forceRole:** The member is forced into the role specified by the **role** element.
- **role (optional):** A string that describes the role the member is forced into when the constraint is not satisfied for the member. This attribute is set when the **type** attribute has the value "forceRole". Possible values for the role attribute are "delegate" and "buddy".

blockRules (optional): The wrapper for the list of **blockRule** elements.

blockRule: Describes a **blockRule** element.

- **type (required):** The **type** attribute is a string that describes the actions that are blocked. The following values are defined:
 - **invites:** Invites are blocked when the rule matches.
 - **subscriberPrompt:** Subscriber prompts are blocked when the rule matches.
- **containers (optional):** The actions from the members of specified containers are blocked.
- **sourceNetworks (optional):** The actions from the members of specified source networks are blocked.
- **maxAvailability (optional):** Defines the upper bound of the availability range of the publisher for which this rule applies.
- **minAvailability (optional):** Defines the lower bound of the availability range of the publisher for which this rule applies.

rolePrecedence (optional): The wrapper for the list of **rolePrecedenceEntry** elements that defines the order of precedence for roles when a member maps to multiple roles.

rolePrecedenceEntry: Describes a role.

- **type (required):** A string that specifies a role. The possible values are "delegate" and "buddy".

For a detailed example, see section 4.2.2.

2.2.2.5.11 Data Model for persistentChatConfiguration provisionGroup

```
provisionGroup (name='persistentChatConfiguration')
```

```
|
|-- propertyEntryList
|   |-- property
|   |   |-- name
```

The following XSD schema fragment defines the requirements to which a **persistentChatConfiguration provisionGroup** element XML document SHOULD conform. [<122>](#)

```
<?xml version="1.0" encoding="utf-16"?>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2006/09/sip/provisioningrouplist-
notification" targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisioningrouplist-
notification" xmlns:xs="http://www.w3.org/2001/XMLSchema">

  <xs:complexType name="propertyEntryType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute name="name" type="xs:string" use="required" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
```

```

        </xs:extension>
    </xs:simpleContent>
</xs:complexType>
<xs:complexType name="propertyEntryListType">
    <xs:sequence>
        <xs:element name="property" type="tns:propertyEntryType" maxOccurs="unbounded"/>
    </xs:sequence>
</xs:complexType>
<xs:element name="propertyEntryList" type="tns:propertyEntryListType" />
</xs:schema>

```

DefaultPersistentChatPoolUri: Specifies the default persistent chat pool URI for the client to connect to.

EnablePersistentChat: Specifies whether persistent chat is enabled for the client. The value MUST be "true" or "false".

PersistentChatPoolUris: Specifies the URIs of all the persistent chat pools that the client can connect to, separated by a single whitespace.

PersistentChatWebManagerUriInt: Specifies the internally accessible URI that serves the chat room management Web application. [<123>](#) This URI SHOULD be used by the client to connect to the Web application when signed on from inside the network.

PersistentChatWebManagerUriExt: Specifies the externally accessible URI that serves the chat room management Web application. [<124>](#). This URI SHOULD be used by the client to connect to the Web application when signed on from outside the network.

For a detailed example, see section [4.2.2](#).

2.2.2.5.12 Data Model for publicProviders provisionGroup

```

provisionGroup (name='publicProviders')
|
|-- propertyEntryList
|   |-- property[name="publicProviders"]
|   |   |-- CDATA
|   |   |   |-- PublicProviders
|   |   |   |   |-- PublicProvider
|   |   |   |   |   - Name
|   |   |   |   |   - ProxyFqdn
|   |   |   |   |   - IconUrl
|   |   |   |   |   - NameDecorationDomain
|   |   |   |   |   - NameDecorationRoutingDomain
|   |   |   |   |   - NameDecorationExcludedDomainList

```

The following XSD schema fragment defines the requirements to which a **publicProviders provisionGroup** element XML document SHOULD conform. [<125>](#)

```

<?xml version="1.0" encoding="utf-16"?>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2006/09/sip/provisiongroup-list-notification" targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisiongroup-list-notification" xmlns:xs="http://www.w3.org/2001/XMLSchema">
    <xs:complexType name="propertyEntryType">
        <xs:simpleContent>

```

```

    <xs:extension base="xs:string">
      <xs:attribute name="name" type="xs:string" use="required" />
    </xs:extension>
  </xs:simpleContent>
</xs:complexType>
<xs:complexType name="propertyEntryListType">
  <xs:sequence>
    <xs:element name="property" type="tns:propertyEntryType" maxOccurs="unbounded"/>
  </xs:sequence>
</xs:complexType>
<xs:element name="propertyEntryList" type="tns:propertyEntryListType" />
</xs:schema>

```

The following XSD schema fragment defines the requirements to which a **publicProviders** **PublicProviders** element XML document SHOULD conform. [<126>](#)

```

<?xml version="1.0" encoding="utf-16"?>
<xs:schema xmlns:tns="http://schemas.microsoft.com/2006/09/sip/provisioningrouplist-
notification" targetNamespace="http://schemas.microsoft.com/2006/09/sip/provisioningrouplist-
notification" xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:complexType name="publicProvidersType">
    <xs:sequence>
      <xs:element name="PublicProvider" type="tns:publicProviderType" minOccurs="0"
maxOccurs="unbounded" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="publicProviderType">
    <xs:attribute name="Name" type="xs:string" use="required" />
    <xs:attribute name="ProxyFqdn" type="xs:string" use="required" />
    <xs:attribute name="IconUrl" type="xs:string" use="optional" />
    <xs:attribute name="NameDecorationDomain" type="xs:string" use="optional" />
    <xs:attribute name="NameDecorationRoutingDomain" type="xs:string" use="optional" />
    <xs:attribute name="NameDecorationExcludedDomainList" type="xs:string" use="optional" />
  </xs:complexType>
  <xs:element name="PublicProviders" type="tns:publicProvidersType" />
</xs:schema>

```

Name: The name of the public provider.

ProxyFqdn: Specifies the fully qualified domain name (FQDN) that is used for presence requests for this public provider.

IconUrl: Specifies the URL of an icon the client uses to graphically indicate the location of contacts.

NameDecorationDomain: Specifies an FQDN the client displays for the public provider.

NameDecorationRoutingDomain: Specifies an FQDN to be used to transform the user name of a public provider in the form of username@domain into username(domain)@NameDecorationRoutingDomain as its SIP URI.

NameDecorationExcludedDomainList: Specifies a comma-separated list of FQDNs. Protocol clients that attempt to route users from public providers belonging to one of the domains on this list are routed without the name transformation described in the preceding paragraph.

3 Protocol Details

3.1 Basic Registration

The SIP registrar accepts SIP REGISTER requests, processes them, and responds to them as specified in [\[RFC3261\]](#), with the additional extensions defined as follows. The basic end-to-end call-flow is shown in the following diagram.

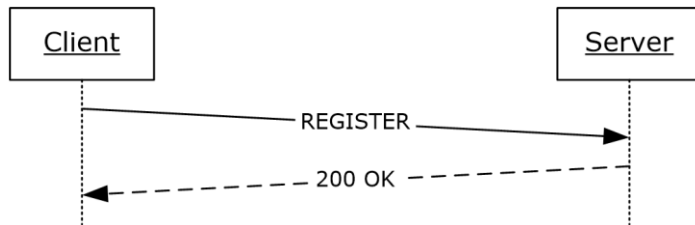


Figure 1: Basic register call flow

3.1.1 Client Role

The UAC registers and refreshes a binding with the SIP registrar as specified in [\[RFC3261\]](#) section 10. The UAC SHOULD select **endpoint identifier (EPID)** and SIP.INSTANCE endpoint identifiers, as described in [\[MS-SIPRE\]](#) section 3.2.

The UAC MUST have exactly one active REGISTER dialog as specified in [\[RFC3261\]](#) for the selected endpoint identifier (EPID).

As part of the registration process, the UAC MUST establish and maintain a **security association (SA)** with the registrar using the **NT LAN Manager (NTLM) Authentication Protocol** or the **Kerberos** authentication protocol, as specified in [\[MS-SIPAE\]](#) section 3.2.

3.1.1.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This protocol does not mandate that implementations adhere to this model as long as their external behavior is consistent with that described in this protocol.

The UAC SHOULD keep the following state:

- **Epid:** A string that is a part of the SIP endpoint identifier. The format and generation of **Epid** are described in [\[MS-SIPRE\]](#) section 3.2.3.1.
- **SIP.INSTANCE:** A string that is a part of the SIP endpoint identifier. The format and generation of **SIP.INSTANCE** are described in [\[MS-SIPRE\]](#) section 3.2.
- **GRUU:** A string that is part of the SIP endpoint identifier. It is generated by the SIP registrar and persisted on the client. The format and generation of **GRUU** are described in [\[MS-SIPRE\]](#) section 3.4.
- **Presence-Service-Available:<127>** A Boolean that is generated by the registrar and persisted on the client, and is updated as described in section [3.1.1.5.1](#).
- **Primary-Cluster-Type:<128>** An identifier associated with the UAC that is generated by the registrar and persisted on the client, and is updated as described in section [3.1.1.5.1](#).

- **Is-Connected-to-Primary:**[<129>](#) A Boolean flag associated with the UAC that is generated by the registrar and persisted on the client, and is updated as described in section 3.1.1.5.1.

3.1.1.2 Timers

The UAC SHOULD start and maintain a SIP registration refresh timer, as specified in [\[RFC3261\]](#) section 10.2.4.

If the UAC specifies support for **ms-keep-alive**, as specified in [\[MS-CONMGMT\]](#) section 2.2.1, it SHOULD maintain a whitespace keep-alive timer for sending whitespaces, as specified in [\[MS-CONMGMT\]](#) section 2.2.2.

3.1.1.3 Initialization

The UAC instance MUST generate an EPID and a **SIP.INSTANCE** value before sending any REGISTER request, as described in [\[MS-SIPRE\]](#) section 3.3.3.1.

3.1.1.4 Higher-Layer Triggered Events

Except as specified in the following subsections, processing rules follow [\[RFC3261\]](#) section 10.1.

3.1.1.4.1 Constructing the Outgoing SIP REGISTER Request

The UAC MUST construct the outgoing SIP REGISTER request, as specified in section [2.2.1.1](#).

It MUST then insert the **Epid** and **SIP.INSTANCE** values in the REGISTER request, as specified in [\[MS-SIPRE\]](#) section 3.2 and [\[MS-SIPRE\]](#) section 3.2.

The UAC SHOULD insert a **User-Agent** header as specified in section [3.4.1.5.1](#).

The UAC SHOULD discover its subnet address and insert an **Ms-Subnet** header. The UAC MUST populate the **Ms-Subnet** header by discovering the IP Subnet Address, as specified in [\[RFC950\]](#), of the TCP connection over which the SIP REGISTER Request will be sent.

The UAC SHOULD insert a **Ms-Device-Info** header as specified in section [2.2.1.7](#). The UAC MUST populate the **Ms-Device-Info** header by discovering the following:

- The MAC address, which is the physical address of the network interface over which the SIP REGISTER request will be sent.
- The version, which is a literal string that describes the version of the UAC.
- The vendor identifier, which is a literal string that identifies the manufacturer of the UAC.

It MUST then send the SIP REGISTER request and process the response as specified in the following subsection.

3.1.1.5 Message Processing Events and Sequencing Rules

Except as specified in the following subsections, the processing rules follow the procedures specified in [\[RFC3261\]](#) section 10.2.7.

3.1.1.5.1 Processing the SIP REGISTER Response

The UAC SHOULD extract the contact **GRUU** returned by the SIP registrar and use it subsequently as specified in [\[IETF DRAFT-OUGRUAUSIP-10\]](#) section 8.

The UAC MUST retrieve the registration expiry timer value, as specified in [\[RFC3261\]](#) section 10, from the SIP REGISTER response. It MUST then set the registration expiry timer value less than the value specified in the 200 OK response to the REGISTER request.

The UAC SHOULD process the **ms-keep-alive** header, as specified in [\[MS-CONMGMT\]](#) section 2.2.1, returned by the registrar as specified in [\[MS-CONMGMT\]](#) section 3.4.5, and use it to initialize the keep-alive timer.

If a **Presence-State** header field is present in the 200 success response, the UAC MUST take the following additional actions:

- If the value of **Register-Action** is "added" or "fixed", and if the REGISTER operation is for the purpose of refreshing a binding instead of adding a new binding, the UAC MUST treat this as a registration failure and all existing **dialogs** SHOULD be considered terminated. The UAC SHOULD restart registration.
- If the value of **Primary-Cluster-Type** is "central", the UAC SHOULD [<130>](#) treat this as an indication of the user being provisioned on a server at a central site and MUST update the **Primary-Cluster-Type** field to "central". If the value is "remote", the UAC MUST treat this as an indication that the user is being provisioned at a remote site and MUST update the **Primary-Cluster-Type** field to "remote".
- If the value of **Is-Connected-to-Primary** is "yes", the UAC SHOULD [<131>](#) treat the endpoint as being connected to the user's primary server and MUST update the **Is-Connected-to-Primary** field to "true". If the value is "no", the UAC MUST [<132>](#) treat the endpoint as not being connected to the user's primary server and MUST update the **Is-Connected-to-Primary** field to "false".
- If the value of **user-services-state** is "unavailable", the UAC SHOULD [<133>](#) set the **Presence-Service-Available** field to "false".
- If **user-services-state** is NOT present, the UAC SHOULD [<134>](#) set the **Presence-Service-Available** field to "true".

After the UAC receives a 200 response for the first REGISTER it sent, the UAC SHOULD attempt to establish a contact or group subscription, as specified in [\[MS-SIP\]](#) section 3.3, and SHOULD attempt to establish a self-subscription, as specified in [\[MS-PRES\]](#) section 3.3, to retrieve the user's contact and group information and published presence information.

If the **Presence-Service-Available** field is set to "false", the UAC SHOULD [<135>](#) treat itself to be in survivable mode and SHOULD NOT attempt to establish any subscription except a self-subscription, as specified in [\[MS-PRES\]](#) section 3.3.

If the self-subscription fails to establish, the UAC is not able to retrieve the published presence information, and the UAC SHOULD [<136>](#) remove the binding by sending a REGISTER request with **Expiry** value of 0, as specified in [\[RFC3261\]](#) section 10.2.2. If the contact or group subscription fails, contacts and groups information is not available.

If the self-subscription fails to establish and the UAC is not in survivable mode, the UAC SHOULD [<137>](#) retry the self-subscription at a randomized interval. If, however, the self-subscription fails to establish while the UAC is in survivable mode, it SHOULD [<138>](#) wait for the retry until the end of survivable mode, as indicated by a response to a future REGISTER with the **Presence-Service-Available** field set to "false", as described in the preceding paragraphs.

This protocol does not mandate how implementations operate on the **Primary-Cluster-Type** and **Is-Connected-to-Primary** fields as long as their external behavior is consistent with that described in this protocol.

3.1.1.6 Timer Events

The UAC SHOULD refresh the registration whenever the registration expiry timer fires, as specified in [\[RFC3261\]](#) section 10.2.4. The UAC MUST use the same TCP connection that it sent the dialog-creating REGISTER request through for sending registration refresh requests.

The UAC SHOULD periodically send whitespaces whenever the keep-alive timer fires, as specified in [\[MS-CONMGMT\]](#) section 3.4.5.4.

3.1.1.7 Other Local Events

None.

3.1.2 Server Role

3.1.2.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This protocol does not mandate that implementations adhere to this model, as long as their external behavior is consistent with that described in this protocol.

The registrar SHOULD keep the following states for each UAC:

- **SIP Endpoint identifier:** An identifier for a UAC. It MAY be a structure with the combination of an EPID, as described in [\[MS-SIPRE\]](#) section 3.2, a **SIP.INSTANCE**, as described in [\[MS-SIPRE\]](#) section 3.2, and a GRUU, as described in [\[MS-SIPRE\]](#) section 3.4.
- **Routable:** A Boolean flag that is set to "true" whenever a valid registration is associated with the UAC, and is set to "false" whenever the registration associated with the UAC ceases to exist.
- **Presence-Service-Available:** [<139>](#) A Boolean flag that is set to "true" when the presence service associated with the user is available, and is set to "false" whenever the presence service associated with the user is unavailable.
- **Primary-Cluster-Type:** [<140>](#) An identifier for the user associated with the UAC that is set to "central" if the user's home server is located in a central site, and is set to "remote" if the user's home server is located in a remote site.
- **Is-Connected-to-Primary:** [<141>](#) A Boolean flag that is set to "true" when the server accepting the registration is the user's home server, and is set to "false" when the server accepting the registration is not the user's home server.
- **Supports survivable mode:** [<142>](#) A Boolean flag that is set to "true" when the UAC indicates support for registration without the availability of the presence component, as defined in section [2.2.1.5](#), and is set to "false" when the UAC does not support registration without the availability of the presence component.

3.1.2.2 Timers

The SIP registrar SHOULD keep a registration timer for each binding. The registration expiry timer value is exchanged as specified in [\[RFC3261\]](#) section 10. The registrar MUST specify an expiry value for the binding of no less than 30 seconds. When the registrar receives a REGISTER request for a binding, the registrar MUST refresh the registration expiration timer for the associated binding.

3.1.2.3 Initialization

None.

3.1.2.4 Higher-Layer Triggered Events

The message validation rules for processing an incoming REGISTER request follow the procedures specified in [\[RFC3261\]](#) section 10.3 as well as [\[MS-SIPRE\]](#) sections 3.4 and 3.5.

3.1.2.5 Message Processing Events and Sequencing Rules

Except as specified in the following subsections, the processing rules follow the procedures specified in [\[RFC3261\]](#) section 10.3 and [\[IETF DRAFT-OUGRUAUSIP-10\]](#) section 7.1.2.1.

3.1.2.5.1 Processing the REGISTER Request

The SIP registrar MUST establish and maintain an SA with the UAC using the NTLM or the Kerberos authentication protocol, as specified in [\[MS-SIPAE\]](#) section 3.2.

When the SA is established, the registrar MUST validate the request according to the rules specified in section [2.2.1.1](#).

If the validation fails after the registrar inspects the REGISTER request headers, as specified in [\[RFC3261\]](#) section 8.2.2, the registrar SHOULD reject the request with an appropriate error response, as specified in [\[RFC3261\]](#) section 8.2.2. The outgoing SIP error response SHOULD include an **ms-diagnostics header** or an ms-diagnostics-public header, as described in [\[MS-OCER\]](#) section 2.2.1. The **ErrorId** to be included in these headers is also listed in the following table.

Special error responses are shown in the following table.

SIP response code	ErrorId	Reason
400	4010	Missing endpoint identifier (Epid and SIP.INSTANCE).
489	4055	Event header is not set to "registration".
421	2057	Supported header does not have gruu-10 , but msrtc-event-categories is present.

The registrar retrieves the **Epid** and **SIP.INSTANCE** value from the REGISTER request. The registrar tries to find an existing SIP endpoint with an identifier that contains an identical **Epid** and **SIP.INSTANCE**. If none is found, this is a new binding for a new UAC. The registrar generates a GRUU as described in [\[MS-SIPRE\]](#) section 2.2.3, and stores the EPID, **SIP.INSTANCE**, and GRUU in a new SIP endpoint identifier. It then processes the request and, if the request is accepted, it SHOULD set the **Routeable** field of the SIP endpoint identifier to "true".

If the registrar finds an existing SIP endpoint identifier, this is either a REGISTER refresh or a new REGISTER for the same endpoint. In either case, the registrar processes the request as specified in [\[RFC3261\]](#) section 10.3. If it accepts the request, it SHOULD set the **Routeable** field of the SIP endpoint identifier to "true".

The server replaces bindings using **SIP.INSTANCE** as the primary key.

If the **Supports survivable mode** field for the UAC is set to "false" and **Presence Service Available** is "false", the registrar SHOULD [<143>](#) reject the request with a 503 Service Unavailable, as specified in [\[RFC3261\]](#) Section 21.5.4. The **ms-diagnostics-public ErrorId** SHOULD be set to "4164".

If the UAC supports survivable mode, the **registrar** SHOULD<144> notify the endpoint if the **Presence-Service-Available** transitions from "true" to "false" using a NOTIFY with the **userservices-unavailable** event, as defined in section [2.2.1.13](#).

The registrar SHOULD<145> retrieve the MAC address, version and vendor identifier from the **Ms-Device-Info** header in the REGISTER request. The information stored in the **Ms-Device-Info** header SHOULD be used by the server to track the devices that are communicating with it.

The 200 OK response for the REGISTER request is generated as specified in [RFC3261] section 10.3, and the following additional rules:

- The registrar retrieves the GRUU from the SIP endpoint identifier and returns the GRUU as specified in [\[IETF DRAFT-OUGRUAUSIP-10\]](#) section 8.2 in the response to the REGISTER request.
- In addition, the registrar SHOULD insert a **Presence-State** header field into the 200 response to the REGISTER request.
 - The value of the **Register-Action** field of the **Presence-State** header is generated as follows:
 - If the REGISTER created a new SIP endpoint (5), the value is "added".
 - If the REGISTER adds a new binding to an existing SIP endpoint and the **Routeable** field changed from "false" to "true" during processing, the value is "fixed".
 - If the REGISTER refreshes an existing binding for an existing SIP endpoint (5), the value is "refreshed".
 - The **Primary-Cluster-Type**<146> field of the **Presence-State** header is generated as follows.
 - If the **Primary-Cluster-Type** field is set to "central", the value is "central".
 - If the **Primary-Cluster-Type** field is set to "remote", the value is "remote".
 - The **Is-Connected-To-Primary**<147> field of the **Presence-State** header is generated as follows.
 - If the **Is-Connected-To-Primary** field is "true", the value is "yes".
 - If the **Is-Connected-To-Primary** field is "false", the value is "no".
 - The **user-services-state**<148> field of the **Presence-State** header is generated as follows.
 - The **user-services-state** field is set to "unavailable" if the **Presence-Service-State** field is set to "false".
 - The **user-services-state** field of the **Presence-State** header SHOULD NOT be set if the **Presence-Service-State** field is set to true.

The registrar MAY perform additional actions based on the version of the UAC, as described in section [3.4.2.5](#).

3.1.2.6 Timer Events

When the SIP registrar's expiry timer for a binding expires, the registrar MUST remove the binding and clear any state related to the binding. If the **routeable** field was previously set to "true", it

SHOULD be changed to "false". If the keep-alive timer specified in [\[MS-CONMGMT\]](#) section 3.4.5.2 fires, the binding SHOULD be removed.

3.1.2.7 Other Local Events

None.

3.2 Removing a Binding from the Registrar

The SIP registrar SHOULD send a deregister NOTIFY to the UAC if it becomes necessary to remove a binding prematurely for any reason. Some of the reasons when this MAY happen are illustrated by the following examples, which are described in section [2.2.1.13](#).

- The registrar enforces a quota on the number of endpoints from which a user MAY be registered. If this quota is exceeded, the registrar might use a policy of deregistering the oldest endpoint. In this case, the message body is set to "unregistered".
- The administrator disables the user for policy reasons, such as when a user leaves the organization. In such cases, the registrar might automatically deregister all the endpoints (5) of the user when the policy change is applied. The message body is set to "rejected".
- The user is being moved across servers. This is possible if the system supports the notion of a dedicated "home server" for each user. In such cases, when the user is moved across servers (2), the existing endpoints (5) in the previous server (2) might be automatically deregistered. The message body is set to "deactivated".
- The user is being moved across registrars. This is possible if the system supports the notion of a dedicated "home registrar" for each user. In such cases, when the user is moved across registrars, the existing endpoints in the previous registrar might be automatically deregistered. The message body is set to "preferred-registrar-change". The **P-Preferred-Registrar** header indicates the new home registrar for the user. [<149>](#)
- The service responsible for presence of a user is unavailable and the endpoint has not indicated support for survivable mode. In such cases, the endpoints (5) that do not indicate support are automatically deregistered. The message body is set to "userservices-unavailable". [<150>](#)

Note: These reasons are illustrative and the list of examples is not comprehensive.

When the registrar deregisters an endpoint (5), it SHOULD send a deregister NOTIFY to the UAC indicating that the registration has terminated.

3.2.1 Client Role

This section follows the behavior described in endnote [<151>](#).

3.2.1.1 Abstract Data Model

None.

3.2.1.2 Timers

None.

3.2.1.3 Initialization

None.

3.2.1.4 Higher-Layer Triggered Events

None.

3.2.1.5 Message Processing Events and Sequencing Rules

This section follows the behavior described in endnote [<152>](#).

No extra sequencing rules beyond those specified in [\[RFC3261\]](#) section 10.2.2 are required, except when a **P-Preferred-Registrar** header is present in the NOTIFY. If the **P-Preferred-Registrar** header is present, the UAC SHOULD attempt to register with the SIP registrar specified in the **P-Preferred-Registrar** header.

3.2.2 Server Role

3.2.2.1 Abstract Data Model

None.

3.2.2.2 Timers

None.

3.2.2.3 Initialization

None.

3.2.2.4 Higher-Layer Triggered Events

The SIP registrar SHOULD send a deregister NOTIFY to the UAC if it becomes necessary to remove a binding prematurely for any reason. If the registrar deregisters an endpoint and a NOTIFY is sent, it MUST construct the NOTIFY request as specified in the following subsection.

3.2.2.4.1 Constructing the Outgoing Deregister NOTIFY Request

The outgoing deregister NOTIFY request is first constructed using the rules specified in section [2.2.1.11](#).

It MUST be sent only to the UAC that is being deregistered, and hence it MUST use the same local and remote tags as those specified in [\[RFC3261\]](#) section 12 and the **call-id** and **local CSeq** of the last REGISTER transaction sent from the deregistered UAC, as specified in [\[RFC3265\]](#) section 3.3.4 and [\[RFC3261\]](#) section 10.3.

3.2.2.5 Message Processing Events and Sequencing Rules

The processing rules follow the procedures specified in [\[RFC3261\]](#) section 10.3, [\[RFC3265\]](#) section 4.4.8, and [\[IETF DRAFT-OUGRUAUSIP-10\]](#) section 7.1.2.

A binding SHOULD be considered terminated when the registration expiry timer or the keep-alive timer fires, as specified in section [3.1.2.6](#). In this case, the SIP registrar MAY do either of the following options:

- Expire the registration binding, but keep the SIP endpoint identifier, as defined in section [3.1.2.1](#), with routable state set to "false". This is useful when the registrar keeps additional state associated with the SIP endpoint that is independent of the binding itself. With this option, when a

subsequent re-registration is received for the expired endpoint, the **Presence-State** header SHOULD contain a **Register-Action** field with the value "fixed".

- Expire the registration binding and remove the SIP endpoint identifier, as defined in section 3.1.2.1. With this option, when a subsequent re-registration is received for the expired endpoint, the **Presence-State** header SHOULD contain a **Register-Action** field with the value "added".

3.2.2.6 Timer Events

None.

3.2.2.7 Other Local Events

None.

3.3 Obtaining Provisioning Information

3.3.1 Client Role

The UAC sends an in-band provisioning request listing all the configuration groups in which it is interested. It then waits for a response and extracts the various configuration parameters for each of the groups listed in the response.

Example call flows are given in section [4.2](#).

3.3.1.1 Abstract Data Model

None.

3.3.1.2 Timers

None.

3.3.1.3 Initialization

None.

3.3.1.4 Higher-Layer Triggered Events

Unless specified otherwise, the message processing rules follow the procedures specified in [\[RFC3265\]](#) section 4.4.6.

The **user agent** constructs an in-band provisioning request using the syntax rules given in section [2.2.2.1](#) and populates it with the body specified in section [2.2.2.4](#). It then sends the SUBSCRIBE requests using the procedures specified in [\[RFC3265\]](#) section 3.1.4.

3.3.1.5 Message Processing Events and Sequencing Rules

Depending on the provisioning groups requested, the client processes the response as described in section [2.2.2.5](#).

3.3.1.6 Timer Events

None.

3.3.1.7 Other Local Events

None.

3.3.2 Server Role

3.3.2.1 Abstract Data Model

None.

3.3.2.2 Timers

None.

3.3.2.3 Initialization

None.

3.3.2.4 Higher-Layer Triggered Events

Except as specified in the following subsection, processing rules follow the procedures specified in [\[RFC3265\]](#) section 4.4.6.

3.3.2.4.1 Processing the Incoming SUBSCRIBE Request

On receiving an in-band provisioning request, the server parses and validates the SIP headers using the rules specified in section [2.2.2.1](#). The server then validates the request body using the rules specified in section [2.2.2.4](#). If either of these validations fails, the server SHOULD reject the request with an appropriate **SIP response**, or as per the rules specified in [\[RFC3265\]](#) section 3.1.6.3 and [\[RFC3261\]](#) section 8.2.6.

The server SHOULD then extract the provisioning groups listed in the request. It SHOULD then construct the response XML, as specified in section [2.2.2.5](#), listing each supported provisioning group and the parameters within the group. All mandatory parameters MUST be populated.

The server SHOULD then generate the SUBSCRIBE response as specified in section [2.2.2.2](#).

3.3.2.5 Message Processing Events and Sequencing Rules

None.

3.3.2.6 Timer Events

None.

3.3.2.7 Other Local Events

None.

3.4 Automatically Updating Client to a Server Compatible Version

3.4.1 Client Role

The UAC provides version information to enable version checks on the server. The following sections specify how the version information is constructed by the UAC.

3.4.1.1 Abstract Data Model

None.

3.4.1.2 Timers

None.

3.4.1.3 Initialization

None.

3.4.1.4 Higher-Layer Triggered Events

None.

3.4.1.5 Message Processing Events and Sequencing Rules

3.4.1.5.1 Construction of User-Agent header

The UAC includes a **User-Agent** header when it sends a REGISTER request. The **User-Agent** header, as specified in [\[RFC3261\]](#) section 20.41, is constructed per the ABNF, as defined in [\[RFC5234\]](#), specified in [\[RFC3261\]](#) section 24.1. The ABNF is as follows:

```
User-Agent = "User-Agent" HCOLON server-val *(LWS server-val)
server-val = product / comment
product    = token [SLASH product-version]
product-version = token
```

The UAC constructs the product-version using the following ABNF:

```
product-version = Major-version "." Minor-version "." Build-number "." Qfe-number
Major-version = 1*DIGIT
Minor-version = 1*DIGIT
Build-number = 4*DIGIT
QfeNumber = 1*DIGIT
```

Following is an example of a **User-Agent** header as constructed by a UAC:

```
User-Agent: UCCAPI/3.5.6889.0 OC/3.5.6889.0
```

3.4.1.5.2 Construction of Dynamic URL

This section follows the behavior described in the product behavior note [<153>](#).

In the cases where a dynamic URL is sent back, as described in section [3.4.2.5.8](#) and section [3.4.2.5.9](#), the user agent constructs an HTTPS URL based on the **base URL** by adding the following possible parameters:

- **lang:** The language of the UAC that needs to be installed. Possible values are the values specified in the first and third columns of the table in section [3.4.1.5.3](#).
- **arch:** The architecture of the UAC that needs to be installed. Possible values are "x32" and "x64".
- **flavor:** The flavor of the UAC that needs to be installed. Possible values are the following:
 - **pm:** Per computer.
 - **pu:** Per user.
- **Build:** The type of build that needs to be installed. Possible values are the following:
 - **fre:** A free or retail build.
 - **chk:** A checked or debug build.

The following is an example of the **Warning** header that contains the base HTTPS URI:

```
Warning: 312 lcs.microsoft.com
"https://fabrikam.com/AutoUpdate/Int/Handler/update.aspx?folder=OCv3.1"
```

The following is an example of the URL a UAC might construct from the preceding base HTTPS URI to download a newer UAC:

```
Warning: 312 lcs.microsoft.com
https://fabrikam.com/AutoUpdate/Int/Handler/update.aspx?folder=OCv3.1&lang=en-us&arch=x32&flavor=pu&build=fre
```

3.4.1.5.3 Possible Values for lang Parameter

The following table provides the possible values for the **lang** parameter described in section [3.4.1.5.2](#).

Language code	Language	Language code	Language
af	Afrikaans	sq	Albanian
ar-sa	Arabic (Saudi Arabia)	ar-iq	Arabic (Iraq)
ar-eg	Arabic (Egypt)	ar-ly	Arabic (Libya)
af	Afrikaans	sq	Albanian
ar-sa	Arabic (Saudi Arabia)	ar-iq	Arabic (Iraq)
ar-eg	Arabic (Egypt)	ar-ly	Arabic (Libya)
ar-dz	Arabic (Algeria)	ar-ma	Arabic (Morocco)
ar-tn	Arabic (Tunisia)	ar-om	Arabic (Oman)
ar-ye	Arabic (Yemen)	ar-sy	Arabic (Syria)
ar-jo	Arabic (Jordan)	ar-lb	Arabic (Lebanon)
ar-kw	Arabic (Kuwait)	ar-ae	Arabic (U.A.E.)

Language code	Language	Language code	Language
ar-bh	Arabic (Bahrain)	ar-qa	Arabic (Qatar)
Eu	Basque	bg	Bulgarian
Be	Belarusian	ca	Catalan
zh-tw	Chinese (Taiwan)	zh-cn	Chinese (PRC)
zh-hk	Chinese (Hong Kong SAR)	zh-sg	Chinese (Singapore)
Hr	Croatian	cs	Czech
Da	Danish	nl	Dutch (Standard)
nl-be	Dutch (Belgium)	en	English
en-us	English (United States)	en-gb	English (United Kingdom)
en-au	English (Australia)	en-ca	English (Canada)
en-nz	English (New Zealand)	en-ie	English (Ireland)
en-za	English (South Africa)	en-jm	English (Jamaica)
En	English (Caribbean)	en-bz	English (Belize)
en-tt	English (Trinidad)	et	Estonian
Fo	Faeroese	fa	Persian
Fi	Finnish	fr	French (Standard)
fr-be	French (Belgium)	fr-ca	French (Canada)
fr-ch	French (Switzerland)	fr-lu	French (Luxembourg)
Gd	Scottish Gaelic (United Kingdom)	gd-ie	Irish (Ireland)
De	German (Standard)	de-ch	German (Switzerland)
de-at	German (Austria)	de-lu	German (Luxembourg)
de-li	German (Liechtenstein)	el	Greek
He	Hebrew	hi	Hindi
Hu	Hungarian	is	Icelandic
Id	Indonesian	it	Italian (Standard)
it-ch	Italian (Switzerland)	ja	Japanese
Ko	Korean	ko	Korean (Johab)
Lv	Latvian	lt	Lithuanian

Language code	Language	Language code	Language
Mk	Macedonian (FYROM)	ms	Malaysian
Mt	Maltese	no	Norwegian (Bokmal)
No	Norwegian (Nynorsk)	pl	Polish
pt-br	Portuguese (Brazil)	pt	Portuguese (Portugal)
Rm	Rhaeto-Romanic	ro	Romanian
ro-mo	Romanian (Republic of Moldova)	ru	Russian
ru-mo	Russian (Republic of Moldova)	sz	Sami (Lappish)
Sr	Serbian (Cyrillic)	sr	Serbian (Latin)
Sk	Slovak	sl	Slovenian
Sb	Sorbian	es	Spanish (Spain)
es-mx	Spanish (Mexico)	es-gt	Spanish (Guatemala)
es-cr	Spanish (Costa Rica)	es-pa	Spanish (Panama)
es-do	Spanish (Dominican Republic)	es-ve	Spanish (Venezuela)
es-co	Spanish (Colombia)	es-pe	Spanish (Peru)
es-ar	Spanish (Argentina)	es-ec	Spanish (Ecuador)
es-cl	Spanish (Chile)	es-uy	Spanish (Uruguay)
es-py	Spanish (Paraguay)	es-bo	Spanish (Bolivia)
es-sv	Spanish (El Salvador)	es-hn	Spanish (Honduras)
es-ni	Spanish (Nicaragua)	es-pr	Spanish (Puerto Rico)
Sx	Sutu	sv	Swedish
sv-fi	Swedish (Finland)	th	Thai
Ts	Tsonga	tn	Tswana
Tr	Turkish	uk	Ukrainian
Ur	Urdu	ve	Venda
Vi	Vietnamese	xh	Xhosa
Ji	Yiddish	zu	Zulu

3.4.1.6 Timer Events

None.

3.4.1.7 Other Local Events

None.

3.4.2 Server Role

3.4.2.1 Abstract Data Model

None.

3.4.2.2 Timers

None.

3.4.2.3 Initialization

None.

3.4.2.4 Higher-Layer Triggered Events

None.

3.4.2.5 Message Processing Events and Sequencing Rules

The server can provide a way for the administrator to configure versions of the UAC that it supports and the actions that could be taken based on the version of the UAC. The server can also provide a way to configure the location to download a newer client that implements a compatible version of a UAC.

A client is a desktop application that acts on behalf of a user as a UAC. For the purposes of this discussion, a client is a UAC. A client is identified by the UAC version specified in the **SIP request** generated by the client, as specified in section [3.4.1.5.1](#).

Based on the version of the UAC, it is possible for the administrator to configure the following:

- **Allow client:** In the case of a compatible UAC version, the server SHOULD allow the client to register with the server.
- **Allow clients with static URL:** In the case of a deprecated compatible UAC version, the server SHOULD allow the client to register with the server and provide static information to the client and network location where another version of the client can be downloaded.
- **Block client with a prompt:** In the case of an incompatible UAC version, the server (2) SHOULD reject the client with an error and include a prompt that can be displayed on the client indicating that its UAC version is incompatible with the server (2).
- **Block client with static URL:** In the case of an incompatible UAC version, the server SHOULD reject the client with an error and provide static information to the client pertaining to the location from which a compatible version of the client can be downloaded.
- **Block client with dynamic URL:** In the case of an incompatible UAC version, the server (2) SHOULD [<154>](#) reject the client with an error and provide location information to the client

indicating where to download a newer client. In this case, the client SHOULD also be able to specify which version of the UAC needs to be downloaded. For example, a language-specific version of a UAC can be automatically downloaded as a newer version of the client.

- **Block clients with upgrade:** In the case of an incompatible UAC version, the server SHOULD reject the client with an error and include a non-localized, non-displayable message in a warning header indicating that an upgrade tool needs to be used to upgrade.
- **Allow client and provide a dynamic URL to download a newer version of client:** In the case where the server supports a UAC version, but needs to recommend that the client upgrade to a client that implements a newer version of the UAC, the server SHOULD allow the UAC to register with the server and provide location information to the client where to download a newer client. In this case, the UAC SHOULD<155> also be able to specify which version of the client needs to be downloaded, for example by language, and automatically download a newer version of the client.

This is discussed in more detail in the following sections.

3.4.2.5.1 Processing Incoming REGISTER Request

The client includes the version of the UAC it is running in the format described in section 3.4.1.5 in the **User-Agent** header, as specified in [RFC3261] section 20.41, of the REGISTER request. When the server (2) receives this request from a client, it SHOULD retrieve the UAC version from the value of the **User-Agent** header. Based on the configuration on the server (2) that specifies the versions of the UAC that are compatible with the server (2), the server (2) can respond in one of the following ways:

- It SHOULD respond with a 200 OK response code, as specified in [RFC3261] section 7.2, if the UAC version is compatible with the server (2). The server (2) SHOULD include a **Warning** header, as specified in [RFC3261] section 7.3.1, which could provide more information to the client with regard to the location where a compatible version of the client could be downloaded.
- It SHOULD respond with a **403 Forbidden** response code, as specified in [RFC3261] section 7.2, if the UAC version is incompatible with the server (2). In this case, the server (2) SHOULD include a **Warning** header, as specified in [RFC3261], section 7.3.1, which could provide more information to the client with regard to the location where a compatible version of the client could be downloaded.

Following is the ABNF, as defined in [RFC5234], for a **Warning** header, as specified in [RFC3261] section 25.1.

```
Warning           = "Warning" HCOLON warning-value *(COMMA warning-value)
warning-value    = warn-code SP warn-agent SP warn-text
warn-code        = 3DIGIT
warn-agent       = hostport / pseudonym
                  ; the name or pseudonym of the server adding
                  ; the Warning header, for use in debugging
warn-text        = quoted-string
```

Sections 3.4.2.5.2 through 3.4.2.5.9 describe these response codes in more detail.

3.4.2.5.2 Sending 200 OK response (Action: Allow client)

If the request is valid and the version specified in the **User-Agent** header is supported, the server SHOULD allow processing of the REGISTER request and respond with a 200 OK REGISTER response.

3.4.2.5.3 Sending 403 Forbidden response (Action: Block client with prompt)

If the server determines that it does not support the version specified in the **User-Agent** header, and if the action that is configured for the client is to prompt, such as when there is no location configured

to download newer clients, the server SHOULD respond with a 403 Forbidden error. In this case, the server SHOULD include a **Warning** header, as specified in [\[RFC3261\]](#) section 7.3.1. The value of the **Warning** header MUST be constructed as follows:

- **Warn-code** is set to the value 310.
- **Warn-agent** is set to the value "lcs.microsoft.com".
- **Warn-text** is set to the quoted string "You are currently not using the recommended version of the client".

The following is an example of a **Warning** header:

```
Warning: 310 lcs.microsoft.com "You are currently not using the recommended version of the client"
```

3.4.2.5.4 Sending 403 Forbidden response (Action: Block client with static URL)

If the server determines that it does not support the version specified in the **User-Agent** header, it SHOULD respond with a 403 Forbidden error. If the action configured for the version of the UAC is to block and respond with a static location to download a newer client, the server SHOULD include a **Warning** header, as specified in [\[RFC3261\]](#) section 7.3.1. The value of the **Warning** header MUST be constructed as follows:

- **Warn-code** is set to the value 311.
- **Warn-agent** is set to the value "lcs.microsoft.com".
- **Warn-text** is set to the static location information that is configured for this version of the UAC. Typically, the client prompts the user to go to this location to download a newer version of the client. This information is an **HTTP** or HTTPS URI where the client version can be downloaded.

The following is an example of this **Warning** header.

```
Warning: 311 lcs.microsoft.com http://fabrikam.com
```

3.4.2.5.5 Sending 200 OK response (Action: Allow clients with static URL)

This section follows the behavior described in product behavior note [<156>](#).

If the server determines that it can support the version specified in the **User-Agent** header, it SHOULD allow processing of the REGISTER request and respond with a 200 OK response if the processing succeeds. If the action configured for the version of the UAC is to allow and respond with a static location to download a newer client, the server SHOULD include a **Warning** header, as specified in [\[RFC3261\]](#) section 7.3.1. The value of the **Warning** header MUST be constructed as follows:

- **Warn-code** is set to the value 313.
- **Warn-agent** is set to the value "lcs.microsoft.com".
- **Warn-text** is set to the static location information that is configured for this version of the UAC. Typically, the client prompts the user to go to this location to download a newer version of the client. This information is an HTTP or HTTPS URI where the client version can be downloaded.

The following is an example of this **Warning** header.

```
Warning: 313 lcs.microsoft.com http://fabrikam.com
```

3.4.2.5.6 Sending 403 Forbidden response (Action: Block clients with upgrade)

This section follows the behavior described in product behavior note [<157>](#).

If the server determines that it does not support the version specified in the **User-Agent** header, it SHOULD respond with a 403 Forbidden error. If the action configured for the version of the UAC is to block and respond with an upgrade notification hinting to the client to upgrade from an upgrade service, the server SHOULD include a **Warning** header, as specified in [\[RFC3261\]](#) section 7.3.1. The value of the **Warning** header MUST be constructed as follows:

- **Warn-code** is set to the value 314.
- **Warn-agent** is set to the value "lcs.microsoft.com".
- **Warn-text** is set to the quoted string "Check Microsoft Update for the newest version."

The following is an example of this **Warning** header.

```
Warning: 314 lcs.microsoft.com "Check Microsoft Update for the newest version."
```

3.4.2.5.7 Sending 200 OK response (Action: Allow clients with upgrade)

This section follows the behavior described in product behavior note [<158>](#).

If the server determines that it can support the version specified in the **User-Agent** header, it SHOULD allow processing of the REGISTER request and respond with a 200 OK response if the processing succeeds. If the action configured for the version of the UAC is to allow and respond with an upgrade notification hinting to the client to upgrade from an upgrade service, the server SHOULD include a **Warning** header, as specified in [\[RFC3261\]](#) section 7.3.1. The value of the **Warning** header MUST be constructed as follows:

- **Warn-code** is set to the value 314.
- **Warn-agent** is set to the value "lcs.microsoft.com".
- **Warn-text** is set to the quoted string "Check Microsoft Update for the newest version."

The following is an example of this **Warning** header.

```
Warning: 314 lcs.microsoft.com "Check Microsoft Update for the newest version."
```

3.4.2.5.8 Sending 403 Forbidden response (Action: Block client with dynamic URL)

This section follows the behavior described in product behavior note [<159>](#).

If the server determines that it does not support the version specified in the **User-Agent** header, it SHOULD respond with a 403 Forbidden error. If the action configured for the version of the client is to block and respond with a dynamic location to download a newer client, the response SHOULD include a **Warning** header, as specified in [\[RFC3261\]](#) section 7.3.1. The value of the **Warning** header MUST be constructed as follows:

- **Warn-code** is set to the value 312.
- **Warn-agent** is set to the value "lcs.microsoft.com".
- **Warn-text** is set to the base location to download a newer version of the client. This location is an HTTPS URI and points to a webpage. The URI MUST contain a parameter with the name **folder**. The server constructs a path relative to this base location, as described in section [3.4.1.5.2](#). The

installers are either **Windows Installer (.msi) file** or .msp file. The client can automatically install a newer version of the client from this location.

The following is an example of a **Warning** header with a base HTTPS URL that is sent back in the response.

```
Warning: 312 lcs.microsoft.com
"https://fabrikam.com/AutoUpdate/Int/Handler/update.aspx?folder=OCv3.1
```

3.4.2.5.9 Sending 200 OK response (Action: Allow client with dynamic URL)

This section follows the behavior described in product behavior note [<160>](#).

If the server determines that it can support the version specified in the **User-Agent** header, it SHOULD allow processing of the REGISTER request and respond with a 200 OK response if the processing succeeds. If the action configured for the version of the client is to respond with a dynamic location, such as when an upgrade of the client is recommended, the server SHOULD include a **Warning** header, as specified in [\[RFC3261\]](#) section 7.3.1. The value of the **Warning** header MUST be constructed as follows:

- **Warn-code** is set to the value 312.
- **Warn-agent** is set to the value "lcs.microsoft.com".
- **Warn-text** is set to the base location to download a newer version of the client. This location is an HTTPS URI and points to a webpage. The URI MUST contain a parameter with the name **folder**. The server constructs a path relative to this base location, as described in section [3.4.1.5.2](#). The installers are either Windows Installer (.msi) files or .msp files. The client can automatically install a newer version of the client from this location.

The following is an example of a **Warning** header with a base HTTPS URL that is sent back in the response.

```
Warning: 312 lcs.microsoft.com
"https://fabrikam.com/AutoUpdate/Int/Handler/update.aspx?folder=OCv3.1
```

3.4.2.6 Timer Events

None.

3.4.2.7 Other Local Events

None.

3.5 Notifying the Client of Survivable Mode

This section follows the behavior described in product behavior note [<161>](#).

3.5.1 Server Role

None.

3.5.1.1 Abstract Data Model

None.

3.5.1.2 Timers

None.

3.5.1.3 Initialization

None.

3.5.1.4 Higher-Layer Triggered Events

The SIP registrar SHOULD send a NOTIFY to the UAC if it changes the state of the **Presence-Service-Available** field described in section [3.1.2.1](#) from "true" to "false". If the registrar notifies an endpoint and a NOTIFY is sent, it MUST construct the NOTIFY request as specified in the following subsection.

3.5.1.4.1 Constructing the Survivable Mode NOTIFY Request

The outgoing NOTIFY request is first constructed using the rules specified in section [2.2.1.12](#).

It MUST be sent only to the UAC that is being notified, and hence it MUST use the same local and remote tags, as specified in [\[RFC3261\]](#) section 12, and the **call-id** and **local CSeq** of the last REGISTER transaction sent from the UAC being notified, as specified in [\[RFC3265\]](#) section 3.3.4 and [\[RFC3261\]](#) section 10.3.

3.5.1.5 Message Processing Events and Sequencing Rules

None.

3.5.1.6 Timer Events

None.

3.5.1.7 Other Local Events

None.

4 Protocol Examples

4.1 Registration

4.1.1 Basic Registration

In the following example, user `alice@fabrikam.com` registers with the SIP registrar server. This is the first endpoint to register. The call flow is shown in the following diagram.

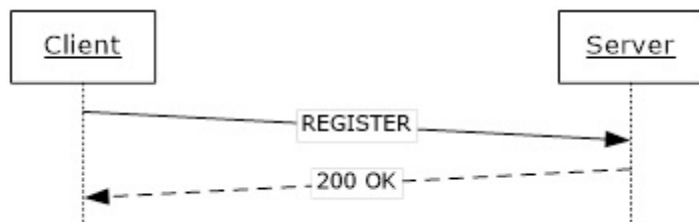


Figure 2: Basic registration call flow

The **authentication** challenge that is typically part of the registration sequence is not shown here.

```
REGISTER sip:fabrikam.com SIP/2.0
Via: SIP/2.0/TLS 10.1.2.50:4237
Max-Forwards: 70
From: <sip:alice@fabrikam.com>;tag=2d0be03279;epid=492a7ce35f
To: <sip:alice@fabrikam.com>
Call-ID: 74c55d45a6ee404680aa55c8fe126f11
CSeq: 3 REGISTER
Contact: <sip:10.1.2.50:4237;transport=tls;ms-opaque=6fb3a8330a>;methods="INVITE,
MESSAGE, INFO, OPTIONS, BYE, CANCEL, NOTIFY, ACK, REFER,
BENOTIFY";proxy=replace;+sip.instance="<urn:uuid:B43B3D1D-9F8F-5FDC-9F74-3CA273CADB97>"
User-Agent: UCCP/2.0.6362.0 OC/2.0.6362.0 (Microsoft Office Communicator)
Authorization: NTLM qop="auth", realm="SIP Communications Service",
targetname="ocs.fabrikam.com", gssapi-
data="TLRMTVNTUADAAAAGAAAYHoAAAAYABgAkgAAAAoACgBIAAAAEAAQAFIAAAAYABgAYgAAABAAEACqAAAAVYK
QQgUCzg4AAAAPTgBUAEQARQBWAHMAVQBuAGsAYQByAGEAbgBTAEAEATgBLAEAAUgBBAAE4ALQBAAEUAVgDnJV84fE5
9ziCDecY7XprbBXcdABFbpbpbnJVs84fE59ziCDecY7XprbBXcdABFbpbZwMLke/elE9fi69enePNxU",
opaque="99052D67", version=3
Supported: gruu-10, adhoclist, msrtc-event-categories
Supported: ms-cluster-failover
Supported: ms-userservices-state-notification
ms-keep-alive: UAC;hop-hop=yes
ms-subnet: 10.1.2.1
Event: registration
Content-Length: 0
```

The server responds with a 200 OK response and includes the GRUU contact URI to be used in subsequent dialog initiating requests, as follows.

```
SIP/2.0 200 OK
ms-keep-alive: UAS; tcp=no; hop-hop=yes; end-end=no; timeout=300
Authentication-Info: NTLM rspauth="0100000000636F6E9575D2297F8F5EF3", srand="168CE345",
snum="1", opaque="99052D67", qop="auth", targetname="ocs.fabrikam.com", realm="SIP
Communications Service"
From: <sip:alice@fabrikam.com>;tag=2d0be03279;epid=492a7ce35f
To: <sip:alice@fabrikam.com>;tag=0D08337B2339766A952BC5B9D8E0A41A
Call-ID: 74c55d45a6ee404680aa55c8fe126f11
CSeq: 3 REGISTER
Via: SIP/2.0/TLS 10.1.2.50:4237;received=10.54.78.109;ms-received-port=4237;ms-received-
cid=10A0D00
```

```
Contact: <sip:10.54.78.109:4237;transport=tls;ms-opaque=6fb3a8330a;ms-received-
cid=10A0D00>;expires=7200;+sip.instance="<urn:uuid:b43b3d1d-9f8f-5fdc-9f74-
3ca273cadb97>";gruu="sip:alice@fabrikam.com;opaque=user:epid:HT07tI-f3F-
fdDyic8rblwAA;gruu"
Expires: 7200
presence-state: register-action="added";primary-cluster-type="central";is-connected-to-
primary="yes"
Allow-Events: vnd-microsoft-provisioning,vnd-microsoft-roaming-contacts,vnd-microsoft-
roaming-ACL,presence,presence.wpending,vnd-microsoft-roaming-self,vnd-microsoft-
provisioning-v2
Supported: adhoclist
Server: RTC/3.0
Supported: msrtc-event-categories
Content-Length: 0
```

4.1.2 Basic Unregistration

The user unregisters from the SIP registrar by sending a REGISTER request with an **Expires: 0** header, as follows.

```
REGISTER sip:fabrikam.com SIP/2.0
Via: SIP/2.0/TLS 10.1.2.50:4237
Max-Forwards: 70
From: <sip:alice@fabrikam.com>;tag=2d0be03279;epid=492a7ce35f
To: <sip:alice@fabrikam.com>
Call-ID: 74c55d45a6ee404680aa55c8fe126f11
CSeq: 3 REGISTER
Contact: <sip:10.1.2.50:4237;transport=tls;ms-opaque=6fb3a8330a>;methods="INVITE, MESSAGE,
INFO, OPTIONS, BYE, CANCEL, NOTIFY, ACK, REFER,
BENOTIFY";proxy=replace;+sip.instance="<urn:uuid:B43B3D1D-9F8F-5FDC-9F74-3CA273CADB97>"
User-Agent: UCCP/2.0.6362.0 OC/2.0.6362.0 (Microsoft Office Communicator)
Authorization: NTLM qop="auth", realm="SIP Communications Service",
targetname="ocs.fabrikam.com", gssapi-
data="TlRMTVNTUAADAAAAGAAAYHoAAAYABgAkgAAAAoACgBIAAAAEAAQAFIAAAAYBgAYgAAABAAEACqAAAAVYKQOgU
Czg4AAAAPTgBUAEQARQBWAHMAVQBuAGsAYQByAGEAbgBTAEATgBLAEEAUgBBAE4ALQBAAEUAVgDnJV84fE59ziCDecY
7XprbBXcdABFbpbJVs84fE59ziCDecY7XprbBXcdABFbpbZwMLke/elE9fi69enePNxU", opaque="99052D67",
version=3
Allow-Events: vnd-microsoft-provisioning,vnd-microsoft-roaming-contacts,vnd-microsoft-
roaming-ACL,presence,presence.wpending,vnd-microsoft-roaming-self,vnd-microsoft-provisioning-
v2
Supported: gruu-10, adhoclist, msrtc-event-categories
ms-keep-alive: UAC;hop-hop=yes
Event: registration
Expires: 0
Content-Length: 0
```

The server responds with a 200 OK response, terminating the registration.

```
SIP/2.0 200 OK
ms-keep-alive: UAS; tcp=no; hop-hop=yes; end-end=no; timeout=300
Authentication-Info: NTLM rspauth="010000000636F6E9575D2297F8F5EF3", srand="168CE345",
snum="1", opaque="99052D67", qop="auth", targetname="ocs.fabrikam.com", realm="SIP
Communications Service"
From: <sip:alice@fabrikam.com>;tag=2d0be03279;epid=492a7ce35f
To: <sip:alice@fabrikam.com>;tag=0D08337B2339766A952BC5B9D8E0A41A
Call-ID: 74c55d45a6ee404680aa55c8fe126f11
CSeq: 3 REGISTER
Via: SIP/2.0/TLS 10.1.2.50:4237;received=10.54.78.109;ms-received-port=4237;ms-received-
cid=10A0D00
Allow-Events: vnd-microsoft-provisioning,vnd-microsoft-roaming-contacts,vnd-microsoft-
roaming-ACL,presence,presence.wpending,vnd-microsoft-roaming-self,vnd-microsoft-provisioning-
v2
Supported: adhoclist, msrtc-event-categories
```

Expires: 0
Server: RTC/3.0
Content-Length: 0

4.1.3 Deregistration

If the server deregisters a user's endpoint because of policy, it sends a deregister NOTIFY, as follows. Note that such deregistrations are usually triggered by server (2) policies, such as endpoint (5) quota enforcement.

```
NOTIFY sip:10.1.2.178:4126;transport=tls;ms-opaque=9a9503bcd;ms-received-cid=600 SIP/2.0
Via: SIP/2.0/TLS 10.1.2.178:5061;branch=z9hG4bKD948963B.BFDD97BF;branched=FALSE;ms-
ts="Wed, 06 Feb 2008 00:41:50 GMT 0"
Authentication-Info: Kerberos
rspauth="602306092A864886F71201020201011100FFFFFFFFFCC4C8119E16FEC024B25E89E96BA0865",
srand="AA69CAF1", snum="9", opaque="28CE6668", qop="auth",
targetname="sip/ocs.fabrikam.com", realm="SIP Communications Service"
Max-Forwards: 70
Content-Length: 27
From: <sip:alice@fabrikam.com>;tag=25498F391570ACE435E286F873E6AACC
To: sip:alice@fabrikam.com;tag=7c503117ea;epid=90e62db136
Call-ID: 6d7b09278d574162a4ef50c25e4e5051
CSeq: 1 NOTIFY
Content-Type: text/registration-event
Event: registration-notify
subscription-state: terminated;expires=0
ms-diagnostics-public: 4141;reason="User disabled"

deregistered;event=rejected
```

4.1.4 Survivable Mode Notify

If the presence service for a UAC becomes unavailable and the UAC has indicated support for survivable mode, the server sends a NOTIFY.

```
NOTIFY sip:10.1.2.178:4126;transport=tls;ms-opaque=9a9503bcd;ms-received-cid=600 SIP/2.0
Via: SIP/2.0/TLS 10.1.2.178:5061;branch=z9hG4bKD948963B.BFDD97BF;branched=FALSE;ms-
ts="Wed, 06 Feb 2008 00:41:50 GMT 0"
Authentication-Info: Kerberos
rspauth="602306092A864886F71201020201011100FFFFFFFFFCC4C8119E16FEC024B25E89E96BA0865",
srand="AA69CAF1", snum="9", opaque="28CE6668", qop="auth",
targetname="sip/ocs.fabrikam.com", realm="SIP Communications Service"
Max-Forwards: 70
Content-Length: 41
From: <sip:alice@fabrikam.com>;tag=25498F391570ACE435E286F873E6AACC
To: sip:alice@fabrikam.com;tag=7c503117ea;epid=90e62db136
Call-ID: 6d7b09278d574162a4ef50c25e4e5051
CSeq: 1 NOTIFY
Content-Type: text/registration-event
Event: registration-notify
Subscription-state: active;expires=900
ms-diagnostics-public: 4165;reason="User Services Pool is unavailable."

registered;event=userservices-unavailable
```

4.1.5 Notify for Registrar Change

If the system supports a dedicated "home registrar" and the user is being moved across registrars, the server sends a NOTIFY.

```

NOTIFY sip:10.1.2.178:4126;transport=tls;ms-opaque=9a9503bcd;ms-received-cid=600 SIP/2.0
Via: SIP/2.0/TLS 10.1.2.178:5061;branch=z9hG4bKD948963B.BFDD97BF;branched=FALSE;ms-
ts="Wed, 06 Feb 2008 00:41:50 GMT 0"
Authentication-Info: Kerberos
rspauth="602306092A864886F71201020201011100FFFFFFFFFCC4C8119E16FEC024B25E89E96BA0865",
srand="AA69CAF1", snum="9", opaque="28CE6668", qop="auth",
targetname="sip/ocs.fabrikam.com", realm="SIP Communications Service"
Max-Forwards: 70
Content-Length: 41
From: <sip:alice@fabrikam.com>;tag=25498F391570ACE435E286F873E6AACC
To: sip:alice@fabrikam.com>;tag=7c503117ea;epid=90e62db136
Call-ID: 6d7b09278d574162a4ef50c25e4e5051
CSeq: 25 NOTIFY
Content-Type: text/registration-event
Event: registration-notify
Subscription-state: terminated;expires=0
P-Preferred-Registrator: <sip:server1.fabrikam.com:5061;transport=tls>
ms-diagnostics-public: 4166;reason="Preferred Registrar Change."

deregistered;event=preferred-registrar-change

```

4.2 In-band Provisioning

The UAC sends an in-band provisioning request to the server. The server responds to the request and generates a notification with the in-band provisioning data. The following example assumes support for piggyback-notifications, as described in [MS-SIP] section 3.4, where the response document is returned in the 200 OK SUBSCRIBE response itself. This is shown in the following diagram.

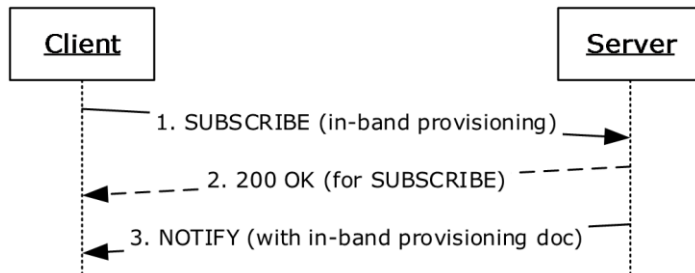


Figure 3: In-band provisioning call flow

4.2.1 Client to Server Request

In the following request, user `alice@fabrikam.com` sends an **application/vnd-microsoft-roaming-provisioning-v2+xml** request to the server, requesting all supported provisioning groups.

```

SUBSCRIBE sip:alice@fabrikam.com SIP/2.0
Via: SIP/2.0/TLS 172.24.32.149:53990
Max-Forwards: 70
From: <sip:alice@fabrikam.com>;tag=77399b3fb7;epid=e48fdaa231
To: <sip:alice@fabrikam.com>
Call-ID: dcb605d5e73e44ccbe84dbe4886d210f
CSeq: 1 SUBSCRIBE
Contact: <sip:alice@fabrikam.com;opaque=user:epid:A3zr0x5_sVSbfPZ-4_JkrgAA;gruu>

User-Agent: UCCAPI/4.0.7577.0 OC/4.0.7577.0 (Microsoft Communicator 2010)

Event: vnd-microsoft-provisioning-v2
Accept: application/vnd-microsoft-roaming-provisioning-v2+xml
Supported: com.microsoft.autoextend

```



```

Supported: ms-benotify
Proxy-Require: ms-benotify
Supported: ms-piggyback-first-notify
Expires: 0
Proxy-Authorization: Kerberos qop="auth", realm="SIP Communications Service",
opaque="224773FA", targetname="sip/clientmainse.fabrikam.com", crand="571d3235",
cnum="4", response="040400fffffffff0000000000000000d2a625d0f476403d15d58285"
Content-Type: application/vnd-microsoft-roaming-provisioning-v2+xml
Content-Length: 625
<provisioningGroupList
xmlns="http://schemas.microsoft.com/2006/09/sip/provisioninggrouplist"
subnet="172.24.32.0">
  <provisioningGroup name="endpointConfiguration" />
  <provisioningGroup name="locationPolicy" />
  <provisioningGroup name="mediaConfiguration" />
  <provisioningGroup name="meetingPolicy" />
  <provisioningGroup name="presencePolicyV2" />
  <provisioningGroup name="privacyPublicationGrammar" />
  <provisioningGroup name="publicationGrammar" />
  <provisioningGroup name="ServerConfiguration" />
  <provisioningGroup name="ucPolicy" />
  <provisioningGroup name="userSetting" />
  <provisioningGroup name="publicProviders" />
</provisioningGroupList>

```

4.2.2 Server to Client Response

In the following response, the server responds with the data for all requested provisioning groups.

```

SIP/2.0 200 OK
Contact: <sip:clientmainse.fabrikam.com:5061;transport=tls>
Proxy-Authentication-Info: Kerberos qop="auth", opaque="224773FA", srand="84924759",
snum="4", rspauth="040401fffffffff00000000000000001516d27bd2ddfb27246db52a",
targetname="sip/clientmainse.fabrikam.com", realm="SIP Communications Service", version=4
Content-Length: 52146
From: "Alice" <sip:alice@fabrikam.com>;tag=77399b3fb7;epid=e48fdaa231
To: <sip:alice@fabrikam.com>;tag=24010080
Call-ID: dcb605d5e73e44ccbe84dbe4886d210f
CSeq: 1 SUBSCRIBE
Via: SIP/2.0/TLS 172.24.32.149:53990;ms-received-port=53990;ms-received-cid=914800
Expires: 0
Content-Type: application/vnd-microsoft-roaming-provisioning-v2+xml
Event: vnd-microsoft-provisioning-v2
subscription-state: terminated;expires=0
ms-piggyback-cseq: 1
Supported: ms-benotify, ms-piggyback-first-notify
<provisionGroupList xmlns="http://schemas.microsoft.com/2006/09/sip/provisiongrouplist-
notification">
  <provisionGroup name="endpointConfiguration">
    <propertyEntryList>
      <property name="ShowRecentContacts">true</property>
      <property name="ShowManagePrivacyRelationships">>false</property>
      <property name="MaxPhotoSizeKB">10</property>
      <property name="DisableMusicOnHold" >>false</property>
      <property name="EnableTracing" >true</property>
      <property name="PhotoUsage">AllPhotos</property>
      <property name="AbsUsage">WebSearchAndFileDownload</property>
      <property name="EnableHotdesking">true</property>
      <property name="HotdeskingTimeout">180</property>
      <property name="SPSearchInternalUrl">http://my/_vti_bin/search.aspx</property>
      <property name="SPSearchExternalUrl">http://my/_vti_bin/search.aspx</property>
    </propertyEntryList>
  </provisionGroup>
  <provisionGroup name="SPSearchCenterInternalUrl">http://msw/searchcenter/pages/peopleresults.aspx</property>
  <property name="EnableContactSync" >true</property>
  <property name="ShowSharepointPhotoEditLink" >true</property>

```

```

    <property name="EnableVOIPCallDefault" >false</property>
    <property name="MaximumDGsAllowedInContactList" >10</property>
    <property name="imLatencySpinnerDelay" >1500</property>
    <property name="imLatencyErrorThreshold" >1500</property>
    <property name="EnableMediaRedirection" >true</property>
    <property name="P2PAppSharingEncryption" >0</property>
    <property name="CustomLinkInErrorMessages">http://octop</property>
    <property name="BITSServerAddressInternal">http://Oly11-
ocg.exchange.corp.microsoft.com/trdin</property>
    <property name="BITSServerAddressExternal">https://comodf.microsoft.com/trd</property>
    <property name="EnableSendFeedback">1</property>
    <property name="EnableIssueReports">1</property>
    <property name="BotSipUriForTestCall">ootycorp@microsoft.com</property>
    <property name="EnableBugFiling">1</property>
    <property name="EnableBackgroundDataCollection">1</property>
    <property
name="OnlineFeedbackURL">http://vote/scenarios/communicator/dogfood.aspx?&Site=2b08429f-
4651-4901-af51-1d0ca5bca2b7</property>
    <property name="PrivacyStatementURL">http://www.microsoft.com/communicator</property>
    <property name="SendFeedbackURL">df-mqd-dcs-01</property>
  </propertyEntryList>
</provisionGroup>
<provisionGroup name="locationPolicy">
  <propertyEntryList>
    <property name="EnhancedEmergencyServicesEnabled">true</property>
    <property name="LocationPolicyTagID">user-tagid</property>
    <property name="LocationRequired">yes</property>
    <property name="UseLocationForE911Only">false</property>
    <property name="EmergencyDialString">119</property>
    <property name="EmergencyDialMask">112;100</property>
    <property name="NotificationUri">sip:notify@microsoft.com</property>
    <property name="ConferenceUri">sip:security@microsoft.com</property>
    <property name="EmergencyServiceDisclaimer">Your company policy requires you to set a
location. If you do not set a location emergency services will not be able to locate you in
an emergency. Please set a location.</property>
    <property name="LocationRefreshInterval">4</property>
    <property name="RateMyCallAllowCustomUserFeedback" >false</property>
    <property name="RateMyCallDisplayPercentage" >10</property>
    <property name="imLatencySpinnerDelay" >1500</property>
    <property name="imLatencyErrorThreshold" >1500</property>
    <property name="P2PAppSharingEncryption" >0</property>
    <property name="EnableHighPerformanceP2PAppSharing" >false</property>
    <property name="EnableHighPerformanceConferencingAppSharing" >false</property>
    <property name="EnableServerConversationHistory" >false</property>
    <property name="TracingLevel" >Light</property>
    <property name="EnableSkypeUI" >true</property>
    <property name="CallViaWorkEnabled" >true</property>
    <property name="UseAdminCallbackNumber" >true</property>
    <property name="AdminCallbackNumber" >425111111</property>
  </propertyEntryList>
</provisionGroup>
<provisionGroup name="mediaConfiguration">
  <propertyEntryList>
    <property name="bypassEnabled">true</property>
    <property name="internalBypassMode">Any</property>
    <property name="externalBypassMode">Off</property>
  </propertyEntryList>
</provisionGroup>
<provisionGroup name="meetingPolicy">
  <instance>
    <property name="AllowIPAudio">true</property>
    <property name="AllowIPVideo">true</property>
    <property name="EnableAppDesktopSharing">true</property>
    <property name="AllowAppSharingForExternalMeeting">Desktop</property>
    <property name="RetainPPTForExternalMeeting">true</property>
    <property name="AllowPresenterToRecord">true</property>
    <property name="EnableDataCollaboration">true</property>
    <property name="MeetingSize">50</property>
    <property name="EnablePSTNConferencing">true</property>
  </instance>

```

```

    <property name="TrustedConferencingPinRequired">false</property>
    <property name="AllowParticipantControl">true</property>
    <property name="AllowAnnotations">true</property>
    <property name="AllowAnonymousParticipants">true</property>
    <property name="AllowExternalUserControl">true</property>
    <property name="AllowExternalUserToSaveContent">true</property>
    <property name="AllowExternalUserRecording">true</property>
    <property name="AllowRecording">true</property>
    <property name="AllowFileTransfer">true</property>
    <property name="MaxConferenceVideoResolution">VGA</property>
    <property name="AllowUserToScheduleMeetingsWithAppSharing">true</property>
    <property name="EnableP2PFileTransfer">true</property>
    <property name="AllowedAppDesktopSharingLevel">Desktop</property>
    <property name="RequirePasscode">false</property>
    <property name="AudioBitRate">200</property>
    <property name="VideoBitRate">2000</property>
    <property name="AppSharingBitRate">2000</property>
    <property name="FileTransferBitRate">2000</property>
    <property name="EnableMultiViewJoin">true</property>
    <property name="TotalReceiveVideoBitRate">6000</property>
    <property name="DisablePowerPointAnnotations">false</property>
    <property name="AllowSharedNotes">false</property>
    <property name="AllowQna">false</property>
    <property name="AllowOfficeContent">true</property>
    <property name="EnableOnlineMeetingPromptForLyncResources">false</property>
  </instance>
</provisionGroup>
<provisionGroup name="presencePolicyV2">
  <propertyEntryList>
    <property name="EnablePrivacyMode">false</property>
    <property name="AutoInitiateContacts">true</property>
    <property name="PublishLocationDataDefault">true</property>
    <property name="DisplayPublishedPhotoDefault">true</property>
    <property name="PersonalNoteHistoryDepth">3</property>
    <property name="SubscribeToCollapsedDG" >true</property>
  </propertyEntryList>
</provisionGroup>
<provisionGroup name="privacyPublicationGrammar">
  <containerManifestList xmlns="http://schemas.microsoft.com/2008/09/sip/ContainerManifest"
  xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"
  minSupportedClientVersion="2.0.0.0">
    <containerManifest>
      <containers>
        <container id="0">
          <visible>false</visible>
          <avoidRedundantMembers>false</avoidRedundantMembers>
        </container>
        <container id="100">
          <visible>true</visible>
          <avoidRedundantMembers>true</avoidRedundantMembers>
        </container>
        <container id="200">
          <visible>true</visible>
          <avoidRedundantMembers>true</avoidRedundantMembers>
        </container>
        <container id="300">
          <visible>true</visible>
          <avoidRedundantMembers>true</avoidRedundantMembers>
        </container>
        <container id="400">
          <visible>true</visible>
          <avoidRedundantMembers>true</avoidRedundantMembers>
        </container>
        <container id="32000">
          <visible>true</visible>
          <avoidRedundantMembers>false</avoidRedundantMembers>
          <blocked>true</blocked>
        </container>
      </containers>
    </containerManifest>
  </containerManifestList>

```

```

<members>
  <member role="buddy">
    <allowedContainers type="visibleContainers" />
    <defaultContainer id="100" />
    <occurrenceConstraint type="one" />
    <ct:delimiter />
    <sourceNetworks>
      <sourceNetwork type="publicCloud" />
      <sourceNetwork type="federated" />
    </sourceNetworks>
    <resolutionRules>
      <resolutionRule type="moveToDefaultContainer" />
    </resolutionRules>
    <ct:end />
  </member>
  <member role="buddy">
    <allowedContainers type="visibleContainers" />
    <defaultContainer id="200" />
    <occurrenceConstraint type="one" />
    <ct:delimiter />
    <sourceNetworks>
      <sourceNetwork type="sameEnterprise" />
    </sourceNetworks>
    <resolutionRules>
      <resolutionRule type="moveToDefaultContainer" />
    </resolutionRules>
    <ct:end />
  </member>
  <member type="domain">
    <allowedContainers type="custom">
      <container id="32000" />
    </allowedContainers>
    <ct:delimiter />
    <resolutionRules>
      <resolutionRule type="removeFromContainer" />
    </resolutionRules>
    <ct:end />
  </member>
  <member type="user">
    <allowedContainers type="custom">
      <container id="32000" />
    </allowedContainers>
    <ct:delimiter />
    <resolutionRules>
      <resolutionRule type="forceRole" role="buddy" />
    </resolutionRules>
    <ct:end />
  </member>
  <member type="sameEnterprise">
    <allowedContainers type="custom">
      <container id="32000" />
    </allowedContainers>
    <occurrenceConstraint type="zeroOrOne" />
    <ct:delimiter />
    <resolutionRules>
      <resolutionRule type="removeFromContainer" />
    </resolutionRules>
    <ct:end />
  </member>
  <member type="federated">
    <allowedContainers type="custom">
      <container id="32000" />
    </allowedContainers>
    <occurrenceConstraint type="zeroOrOne" />
    <ct:delimiter />
    <resolutionRules>
      <resolutionRule type="removeFromContainer" />
    </resolutionRules>
    <ct:end />
  </member>

```

```

</member>
<member type="publicCloud">
  <allowedContainers type="custom">
    <container id="32000" />
  </allowedContainers>
  <occurrenceConstraint type="zeroOrOne" />
  <ct:delimiter />
  <resolutionRules>
    <resolutionRule type="removeFromContainer" />
  </resolutionRules>
  <ct:end />
</member>
<member role="delegate">
  <allowedContainers type="custom">
    <container id="300" />
  </allowedContainers>
  <defaultContainer id="300" />
  <occurrenceConstraint type="one" />
  <ct:delimiter />
  <resolutionRules>
    <resolutionRule type="moveToDefaultContainer" />
  </resolutionRules>
  <ct:end />
</member>
</members>
<ct:delimiter />
<blockRules>
  <blockRule type="invites">
    <containers type="custom">
      <container id="0" />
    </containers>
    <sourceNetworks>
      <sourceNetwork type="publicCloud" />
    </sourceNetworks>
  </blockRule>
  <blockRule type="invites">
    <maxAvailability>11999</maxAvailability>
    <minAvailability>9000</minAvailability>
  </blockRule>
  <blockRule type="invites">
    <containers type="custom">
      <container id="32000" />
    </containers>
  </blockRule>
  <blockRule type="subscriberPrompt">
    <containers type="custom">
      <container id="32000" />
    </containers>
  </blockRule>
</blockRules>
<rolePrecedence>
  <rolePrecedenceEntry type="delegate" />
  <rolePrecedenceEntry type="buddy" />
</rolePrecedence>
<ct:end />
</containerManifest>
</containerManifestList>
</provisionGroup>
<provisionGroup name="publicationGrammar">
  <containerManifestList xmlns="http://schemas.microsoft.com/2008/09/sip/ContainerManifest"
  xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"
  minSupportedClientVersion="2.0.0.0">
    <containerManifest>
      <containers>
        <container id="0">
          <visible>>false</visible>
          <avoidRedundantMembers>>false</avoidRedundantMembers>
        </container>
        <container id="100">

```

```

    <visible>true</visible>
    <avoidRedundantMembers>true</avoidRedundantMembers>
</container>
<container id="200">
    <visible>true</visible>
    <avoidRedundantMembers>true</avoidRedundantMembers>
</container>
<container id="300">
    <visible>true</visible>
    <avoidRedundantMembers>true</avoidRedundantMembers>
</container>
<container id="400">
    <visible>true</visible>
    <avoidRedundantMembers>true</avoidRedundantMembers>
</container>
<container id="32000">
    <visible>true</visible>
    <avoidRedundantMembers>false</avoidRedundantMembers>
    <blocked>true</blocked>
</container>
</containers>
<members>
  <member role="buddy">
    <allowedContainers type="visibleContainers" />
    <defaultContainer id="100" />
    <occurrenceConstraint type="one" />
    <ct:delimiter />
    <sourceNetworks>
      <sourceNetwork type="publicCloud" />
      <sourceNetwork type="federated" />
    </sourceNetworks>
    <resolutionRules>
      <resolutionRule type="moveToDefaultContainer" />
    </resolutionRules>
    <ct:end />
  </member>
  <member role="buddy">
    <allowedContainers type="visibleContainers" />
    <defaultContainer id="200" />
    <occurrenceConstraint type="one" />
    <ct:delimiter />
    <sourceNetworks>
      <sourceNetwork type="sameEnterprise" />
    </sourceNetworks>
    <resolutionRules>
      <resolutionRule type="moveToDefaultContainer" />
    </resolutionRules>
    <ct:end />
  </member>
  <member type="sameEnterprise">
    <allowedContainers type="visibleContainers" />
    <defaultContainer id="200" />
    <occurrenceConstraint type="one" />
  </member>
  <member type="federated">
    <allowedContainers type="visibleContainers" />
    <defaultContainer id="100" />
    <occurrenceConstraint type="one" />
  </member>
  <member type="publicCloud">
    <allowedContainers type="visibleContainers" />
    <occurrenceConstraint type="zeroOrOne" />
  </member>
  <member role="delegate">
    <allowedContainers type="custom">
      <container id="300" />
    </allowedContainers>
    <defaultContainer id="300" />
    <occurrenceConstraint type="one" />
  </member>

```

```

        <ct:delimiter />
        <resolutionRules>
            <resolutionRule type="moveToDefaultContainer" />
        </resolutionRules>
        <ct:end />
    </member>
</members>
<ct:delimiter />
<blockRules>
    <blockRule type="invites">
        <containers type="custom">
            <container id="0" />
        </containers>
        <sourceNetworks>
            <sourceNetwork type="publicCloud" />
        </sourceNetworks>
    </blockRule>
    <blockRule type="invites">
        <maxAvailability>11999</maxAvailability>
        <minAvailability>9000</minAvailability>
    </blockRule>
    <blockRule type="invites">
        <containers type="custom">
            <container id="32000" />
        </containers>
    </blockRule>
    <blockRule type="subscriberPrompt">
        <containers type="custom">
            <container id="32000" />
        </containers>
    </blockRule>
</blockRules>
<rolePrecedence>
    <rolePrecedenceEntry type="delegate" />
    <rolePrecedenceEntry type="buddy" />
</rolePrecedence>
<ct:end />
</containerManifest>
</containerManifestList>
<categoryPublicationManifest xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns="http://schemas.microsoft.com/2008/09/sip/categoryPublicationManifest"
minSupportedClientVersion="2.0.0.0">
    <publicationList>
        <publicationRule ruleType="transformation" categoryName="state" containerId="2"
preferredSelfConsumption="true" />
        <publicationRule ruleType="transformation" categoryName="state" containerId="3">
            <categoryData><![CDATA[ <xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commonTypes"
xmlns:tns="http://schemas.microsoft.com/2006/09/sip/state" exclude-result-prefixes="tns
ct">
    <xsl:output method="xml" encoding="utf-8" indent="yes"/>
    <xsl:strip-space
elements="*" />
    <xsl:template match="node()">
        <xsl:apply-templates
select="//tns:state"/>
    </xsl:template>
    <xsl:template match="tns:state">
<xsl:copy>
    <xsl:copy-of select="@*" />
    <xsl:choose>
        <xsl:when
test="@xsi:type='userState' and tns:availability=9000 and 11999">tns:availability and
@manual='true'">
            <xsl:element name="availability"
xmlns="http://schemas.microsoft.com/2006/09/sip/state">6900</xsl:element>
<xsl:element name="activity"
xmlns="http://schemas.microsoft.com/2006/09/sip/state">
            <xsl:attribute
name="token">urgent-interruptions-only</xsl:attribute>
            <xsl:attribute
name="minAvailability">6900</xsl:attribute>
            <xsl:attribute
name="maxAvailability">8999</xsl:attribute>
            </xsl:element>
            </xsl:when>
<xsl:otherwise>
            <xsl:copy-of select="tns:availability" />
            <xsl:copy-of
select="tns:activity" />
            </xsl:otherwise>
            </xsl:choose>
            <xsl:copy-of
select="tns:endpointLocation" />
            <xsl:copy-of select="tns:meetingSubject" />
            <xsl:copy-of select="tns:meetingLocation" />
            <xsl:choose>
                <xsl:when
test="count(ct:delimiter) > 0">
                    <xsl:copy-of select="ct:delimiter[1]" />

```

```

<xsl:copy-of select="tns:timeZoneBias" />          <xsl:copy-of select="tns:timeZoneName"
/>          <xsl:copy-of select="tns:timeZoneAbbreviation" />          <xsl:copy-of
select="tns:device" />          </xsl:when>          </xsl:choose>          <xsl:choose>
<xsl:when test="count(ct:delimiter) > 1">          <xsl:copy-of select="ct:delimiter[2]"
/>          <xsl:copy-of select="ct:delimiter[2]/following-sibling::*" />
</xsl:when>          <xsl:otherwise>          <xsl:copy-of select="ct:end" />
<xsl:copy-of select="ct:extension" />          </xsl:otherwise>          </xsl:choose>
</xsl:copy>          </xsl:template> </xsl:stylesheet>    ]></categoryData>
  </publicationRule>
  <publicationRule ruleType="transformation" categoryName="device" containerId="2"
preferredSelfConsumption="true" />
  <publicationRule ruleType="transformation" categoryName="contactCard"
containerId="100">
    <categoryData><![CDATA[ <xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"
xmlns:tns="http://schemas.microsoft.com/2006/09/sip/contactcard" exclude-result-
prefixes="tns ct"> <xsl:output method="xml" encoding="utf-8" indent="yes"/> <xsl:strip-
space elements="*" /> <xsl:template match="node()"> <xsl:apply-templates
select="//tns:contactCard"/> </xsl:template> <xsl:template match="tns:name">
<xsl:copy> <xsl:copy-of select="@*" /> <xsl:copy-of select="tns:displayName"
/> <xsl:choose> <xsl:when test="count(ct:delimiter) > 0"> <xsl:copy-of
select="ct:delimiter[1]" /> <xsl:copy-of select="ct:delimiter[1]/following-
sibling::*" /> </xsl:when> <xsl:otherwise> <xsl:copy-of
select="ct:end" /> <xsl:copy-of select="ct:extension" />
</xsl:otherwise> </xsl:choose> </xsl:copy> </xsl:template> <xsl:template
match="tns:identity"> <xsl:copy> <xsl:copy-of select="@*" /> <xsl:apply-
templates select="tns:name" /> <xsl:copy-of select="tns:email" />
<xsl:choose> <xsl:when test="count(ct:delimiter) > 0"> <xsl:copy-of
select="ct:delimiter[1]" /> <xsl:copy-of select="ct:delimiter[1]/following-
sibling::*" /> </xsl:when> <xsl:otherwise> <xsl:copy-of
select="ct:end" /> <xsl:copy-of select="ct:extension" />
</xsl:choose> </xsl:otherwise> </xsl:copy> </xsl:template> <xsl:template
match="tns:contactCard"> <xsl:copy> <xsl:copy-of select="@*" /> <xsl:apply-
templates select="tns:identity" /> <xsl:copy-of select="tns:company" />
<xsl:copy-of select="tns:department" /> <xsl:copy-of select="tns:title" />
<xsl:copy-of select="tns:url[@type='voicemail']" /> <xsl:choose> <xsl:when
test="count(ct:delimiter) > 0"> <xsl:copy-of select="ct:delimiter[1]" />
</xsl:when> </xsl:choose> <xsl:choose> <xsl:when test="count(ct:delimiter)
> 1"> <xsl:copy-of select="ct:delimiter[2]" /> <xsl:copy-of
select="tns:displayADPhoto" /> <xsl:copy-of select="tns:photo" />
</xsl:when> </xsl:choose> <xsl:choose> <xsl:when test="count(ct:delimiter)
> 2"> <xsl:copy-of select="ct:delimiter[3]" /> <xsl:copy-of
select="ct:delimiter[3]/following-sibling::*" /> </xsl:when>
<xsl:otherwise> <xsl:copy-of select="ct:end" /> <xsl:copy-of
select="ct:extension" /> </xsl:otherwise> </xsl:choose> </xsl:copy>
</xsl:template> </xsl:stylesheet>    ]></categoryData>
  </publicationRule>
  <publicationRule ruleType="transformation" categoryName="contactCard"
containerId="200">
    <categoryData><![CDATA[ <xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"
xmlns:tns="http://schemas.microsoft.com/2006/09/sip/contactcard" exclude-result-
prefixes="tns ct"> <xsl:output method="xml" encoding="utf-8" indent="yes"/> <xsl:strip-
space elements="*" /> <xsl:template match="node()"> <xsl:apply-templates
select="//tns:contactCard"/> </xsl:template> <xsl:template match="tns:name">
<xsl:copy> <xsl:copy-of select="@*" /> <xsl:copy-of select="tns:displayName"
/> <xsl:choose> <xsl:when test="count(ct:delimiter) > 0"> <xsl:copy-of
select="ct:delimiter[1]" /> <xsl:copy-of select="ct:delimiter[1]/following-
sibling::*" /> </xsl:when> <xsl:otherwise> <xsl:copy-of
select="ct:end" /> <xsl:copy-of select="ct:extension" />
</xsl:otherwise> </xsl:choose> </xsl:copy> </xsl:template> <xsl:template
match="tns:identity"> <xsl:copy> <xsl:copy-of select="@*" /> <xsl:apply-
templates select="tns:name" /> <xsl:copy-of select="tns:email" />
<xsl:choose> <xsl:when test="count(ct:delimiter) > 0"> <xsl:copy-of
select="ct:delimiter[1]" /> <xsl:copy-of select="ct:delimiter[1]/following-
sibling::*" /> </xsl:when> <xsl:otherwise> <xsl:copy-of

```



```

select="ct:end" />          <xsl:copy-of select="ct:extension" />          </xsl:otherwise>
  </xsl:choose>          </xsl:copy>          </xsl:template>          <xsl:template match="tns:address">
<xsl:copy>          <xsl:copy-of select="@*" />          <xsl:copy-of select="tns:street" />
<xsl:copy-of select="tns:city" />          <xsl:copy-of select="tns:state" />          <xsl:copy-of
select="tns:zipcode" />          <xsl:copy-of select="tns:countryCode" />
<xsl:choose>          <xsl:when test="count(ct:delimiter) > 0">          <xsl:copy-of
select="ct:delimiter[1]" />          <xsl:copy-of select="ct:delimiter[1]/following-
sibling::*" />          </xsl:when>          <xsl:otherwise>          <xsl:copy-of
select="ct:end" />          <xsl:copy-of select="ct:extension" />
</xsl:otherwise>          </xsl:choose>          </xsl:copy>          </xsl:template>          <xsl:template
match="tns:phone">          <xsl:copy>          <xsl:copy-of select="@*" />          <xsl:copy-of
select="tns:uri" />          <xsl:copy-of select="tns:displayString" />
<xsl:choose>          <xsl:when test="count(ct:delimiter) > 0">          <xsl:copy-of
select="ct:delimiter[1]" />          <xsl:copy-of select="ct:delimiter[1]/following-
sibling::*" />          </xsl:when>          <xsl:otherwise>          <xsl:copy-of
select="ct:end" />          <xsl:copy-of select="ct:extension" />
</xsl:otherwise>          </xsl:choose>          </xsl:copy>          </xsl:template>          <xsl:template
match="tns:contactCard">          <xsl:copy>          <xsl:copy-of select="@*" />          <xsl:apply-
templates select="tns:identity" />          <xsl:apply-templates
select="tns:address[@type='work']" />          <xsl:copy-of select="tns:company" />
<xsl:copy-of select="tns:department" />          <xsl:copy-of select="tns:title" />
<xsl:copy-of select="tns:office" />          <xsl:copy-of select="tns:url" />          <xsl:apply-
templates select="tns:phone[@type='work']" />          <xsl:copy-of select="tns:automaton"
/>          <xsl:choose>          <xsl:when test="count(ct:delimiter) > 0">          <xsl:copy-of
select="ct:delimiter[1]" />          <xsl:copy-of select="tns:type" />          <xsl:copy-of
select="tns:description" />          </xsl:when>          </xsl:choose>          <xsl:choose>
<xsl:when test="count(ct:delimiter) > 1">          <xsl:copy-of select="ct:delimiter[2]"
/>          <xsl:copy-of select="tns:displayADPhoto" />          <xsl:copy-of
select="tns:photo" />          </xsl:when>          </xsl:choose>          <xsl:choose>
<xsl:when test="count(ct:delimiter) > 2">          <xsl:copy-of select="ct:delimiter[3]"
/>          <xsl:copy-of select="ct:delimiter[3]/following-sibling::*" />          </xsl:when>
          <xsl:otherwise>          <xsl:copy-of select="ct:end" />          <xsl:copy-of
select="ct:extension" />          </xsl:otherwise>          </xsl:choose>          </xsl:copy>
</xsl:template> </xsl:stylesheet> ]></categoryData>
  </publicationRule>
  <publicationRule ruleType="transformation" categoryName="contactCard"
containerId="300">
    <categoryData><![CDATA[ <xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"
xmlns:tns="http://schemas.microsoft.com/2006/09/sip/contactcard" exclude-result-
prefixes="tns ct">          <xsl:output method="xml" encoding="utf-8" indent="yes"/>          <xsl:strip-
space elements="*" />          <xsl:template match="node()">          <xsl:apply-templates
select="//tns:contactCard"/>          </xsl:template>          <xsl:template match="tns:name">
<xsl:copy>          <xsl:copy-of select="@*" />          <xsl:copy-of select="tns:displayName"
/>          <xsl:choose>          <xsl:when test="count(ct:delimiter) > 0">          <xsl:copy-of
select="ct:delimiter[1]" />          <xsl:copy-of select="ct:delimiter[1]/following-
sibling::*" />          </xsl:when>          <xsl:otherwise>          <xsl:copy-of
select="ct:end" />          <xsl:copy-of select="ct:extension" />
</xsl:otherwise>          </xsl:choose>          </xsl:copy>          </xsl:template>          <xsl:template
match="tns:identity">          <xsl:copy>          <xsl:copy-of select="@*" />          <xsl:apply-
templates select="tns:name" />          <xsl:copy-of select="tns:email" />
<xsl:choose>          <xsl:when test="count(ct:delimiter) > 0">          <xsl:copy-of
select="ct:delimiter[1]" />          <xsl:copy-of select="ct:delimiter[1]/following-
sibling::*" />          </xsl:when>          <xsl:otherwise>          <xsl:copy-of
select="ct:end" />          <xsl:copy-of select="ct:extension" />
</xsl:otherwise>          </xsl:choose>          </xsl:template>          <xsl:template
match="tns:address">          <xsl:copy>          <xsl:copy-of select="@*" />          <xsl:copy-of
select="tns:street" />          <xsl:copy-of select="tns:city" />          <xsl:copy-of
select="tns:state" />          <xsl:copy-of select="tns:zipcode" />          <xsl:copy-of
select="tns:countryCode" />          <xsl:choose>          <xsl:when test="count(ct:delimiter) >
0">          <xsl:copy-of select="ct:delimiter[1]" />          <xsl:copy-of
select="ct:delimiter[1]/following-sibling::*" />          </xsl:when>
          <xsl:otherwise>          <xsl:copy-of select="ct:end" />          <xsl:copy-of
select="ct:extension" />          </xsl:otherwise>          </xsl:choose>          </xsl:copy>
</xsl:template>          <xsl:template match="tns:phone">          <xsl:copy>          <xsl:copy-of
select="@*" />          <xsl:copy-of select="tns:uri" />          <xsl:copy-of
select="tns:displayString" />          <xsl:choose>          <xsl:when test="count(ct:delimiter) >
0">          <xsl:copy-of select="ct:delimiter[1]" />          <xsl:copy-of

```

```

select="ct:delimiter[1]/following-sibling::*" /> </xsl:when>
<xsl:otherwise> <xsl:copy-of select="ct:end" /> <xsl:copy-of
select="ct:extension" /> </xsl:otherwise> </xsl:choose> </xsl:copy>
</xsl:template> <xsl:template match="tns:contactCard"> <xsl:copy> <xsl:copy-of
select="@*" /> <xsl:apply-templates select="tns:identity" /> <xsl:apply-templates
select="tns:address[@type='work']" /> <xsl:copy-of select="tns:company" />
<xsl:copy-of select="tns:department" /> <xsl:copy-of select="tns:title" />
<xsl:copy-of select="tns:office" /> <xsl:copy-of select="tns:url" /> <xsl:apply-
templates select="tns:phone[@type='work']" /> <xsl:apply-templates
select="tns:phone[@type='mobile']" /> <xsl:copy-of select="tns:automaton" />
<xsl:choose> <xsl:when test="count(ct:delimiter) > 0"> <xsl:copy-of
select="ct:delimiter[1]" /> <xsl:copy-of select="tns:type" /> <xsl:copy-of
select="tns:description" /> </xsl:when> </xsl:choose> <xsl:choose>
<xsl:when test="count(ct:delimiter) > 1"> <xsl:copy-of select="ct:delimiter[2]"
/> <xsl:copy-of select="tns:displayADPhoto" /> <xsl:copy-of
select="tns:photo" /> </xsl:when> </xsl:choose> <xsl:choose>
<xsl:when test="count(ct:delimiter) > 2"> <xsl:copy-of select="ct:delimiter[3]"
/> <xsl:copy-of select="ct:delimiter[3]/following-sibling::*" />
</xsl:when> <xsl:otherwise> <xsl:copy-of select="ct:end" />
<xsl:copy-of select="ct:extension" /> </xsl:otherwise> </xsl:choose>
</xsl:copy> </xsl:template> </xsl:stylesheet> ]]></categoryData>
</publicationRule>
<publicationRule ruleType="transformation" categoryName="contactCard"
containerId="400" preferredSelfConsumption="true" />
<publicationRule ruleType="transformation" categoryName="contactCard"
containerId="32000">
<categoryData><![CDATA[ <xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"
xmlns:tns="http://schemas.microsoft.com/2006/09/sip/contactcard" exclude-result-
prefixes="tns ct"> <xsl:output method="xml" encoding="utf-8" indent="yes"/> <xsl:strip-
space elements="*" /> <xsl:template match="node()"> <xsl:apply-templates
select="//tns:contactCard"/> </xsl:template> <xsl:template match="tns:name">
<xsl:copy> <xsl:copy-of select="@*" /> <xsl:copy-of select="tns:displayName"
/> <xsl:choose> <xsl:when test="count(ct:delimiter) > 0"> <xsl:copy-of
select="ct:delimiter[1]" /> <xsl:copy-of select="ct:delimiter[1]/following-
sibling::*" /> </xsl:when> <xsl:otherwise> <xsl:copy-of
select="ct:end" /> <xsl:copy-of select="ct:extension" />
</xsl:otherwise> </xsl:choose> </xsl:copy> </xsl:template> <xsl:template
match="tns:identity"> <xsl:copy> <xsl:copy-of select="@*" /> <xsl:apply-
templates select="tns:name" /> <xsl:copy-of select="tns:email" />
<xsl:choose> <xsl:when test="count(ct:delimiter) > 0"> <xsl:copy-of
select="ct:delimiter[1]" /> <xsl:copy-of select="ct:delimiter[1]/following-
sibling::*" /> </xsl:when> <xsl:otherwise> <xsl:copy-of
select="ct:end" /> <xsl:copy-of select="ct:extension" />
</xsl:otherwise> </xsl:choose> </xsl:copy> </xsl:template> <xsl:template
match="tns:contactCard"> <xsl:copy> <xsl:copy-of select="@*" /> <xsl:apply-
templates select="tns:identity" /> <xsl:choose> <xsl:when
test="count(ct:delimiter) > 0"> <xsl:copy-of select="ct:delimiter[1]" />
</xsl:when> </xsl:choose> <xsl:choose> <xsl:when test="count(ct:delimiter)
> 1"> <xsl:copy-of select="ct:delimiter[2]" /> </xsl:when>
</xsl:choose> <xsl:choose> <xsl:when test="count(ct:delimiter) > 2">
<xsl:copy-of select="ct:delimiter[3]" /> <xsl:copy-of
select="ct:delimiter[3]/following-sibling::*" /> </xsl:when>
<xsl:otherwise> <xsl:copy-of select="ct:end" /> <xsl:copy-of
select="ct:extension" /> </xsl:otherwise> </xsl:choose> </xsl:copy>
</xsl:template> </xsl:stylesheet> ]]></categoryData>
</publicationRule>
<publicationRule ruleType="transformation" categoryName="calendarData"
containerId="1">
<categoryData><![CDATA[ <xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"
xmlns:tns="http://schemas.microsoft.com/2006/09/sip/calendarData" exclude-result-
prefixes="tns ct t"> <xsl:output method="xml" encoding="utf-8" indent="yes"/> <xsl:strip-
space elements="*" /> <xsl:template match="node()"> <xsl:apply-templates
select="//tns:calendarData"/> </xsl:template> <xsl:template match="tns:calendarData">

```

```

<xsl:copy>    <xsl:copy-of select="@*" />    <xsl:copy-of select="t:WorkingHours" />
<xsl:choose>    <xsl:when test="count(ct:delimiter) > 0">    <xsl:copy-of
select="ct:delimiter[1]" />    <xsl:copy-of select="ct:delimiter[1]/following-sibling::*"
/>    </xsl:when>    <xsl:otherwise>    <xsl:copy-of select="ct:end" />    <xsl:copy-of
select="ct:extension" />    </xsl:otherwise>    </xsl:choose>    </xsl:copy>    </xsl:template>
</xsl:stylesheet>    ]></categoryData>
    </publicationRule>
    <publicationRule ruleType="transformation" categoryName="calendarData"
containerId="100">
        <categoryData><![CDATA[ <xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"
xmlns:tns="http://schemas.microsoft.com/2006/09/sip/calendarData" exclude-result-
prefixes="tns ct t">    <xsl:output method="xml" encoding="utf-8" indent="yes"/>
<xsl:strip-space elements="*" />    <xsl:template match="node()">    <xsl:apply-templates
select="//tns:calendarData"/>    </xsl:template>    <xsl:template
match="tns:calendarData">    <xsl:copy />    </xsl:template> </xsl:stylesheet>
]]></categoryData>
    </publicationRule>
    <publicationRule ruleType="transformation" categoryName="calendarData"
containerId="200">
        <categoryData><![CDATA[ <xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"
xmlns:tns="http://schemas.microsoft.com/2006/09/sip/calendarData" exclude-result-
prefixes="tns ct t">    <xsl:output method="xml" encoding="utf-8" indent="yes"/>
<xsl:strip-space elements="*" />    <xsl:template match="node()">    <xsl:apply-templates
select="//tns:calendarData"/>    </xsl:template>    <xsl:template
match="tns:calendarData">    <xsl:copy>    <xsl:copy-of select="@*" />    <xsl:copy-of
select="t:WorkingHours" />    <xsl:copy-of select="tns:freeBusy" />
<xsl:choose>    <xsl:when test="count(ct:delimiter) > 0">    <xsl:copy-of
select="ct:delimiter[1]" />    <xsl:copy-of select="ct:delimiter[1]/following-
sibling::*" />    </xsl:when>    <xsl:otherwise>    <xsl:copy-of
select="ct:end" />    <xsl:copy-of select="ct:extension" />
</xsl:otherwise>    </xsl:choose>    </xsl:copy>    </xsl:template> </xsl:stylesheet>
]]></categoryData>
    </publicationRule>
    <publicationRule ruleType="transformation" categoryName="calendarData"
containerId="300">
        <categoryData><![CDATA[ <xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"
xmlns:tns="http://schemas.microsoft.com/2006/09/sip/calendarData" exclude-result-
prefixes="tns ct t">    <xsl:output method="xml" encoding="utf-8" indent="yes"/>
<xsl:strip-space elements="*" />    <xsl:template match="node()">    <xsl:apply-templates
select="//tns:calendarData"/>    </xsl:template>    <xsl:template
match="tns:calendarData">    <xsl:copy>    <xsl:copy-of select="@*" />    <xsl:copy-of
select="t:WorkingHours" />    <xsl:copy-of select="tns:freeBusy" />
<xsl:choose>    <xsl:when test="count(ct:delimiter) > 0">    <xsl:copy-of
select="ct:delimiter[1]" />    <xsl:copy-of select="ct:delimiter[1]/following-
sibling::*" />    </xsl:when>    <xsl:otherwise>    <xsl:copy-of
select="ct:end" />    <xsl:copy-of select="ct:extension" />
</xsl:otherwise>    </xsl:choose>    </xsl:copy>    </xsl:template> </xsl:stylesheet>
]]></categoryData>
    </publicationRule>
    <publicationRule ruleType="transformation" categoryName="calendarData"
containerId="400" preferredSelfConsumption="true" />
    <publicationRule ruleType="transformation" categoryName="calendarData"
containerId="32000">
        <categoryData><![CDATA[ <xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:t="http://schemas.microsoft.com/exchange/services/2006/types"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commontypes"

```

```

xmlns:tns="http://schemas.microsoft.com/2006/09/sip/calendarData" exclude-result-
prefixes="tns ct t"> <xsl:output method="xml" encoding="utf-8" indent="yes"/>
<xsl:strip-space elements="*" /> <xsl:template match="node()"> <xsl:apply-templates
select="//tns:calendarData"/> </xsl:template> <xsl:template
match="tns:calendarData"> <xsl:copy /> </xsl:template> </xsl:stylesheet>
]]</categoryData>
</publicationRule>
<publicationRule ruleType="transformation" categoryName="note" containerId="100">
<categoryData><![CDATA[ <xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commonTypes"
xmlns:tns="http://schemas.microsoft.com/2006/09/sip/note" exclude-result-prefixes="tns
ct"> <xsl:output method="xml" encoding="utf-8" indent="yes"/> <xsl:strip-space
elements="*" /> <xsl:template match="node()"> <xsl:apply-templates
select="//tns:note"/> </xsl:template> <xsl:template match="tns:note"> <xsl:copy
/> </xsl:template> </xsl:stylesheet> ]]]</categoryData>
</publicationRule>
<publicationRule ruleType="transformation" categoryName="note" containerId="200" />
<publicationRule ruleType="transformation" categoryName="note" containerId="300" />
<publicationRule ruleType="transformation" categoryName="note" containerId="400"
preferredSelfConsumption="true" />
<publicationRule ruleType="transformation" categoryName="note" containerId="32000">
<categoryData><![CDATA[ <xsl:stylesheet
xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:ct="http://schemas.microsoft.com/2006/09/sip/commonTypes"
xmlns:tns="http://schemas.microsoft.com/2006/09/sip/note" exclude-result-prefixes="tns
ct"> <xsl:output method="xml" encoding="utf-8" indent="yes"/> <xsl:strip-space
elements="*" /> <xsl:template match="node()"> <xsl:apply-templates
select="//tns:note"/> </xsl:template> <xsl:template match="tns:note"> <xsl:copy
/> </xsl:template> </xsl:stylesheet> ]]]</categoryData>
</publicationRule>
<publicationRule ruleType="transformation" categoryName="noteHistory"
containerId="200" />
<publicationRule ruleType="transformation" categoryName="noteHistory"
containerId="300" />
<publicationRule ruleType="transformation" categoryName="noteHistory"
containerId="400" preferredSelfConsumption="true" />
<publicationRule ruleType="transformation" categoryName="services" containerId="2"
preferredSelfConsumption="true" />
<publicationRule ruleType="transformation" categoryName="routing" containerId="0"
preferredSelfConsumption="true" />
<publicationRule ruleType="transformation"
categoryName="routing" containerId="100" />
<publicationRule ruleType="transformation"
categoryName="routing" containerId="200" />
<publicationRule ruleType="transformation"
categoryName="routing" containerId="300" />
<publicationRule ruleType="transformation"
categoryName="routing" containerId="400" />
<publicationRule ruleType="transformation"
categoryName="userProperties" containerId="1" preferredSelfConsumption="true" />
<publicationRule ruleType="transformation" categoryName="userInformation"
containerId="1" preferredSelfConsumption="true" />
<publicationRule ruleType="transformation" categoryName="alerts" containerId="1"
preferredSelfConsumption="true" />
<publicationRule ruleType="transformation" categoryName="rccOptions" containerId="1"
preferredSelfConsumption="true" />
<publicationRule ruleType="transformation" categoryName="otherOptions"
containerId="1" preferredSelfConsumption="true" />
<publicationRule ruleType="transformation" categoryName="categoryPublicationManifest"
containerId="1" preferredSelfConsumption="true" />
<publicationRule ruleType="bootstrap" categoryName="state" containerId="32000">
<instanceId type="constant" value="0" />
<expireType type="static" />
<categoryData><![CDATA[ <state
xmlns="http://schemas.microsoft.com/2006/09/sip/state"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" manual="false"
xsi:type="aggregateState"> <availability>18500</availability> </state> ]]]</categoryData>
</publicationRule>
<publicationRule ruleType="bootstrap" categoryName="services" containerId="32000">
<instanceId type="constant" value="0" />
<expireType type="static" />

```

```

        <categoryData><![CDATA[ <services
xmlns="http://schemas.microsoft.com/2006/09/sip/service" />    ]]></categoryData>
    </publicationRule>
    <publicationRule ruleType="bootstrap" categoryName="routing" containerId="32000">
        <instanceId type="constant" value="0" />
        <expireType type="static" />
        <categoryData><![CDATA[ <routing name="rtcdefault" version="1"
xmlns="http://schemas.microsoft.com/02/2006/sip/routing"> <preamble> <flags
name="clientflags" value="block" /> </preamble> </routing>    ]]></categoryData>
    </publicationRule>
    <publicationRule ruleType="bootstrap" categoryName="routing" containerId="0">
        <instanceId type="constant" value="0" />
        <expireType type="static" />
        <categoryData><![CDATA[ <routing name="rtcdefault" version="1"
minSupportedClientVersion="4.0.0.0"
xmlns="http://schemas.microsoft.com/02/2006/sip/routing"> <preamble> <flags
name="clientflags" value="" /> <wait name="total" seconds="20" /> </preamble>
</routing>    ]]></categoryData>
    </publicationRule>
    <publicationRule ruleType="bootstrap" categoryName="noteHistory" containerId="100">
        <instanceId type="constant" value="0" />
        <expireType type="static" />
        <categoryData><![CDATA[ <note xmlns="http://schemas.microsoft.com/2006/09/sip/note"
/>    ]]></categoryData>
    </publicationRule>
    <publicationRule ruleType="bootstrap" categoryName="noteHistory" containerId="32000">
        <instanceId type="constant" value="0" />
        <expireType type="static" />
        <categoryData><![CDATA[ <note xmlns="http://schemas.microsoft.com/2006/09/sip/note"
/>    ]]></categoryData>
    </publicationRule>
</publicationList>
</categoryPublicationManifest>
</provisionGroup>
<provisionGroup name="ServerConfiguration">

<lisInternalUrl>https://clientmainse.fabrikam.com:443/locationinformation/liservice.svc</lisI
nternalUrl>

<absInternalServerUrl>https://clientmainse.fabrikam.com:443/abs/handler</absInternalServerUrl
>

<absExternalServerUrl>https://clientmainse.fabrikam.com:4443/abs/handler</absExternalServerUr
l>

<abwqInternalUrl>https://clientmainse.fabrikam.com:443/groupexpansion/service.svc</abwqIntern
alUrl>

<abwqExternalUrl>https://clientmainse.fabrikam.com:4443/groupexpansion/service.svc</abwqExter
nalUrl>

<dlxInternalUrl>https://clientmainse.fabrikam.com:443/groupexpansion/service.svc</dlxInternal
Url>

<dlxExternalUrl>https://clientmainse.fabrikam.com:4443/groupexpansion/service.svc</dlxExterna
lUrl>
    <dlxEnabled>true</dlxEnabled>

<updatesServerInternalUrl>https://clientmainse.fabrikam.com:443/RequestHandler/ucdevice.upx</
updatesServerInternalUrl>

<updatesServerExternalUrl>https://clientmainse.fabrikam.com:4443/RequestHandlerExt/ucdevice.u
px</updatesServerExternalUrl>
    <updatesServerEnabled>true</updatesServerEnabled>
    <ucPortRangeEnabled>false</ucPortRangeEnabled>
    <ucMinMediaPort>5350</ucMinMediaPort>
    <ucMaxMediaPort>5389</ucMaxMediaPort>
    <ucMinSipDynamicPort>7100</ucMinSipDynamicPort>
    <ucMaxSipDynamicPort>7102</ucMaxSipDynamicPort>

```

```

<ucMinAudioPort>5350</ucMinAudioPort>
<ucMaxAudioPort>5389</ucMaxAudioPort>
<ucMinVideoPort>5350</ucMinVideoPort>
<ucMaxVideoPort>5389</ucMaxVideoPort>
<ucMinAppSharingPort>5350</ucMinAppSharingPort>
<ucMaxAppSharingPort>5389</ucMaxAppSharingPort>
<ucMinFileTransferPort>5350</ucMinFileTransferPort>
<ucMaxFileTransferPort>5389</ucMaxFileTransferPort>
<ucPC2PCAVEncryption>RequireEncryption</ucPC2PCAVEncryption>
<ucMaxVideoRateAllowed>VGA-600K</ucMaxVideoRateAllowed>
<qosEnabled>>false</qosEnabled>
<enableInCallQoS>>false</enableInCallQoS>
<inCallQoSIntervalSeconds>35</inCallQoSIntervalSeconds>
<ucDiffServVoice>40</ucDiffServVoice>
<ucVoice802_1p>0</ucVoice802_1p>
<ucEnforcePinLock>>true</ucEnforcePinLock>
<ucMinPinLength>6</ucMinPinLength>
<ucPhoneTimeOut>10</ucPhoneTimeOut>
<ucExchangeMWIPoll>3</ucExchangeMWIPoll>
<ucEnableSIPSecurityMode>High</ucEnableSIPSecurityMode>
<ucEnableUserLogging>>true</ucEnableUserLogging>
<loggingLevel>Medium</loggingLevel>
<enableBWPolicyCheck>>true</enableBWPolicyCheck>

<poolUri>sip:clientmainse.fabrikam.com@fabrikam.com;gruu;opaque=srvr:HomeServer:5duhvIdz_FOht
o7rkSU2gAAA</poolUri>

<mrasUri>sip:clientmain_int.fabrikam.com@fabrikam.com;gruu;opaque=srvr:MRAS:lvmY26x3w1WbYUtPi
viXXgAA</mrasUri>
  <qosUri>sip:clientmainse.fabrikam.com@fabrikam.com;gruu;opaque=srvr:QoS:N7bgtXm_dla-
s17uU81BIWAA</qosUri>

<callParkServerUri>sip:clientmainse.fabrikam.com@fabrikam.com;gruu;opaque=srvr:Microsoft.Rtc.
Applications.Cps:4H28w6olDlm9KG3n15tyBQAA</callParkServerUri>

<responseGroupServiceInternalURL>https://clientmainse.fabrikam.com:443/RgsClients/AgentServic
e.svc</responseGroupServiceInternalURL>

<responseGroupServiceExternalURL>https://clientmainse.fabrikam.com:4443/RgsClients/AgentServi
ce.svc</responseGroupServiceExternalURL>

<responseGroupServiceInternalAgentURL>https://clientmainse.fabrikam.com:443/RgsClients/Tab.as
px</responseGroupServiceInternalAgentURL>

<responseGroupServiceExternalAgentURL>https://clientmainse.fabrikam.com:4443/RgsClients/Tab.a
spx</responseGroupServiceExternalAgentURL>

<botSipUriForTestCall>sip:clientmainse.fabrikam.com@fabrikam.com;gruu;opaque=srvr:Microsoft.R
tc.Applications.TestBot:Hzhw8lv8gF-49c74fEDnRgAA</botSipUriForTestCall>
  <ucLocationProfile>DefaultProfile</ucLocationProfile>

<focusFactoryUri>sip:alice@fabrikam.com;gruu;opaque=app:conf:focusfactory</focusFactoryUri>
  <voiceMailUri>sip:alice@fabrikam.com;opaque=app:voicemail</voiceMailUri>
  <requireFIPSCompliantMedia>>false</requireFIPSCompliantMedia>
</provisionGroup>
<provisionGroup name="ucPolicy">
  <instance>
    <property name="name">DefaultPolicy</property>
    <property name="AllowSimultaneousRinging">>true</property>
    <property name="AllowCallForwarding">>true</property>
    <property name="EnableDelegation">>true</property>
    <property name="EnableTeamCall">>true</property>
    <property name="EnableCallPark">>true</property>
    <property name="EnableCallTransfer">>true</property>
    <property name="EnableMaliciousCallTrace">>true</property>
    <property name="enableBWPolicyOverride">>false</property>
    <property name="UcEnabled">>true</property>
    <property name="VoiceDeploymentMode">OnPrem</property>
  </instance>

```

```

</provisionGroup>
<provisionGroup name="userSetting">
  <disableFileTransfer>>false</disableFileTransfer>
  <ucUserLocationProfile>DefaultProfile</ucUserLocationProfile>
</provisionGroup>
<provisionGroup name="persistentChatConfiguration" >
  <propertyEntryList >
    <property name="EnablePersistentChat" >true</property>
    <property name="DefaultPersistentChatPoolUri"
>sip:mgcdogl.fabrikam.com@fabrikam.com;gruu;opaque=svr:GroupChat:k_7TLwJGsVy7jWMyTAtJcwAA</p
roperity>
    <property name="PersistentChatPoolUris"
>sip:server1.fabrikam.com@fabrikam.com;gruu;opaque=svr:GroupChat:k_7TLwJGsVy7jWMyTAtJcwAA
sip:server2.fabrikam.com@fabrikam.com;gruu;opaque=svr:GroupChat:l_7TLwJGsVy7jWMyTAtJcwBB
</property>
    <property name="PersistentChatWebManagerUriInt"
>https://pool0.fabrikam.com:123/PersistentChat/RM</property>
    <property name="PersistentChatWebManagerUriExt"
>https://ext.fabrikam.com:1234/PersistentChat/RM</property>
  </propertyEntryList>
</provisionGroup>
<provisionGroup name="publicProviders" >
  <propertyEntryList >
    <property name="PublicProviders" >
      <![CDATA[<PublicProviders>< /PublicProviders>]]>
    </property>
  </propertyEntryList>
</provisionGroup>
</provisionGroupList>

```

4.2.3 Client-to-Server Delegated Provisioning Request

In the following request, user bob@fabrikam.com sends an **application/vnd-microsoft-roaming-provisioning-v2+xml** request to the server on behalf of his delegator alice@fabrikam.com.

```

SUBSCRIBE sip:alice@fabrikam.com SIP/2.0
Via: SIP/2.0/TLS 172.24.32.149:53990
Max-Forwards: 70
From: <sip:bob@fabrikam.com>;tag=77399b3fb7;epid=e48fdaa231
To: sip:alice@fabrikam.com
p-session-on-behalf-of: sip:alice@fabrikam.com
Call-ID: dcb605d5e73e44ccbe84dbe4886d210f
CSeq: 1 SUBSCRIBE
Contact: <sip:bob@fabrikam.com;opaque=user:epid:A3zr0x5 sVSbfPZ-4 JkrgAA;gruu>
User-Agent: UCCAPI/4.0.7577.0 OC/4.0.7577.0 (Microsoft Communicator 2010)
Event: vnd-microsoft-provisioning-v2
Accept: application/vnd-microsoft-roaming-provisioning-v2+xml
Supported: com.microsoft.autoextend
Supported: ms-benotify
Proxy-Require: ms-benotify
Supported: ms-piggyback-first-notify
Expires: 0
Proxy-Authorization: Kerberos qop="auth", realm="SIP Communications Service",
opaque="224773FA", targetname="sip/clientmainse.fabrikam.com", crand="571d3235", cnum="4",
response="040400ffffffff0000000000000000d2a625d0f476403d15d58285"
Content-Type: application/vnd-microsoft-roaming-provisioning-v2+xml
Content-Length: 625
<provisioningGroupList xmlns="http://schemas.microsoft.com/2006/09/sip/provisioninggroupList"
subnet="172.24.32.0">
  <provisioningGroup name="endpointConfiguration" />
  <provisioningGroup name="locationPolicy" />
  <provisioningGroup name="mediaConfiguration" />
  <provisioningGroup name="meetingPolicy" />
  <provisioningGroup name="presencePolicyV2" />

```

```

    <provisioningGroup name="privacyPublicationGrammar" />
    <provisioningGroup name="publicationGrammar" />
    <provisioningGroup name="ServerConfiguration" />
    <provisioningGroup name="ucPolicy" />
    <provisioningGroup name="userSetting" />
    <provisioningGroup name="publicProviders" />
</provisioningGroupList>

```

4.3 Automatically Updating Client to a Server-Compatible Version

4.3.1 Processing Incoming Register Request

The following example is a typical REGISTER request with a **User-Agent** header specifying the UAC version.

```

REGISTER sip:production.cdmedia.tv SIP/2.0
From: <sip:alice@fabrikam.com>;tag=90a265ed3f;epid=a0ae9e6cb0
To: <sip:alice@fabrikam.com>
CSeq: 1 REGISTER
Call-ID: 6b2dec5dfd9a4eaf90180bb2f8f699ea
Via: SIP/2.0/TLS 10.2.50.42:1649
Max-Forwards: 70
Contact: <sip:10.2.50.42:1649;transport=tls;ms-opaque=40e5133a3a>;methods="INVITE,
MESSAGE, INFO, OPTIONS, BYE, CANCEL, NOTIFY, ACK, REFER,
BENOTIFY";proxy=replace;+sip.instance="<urn:uuid:C1589240-E660-594F-A8B3-4B55F358CBC1>"
User-Agent: UCCAPI/3.5.6890.0 OC/3.5.6890.0 (Microsoft Office Communicator 2007 R2)
Supported: gruu-10, adhoclist, msrtc-event-categories
ms-keep-alive: UAC;hop-hop=yes
Event: registration
Content-Length: 0

```

4.3.2 Sending 200 OK Response (Action: Allow client)

The following code is an example of a 200 OK REGISTER response sent back when the server allows the UAC.

```

SIP/2.0 200 OK
From: <sip:alice@fabrikam.com>;tag=90a265ed3f;epid=a0ae9e6cb0
To: <sip:alice@fabrikam.com>
CSeq: 1 REGISTER
Call-ID: 6b2dec5dfd9a4eaf90180bb2f8f699ea
ms-keep-alive: UAS; tcp=no; hop-hop=yes; end-end=no; timeout=300

Proxy-Authentication-Info: Kerberos
rspauth="602306092A864886F71201020201011100FFFFFFFF9964699C1D00E6ACA44AEBD821D187D2",
srand="9F9AB255", snum="1", opaque="B3762FF0", qop="auth",
targetname="sip/ABC.fabrikam.com", realm="SIP Communications Service"
Via: SIP/2.0/TLS 10.2.50.42:1649;ms-received-port=1649;ms-received-cid=100
Contact: <sip:10.2.50.42:1649;transport=tls;ms-opaque=40e5133a3a;ms-received-
cid=100>;expires=7200;+sip.instance="<urn:uuid:c1589240-e660-594f-a8b3-
4b55f358cbcl>";gruu="sip:pdawitt5@production.cdmedia.tv;opaque=user:epid:QJJYwWDMt1mos0tV
81jLwQAA;gruu"
Expires: 7200
presence-state: register-action="added";primary-cluster-type="central";is-connected-to-
primary="yes"
Allow-Events: vnd-microsoft-provisioning,vnd-microsoft-roaming-contacts,vnd-microsoft-
roaming-ACL,presence,presence.wpending,vnd-microsoft-roaming-self,vnd-microsoft-
provisioning-v2
Supported: adhoclist
Server: RTC/3.5

```


Supported: msrtc-event-categories
Content-Length: 0

4.3.3 Sending 403 Forbidden Response (Action: Block Client with Prompt)

The following code is an example of a 403 Forbidden response sent back when the server blocks the client. In this case, the UAC displays a prompt to the user indicating that the version of the client needs to be updated.

```
SIP/2.0 403 Forbidden
From: <sip:alice@fabrikam.com>;tag=90a265ed3f;epid=a0ae9e6cb0
To: sip:alice@fabrikam.com
CSeq: 1 REGISTER
Call-ID: 6b2dec5dfd9a4eaf90180bb2f8f699ea
Proxy-Authentication-Info: Kerberos
rspauth="602306092A864886F71201020201011100FFFFFFFF553E0C6CD0DF29D9C9DF1A9132706429",
srand="E7D4FFEC", snum="1", opaque="F2225600", qop="auth", targetname="
sip/ABC.fabrikam.com ", realm="SIP Communications Service"
Via: SIP/2.0/TLS 10.2.50.42:1656;ms-received-port=1656;ms-received-cid=100
Warning: 310 lcs.microsoft.com "You are currently not using the recommended version of
the client"
ms-diagnostics: 17002;reason="Invalid Request";source="DB19-
LCT.Production.cdmedia.tv";appName="ClientVersionFilter"
Content-Length: 0
```

4.3.4 Sending 403 Forbidden Response (Action: Block Client with Static URL)

The following code is an example of a 403 Forbidden response sent back when the server blocks the client. In this case, the user can get the newer version of the client from the following link specified in the **Warning** header.

```
https://fabrikam.com/ClientUpdate.html
```

```
SIP/2.0 403 Forbidden
From: <sip:alice@fabrikam.com>;tag=90a265ed3f;epid=a0ae9e6cb0
To: sip:alice@fabrikam.com
CSeq: 1 REGISTER
Call-ID: 6b2dec5dfd9a4eaf90180bb2f8f699ea
Proxy-Authentication-Info: Kerberos
rspauth="602306092A864886F71201020201011100FFFFFFFF553E0C6CD0DF29D9C9DF1A9132706429",
srand="E7D4FFEC", snum="1", opaque="F2225600", qop="auth", targetname="
sip/ABC.fabrikam.com ", realm="SIP Communications Service"
Via: SIP/2.0/TLS 10.2.50.42:1656;ms-received-port=1656;ms-received-cid=100
Warning: 311 lcs.microsoft.com "https://fabrikam.com/ClientUpdate.html"
ms-diagnostics: 17002;reason="Invalid Request";source="DB19-
LCT.Production.cdmedia.tv";appName="ClientVersionFilter"
Content-Length: 0
```

4.3.5 Sending 403 Forbidden Response (Action: Block Client with Dynamic URL)

The following code is an example of a 403 Forbidden response sent back when the server blocks the client. In this case, the client uses the dynamic URL to construct the following URL, and then automatically downloads a newer version of the client.

```
https://server.vdomain.com/AutoUpdate/Int/Handler/update.aspx?folder=OCv3.1&lang=en-
us&arch=x32&flavor=pu&build=chk
```

```

SIP/2.0 403 Forbidden
From: <sip:alice@fabrikam.com>;tag=90a265ed3f;epid=a0ae9e6cb0
To: sip:alice@fabrikam.com
CSeq: 1 REGISTER
Call-ID: 6b2dec5dfd9a4eaf90180bb2f8f699ea
Proxy-Authentication-Info: Kerberos
rspauth="602306092A864886F71201020201011100FFFFFFFF553E0C6CD0DF29D9C9DF1A9132706429",
srand="E7D4FFEC", snum="1", opaque="F2225600", qop="auth", targetname="
sip/ABC.fabrikam.com ", realm="SIP Communications Service"
Via: SIP/2.0/TLS 10.2.50.42:1656;ms-received-port=1656;ms-received-cid=100
Warning: 312 lcs.microsoft.com
"https://fabrikam.com/ClientUpdate/update.aspx?folder=OCv3.1"
ms-diagnostics: 17002;reason="Invalid Request";source="DB19-
LCT.Production.cdmedia.tv";appName="ClientVersionFilter"
Content-Length: 0

```

4.3.6 Sending 200 OK Response (Action: Allow Client with Dynamic URL)

The following code is an example of a 200 OK response sent back. In this case, the client uses the dynamic URL to construct the following URL, and then downloads a newer version of the client.

```

https://server.vdomain.com/AutoUpdate/Int/Handler/update.aspx?folder=OCv3.1&lang=en-
us&arch=x32&flavor=pu&build=chk

SIP/2.0 200 OK
From: <sip:alice@fabrikam.com>;tag=90a265ed3f;epid=a0ae9e6cb0
To: sip:alice@fabrikam.com
CSeq: 1 REGISTER
Call-ID: 6b2dec5dfd9a4eaf90180bb2f8f699ea
ms-keep-alive: UAS; tcp=no; hop-hop=yes; end-end=no; timeout=300
Proxy-Authentication-Info: Kerberos
rspauth="602306092A864886F71201020201011100FFFFFFFF104A8EDFA12CC095AAA663DF55824DCB",
srand="5E058A82", snum="1", opaque="E0DA4F74", qop="auth", targetname="sip/DB19-
LCT.Production.cdmedia.tv", realm="SIP Communications Service"
Via: SIP/2.0/TLS 10.2.50.42:1699;ms-received-port=1699;ms-received-cid=200
Contact: <sip:10.2.50.42:1699;transport=tls;ms-opaque=89eaec8ad6;ms-received-
cid=200>;expires=7200;+sip.instance="urn:uuid:c1589240-e660-594f-a8b3-
4b55f358cbcl";gruu="sip:alice@fabrikam.com;opaque=user:epid:QJYwWDMTlmos0tV81jLwQAA;gruu"
Expires: 7200
presence-state: register-action="added";primary-cluster-type="central";is-connected-to-
primary="yes"
Allow-Events: vnd-microsoft-provisioning,vnd-microsoft-roaming-contacts,vnd-microsoft-
roaming-ACL,presence,presence.wpending,vnd-microsoft-roaming-self,vnd-microsoft-provisioning-
v2
Supported: adhoclist
Server: RTC/3.5
Supported: msrtc-event-categories
Warning: 312 lcs.microsoft.com "https://fabrikam.com/Client
/update.aspx?folder=OCv3.1"
ms-diagnostics: 17002;reason="Invalid Request";source="DB19-
LCT.Production.cdmedia.tv";appName="ClientVersionFilter"
Content-Length: 0

```

4.3.7 Sending 200 OK Response (Action: Allow Clients with Upgrade)

The following code is an example of a 200 OK response sent back when the server allows the client. In this case, the client connections to an update service to find the latest version.

```

SIP/2.0 200 OK
From: <sip:alice@fabrikam.com>;tag=90a265ed3f;epid=a0ae9e6cb0
To: sip:alice@fabrikam.com
CSeq: 1 REGISTER
Call-ID: 6b2dec5dfd9a4eaf90180bb2f8f699ea

```

```
Proxy-Authentication-Info: Kerberos
rspauth="602306092A864886F71201020201011100FFFFFFFF553E0C6CD0DF29D9C9DF1A9132706429",
srand="E7D4FFEC", snum="1", opaque="F2225600", qop="auth", targetname="
sip/ABC.fabrikam.com ", realm="SIP Communications Service"
Via: SIP/2.0/TLS 10.2.50.42:1656;ms-received-port=1656;ms-received-cid=100
Warning: 314 lcs.microsoft.com "Check Microsoft Update for the newest
version.";appName="ClientVersionFilter"
Content-Length: 0
```

4.3.8 Sending 403 Forbidden (Action: Block Clients with Upgrade)

The following code is an example of a 403 Forbidden response sent back when the server blocks the client. In this case, the client connections to an update service to find the latest version.

```
SIP/2.0 403 Forbidden
From: <sip:alice@fabrikam.com>;tag=90a265ed3f;epid=a0ae9e6cb0
To: sip:alice@fabrikam.com
CSeq: 1 REGISTER
Call-ID: 6b2dec5dfd9a4eaf90180bb2f8f699ea
Proxy-Authentication-Info: Kerberos
rspauth="602306092A864886F71201020201011100FFFFFFFF553E0C6CD0DF29D9C9DF1A9132706429",
srand="E7D4FFEC", snum="1", opaque="F2225600", qop="auth", targetname="
sip/ABC.fabrikam.com ", realm="SIP Communications Service"
Via: SIP/2.0/TLS 10.2.50.42:1656;ms-received-port=1656;ms-received-cid=100
Warning: 314 lcs.microsoft.com "Check Microsoft Update for the newest
version.";appName="ClientVersionFilter"
Content-Length: 0
```

4.3.9 Sending 200 OK Response (Action: Allow Client with Static URL)

The following code is an example of a 200 OK REGISTER response sent back by the server. In this case, the user can get the newer version of the client from the following link specified in the **Warning** header.

```
https://fabrikam.com/ClientUpdate.html

SIP/2.0 200 OK
From: <sip:alice@fabrikam.com>;tag=90a265ed3f;epid=a0ae9e6cb0
To: sip:alice@fabrikam.com
CSeq: 1 REGISTER
Call-ID: 6b2dec5dfd9a4eaf90180bb2f8f699ea
ms-keep-alive: UAS; tcp=no; hop-hop=yes; end-end=no; timeout=300
Proxy-Authentication-Info: Kerberos
rspauth="602306092A864886F71201020201011100FFFFFFFF104A8EDFA12CC095AAA663DF55824DCB",
srand="5E058A82", snum="1", opaque="E0DA4F74", qop="auth", targetname="sip/DB19-
LCT.Production.cdmedia.tv", realm="SIP Communications Service"
Via: SIP/2.0/TLS 10.2.50.42:1699;ms-received-port=1699;ms-received-cid=200
Contact: <sip:10.2.50.42:1699;transport=tls;ms-opaque=89eac8ad6;ms-received-
cid=200>;expires=7200;+sip.instance="urn:uuid:c1589240-e660-594f-a8b3-
4b55f358cbc1>";gruu="sip:alice@fabrikam.com;opaque=user:epid:QJJYwWDMTlmos0tV81jLwQAA;gruu"
Expires: 7200
presence-state: register-action="added";primary-cluster-type="central";is-connected-to-
primary="yes"
Allow-Events: vnd-microsoft-provisioning,vnd-microsoft-roaming-contacts,vnd-microsoft-
roaming-ACL,presence,presence.wpending,vnd-microsoft-roaming-self,vnd-microsoft-provisioning-
v2
Supported: adhoclist
Server: RTC/3.5
Supported: msrtc-event-categories
Warning: 313 lcs.microsoft.com "https://fabrikam.com/ClientUpdate.html"
ms-diagnostics: 17002;reason="Invalid Request";source="DB19-
LCT.Production.cdmedia.tv";appName="ClientVersionFilter"
```

Content-Length: 0

5 Security

5.1 Security Considerations for Implementers

These extensions have the same security considerations as those natively defined for SIP.

In addition, an SA is required to be established and maintained between the UAC and the SIP registrar using the NTLM or the Kerberos authentication protocol, as described in [\[MS-SIPAE\]](#) section 3.2.

5.2 Index of Security Parameters

None.

6 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs.

- Microsoft Office Communications Server 2007
- Microsoft Office Communicator 2007
- Microsoft Office Communications Server 2007 R2
- Microsoft Office Communicator 2007 R2
- Microsoft Lync 2010
- Microsoft Lync Server 2010
- Microsoft Skype for Business (formerly Lync 2013)
- Microsoft Lync Server 2013
- Skype for Business
- Skype for Business Server

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

[<1> Section 2.1](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync 2010, Lync Server 2010: IPv6 is not supported.

[<2> Section 2.2.1.4](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<3> Section 2.2.1.4](#): Supported in Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2 only.

[<4> Section 2.2.1.5](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<5> Section 2.2.1.5](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<6> Section 2.2.1.6](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<7> Section 2.2.1.6](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<8> Section 2.2.1.7](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.

<9> [Section 2.2.1.7](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.

<10> [Section 2.2.1.8](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<11> [Section 2.2.1.8](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<12> [Section 2.2.1.9](#): Office Communications Server 2007: Server header value is the literal string "RTC/3.0". Office Communications Server 2007 R2: Server header value is the literal string "RTC/3.5".

<13> [Section 2.2.1.12](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<14> [Section 2.2.1.13](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This message body is not supported.

<15> [Section 2.2.1.13](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This message body is not supported.

<16> [Section 2.2.1.13](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<17> [Section 2.2.2.1](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<18> [Section 2.2.2.1](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<19> [Section 2.2.2.1](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<20> [Section 2.2.2.4](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<21> [Section 2.2.2.4](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<22> [Section 2.2.2.4](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<23> [Section 2.2.2.4](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<24> [Section 2.2.2.4](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<25> [Section 2.2.2.4](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<26> [Section 2.2.2.4](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync 2010: This behavior is not supported.

<27> [Section 2.2.2.4](#): Supported in Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2 only.

<28> [Section 2.2.2.5](#): Office Communications Server 2007, Office Communicator 2007: The response is a server error, based on an incompatible UAC version.

<29> [Section 2.2.2.5](#): Office Communications Server 2007, Office Communicator 2007: The response is a server error, based on an incompatible UAC version.

<30> [Section 2.2.2.5](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<31> [Section 2.2.2.5](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<32> [Section 2.2.2.5](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<33> [Section 2.2.2.5](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<34> [Section 2.2.2.5](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<35> [Section 2.2.2.5](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync 2010: This behavior is not supported.

<36> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007: This behavior is not supported.

Office Communications Server 2007 R2, Office Communicator 2007 R2: Only the following properties are supported:

```
|-- lisInternalUrl
|-- absInternalServerUrl
|-- absExternalServerUrl
|-- abwqInternalUrl
|-- abwqExternalUrl
|-- dlxInternalUrl
|-- dlxExternalUrl
|-- dlxEnabled
|-- updatesServerInternalUrl
|-- updatesServerExternalUrl
|-- updatesServerEnabled
|-- organization
|-- consoleDownloadInternalUrl
|-- consoleDownloadExternalUrl
|-- ucPortRangeEnabled
|-- ucMinMediaPort
|-- ucMaxMediaPort
|-- ucMinSipDynamicPort
|-- ucMaxSipDynamicPort
|-- ucMinAudioPort
|-- ucMaxAudioPort
|-- ucMinVideoPort
|-- ucMaxVideoPort
|-- ucMinAppSharingPort
|-- ucMaxAppSharingPort
|-- ucMinFileTransferPort
|-- ucMaxFileTransferPort
|-- ucPC2PCAVEncryption
|-- ucMaxVideoRateAllowed
|-- qosEnabled
|-- enableInCallQoS
|-- inCallQoSIntervalSeconds
|-- ucDiffServVoice
|-- ucVoice802_1p
|-- ucEnforcePinLock
|-- ucMinPinLength
|-- ucPhoneTimeout
```



```
|-- ucExchangeMWIPoll
|-- ucEnableSIPSecurityMode
|-- ucEnableUserLogging
|-- loggingLevel
|-- enableBWPolicyCheck
|-- poolUri
|-- mrasUri
|-- qosUri
|-- callParkServerUri
|-- responseGroupServiceInternalURL
|-- responseGroupServiceExternalURL
|-- responseGroupServiceInternalAgentURL
|-- responseGroupServiceExternalAgentURL
|-- botSipUriForTestCall
|-- cwaInternalUri
|-- cwaExternalUri
|-- ucLocationProfile
|-- focusFactoryUri
|-- voiceMailUri
```

<37> [Section 2.2.2.5.1](#): Supported in Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2 only.

<38> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007: This value is not supported.

<39> [Section 2.2.2.5.1](#): Supported in Office Communicator 2007, Office Communicator 2007 R2 only.

<40> [Section 2.2.2.5.1](#): Supported in Office Communicator 2007, Office Communicator 2007 R2 only.

<41> [Section 2.2.2.5.1](#): Supported in Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2 only.

<42> [Section 2.2.2.5.1](#): Supported in Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2 only.

<43> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007: This behavior is not supported.

<44> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007: This behavior is not supported.

<45> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007: This behavior is not supported.

<46> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007: This behavior is not supported.

<47> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007: This behavior is not supported.

<48> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007: This behavior is not supported.

<49> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007: This behavior is not supported.

<50> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007: This behavior is not supported.

<71> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<72> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<73> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.

<74> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.

<75> [Section 2.2.2.5.1](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.

<76> [Section 2.2.2.5.2](#): Supported in Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2 only.

<77> [Section 2.2.2.5.2](#): Supported in Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2 only.

<78> [Section 2.2.2.5.2](#): Supported in Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2 only.

<79> [Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: Optional/Ignored.

<80> [Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: Optional/Ignored.

<81> [Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: Optional/Ignored.

<82> [Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: Optional/Ignored.

<83> [Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007: A UAC sending a request with this parameter receives a server error, based on an incompatible UAC version.

<84> [Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: Optional/Ignored.

<85> [Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: Optional/Ignored.

<86> [Section 2.2.2.5.2](#): Supported in Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2 only.

<87> [Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<88> [Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

<89> [Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<90> Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<91> Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.

[<92> Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.

[<93> Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.

[<94> Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.

[<95> Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<96> Section 2.2.2.5.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<97> Section 2.2.2.5.3](#): Supported in Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2 only.

[<98> Section 2.2.2.5.3](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<99> Section 2.2.2.5.3](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.

[<100> Section 2.2.2.5.3](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.

[<101> Section 2.2.2.5.4](#): Office Communications Server 2007, Office Communicator 2007: The **ProvisionGroupType** and **ProvisionGroup** element are not supported.

[<102> Section 2.2.2.5.4](#): Office Communications Server 2007, Office Communicator 2007: This structure is not supported.

[<103> Section 2.2.2.5.4](#): Office Communications Server 2007, Office Communicator 2007: This structure is not supported.

[<104> Section 2.2.2.5.4](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: Ignored.

[<105> Section 2.2.2.5.4](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<106> Section 2.2.2.5.4](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: **visibleContainers**: The list of containers for the member is the same as the visible list of containers present in the containers element.

[<107> Section 2.2.2.5.5](#): Office Communications Server 2007, Office Communicator 2007: This behavior is not supported.

- <108> [Section 2.2.2.5.5](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.
- <109> [Section 2.2.2.5.6](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This value is not supported.
- <110> [Section 2.2.2.5.6](#): Supported by Microsoft Outlook 2010 or Microsoft Outlook 2010 Service Pack 1 (SP1) with Lync Server 2010, or Microsoft Outlook 2013 with Lync Server 2013 only.
- <111> [Section 2.2.2.5.6](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.
- <112> [Section 2.2.2.5.6](#): Supported by Microsoft OneNote 2010 with Lync Server 2010 only.
- <113> [Section 2.2.2.5.6](#): Supported by Outlook 2010, Outlook 2010 SP1, and Outlook 2013 only.
- <114> [Section 2.2.2.5.6](#): Supported by Microsoft Exchange Server 2013 only.
- <115> [Section 2.2.2.5.6](#): This setting applies only to Lync Server 2013 when Skype for Business (formerly Lync 2013) is deployed in a Virtualized Environment. When enabled, this setting dictates that Lync Server 2013 connects with the Skype for Business (formerly Lync 2013) Virtualization Plugin (if deployed on the access device) to provide the user with a high-quality audio and video experience.
- <116> [Section 2.2.2.5.6](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This value is not supported.
- <117> [Section 2.2.2.5.7](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.
- <118> [Section 2.2.2.5.8](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.
- <119> [Section 2.2.2.5.9](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.
- <120> [Section 2.2.2.5.10](#): Office Communications Server 2007, Office Communicator 2007: A UAC sending a request that includes this element receives a server error, based on an incompatible UAC version.
- <121> [Section 2.2.2.5.10](#): Office Communications Server 2007, Office Communicator 2007: A UAC sending a request that includes this element receives a server error, based on an incompatible UAC version.
- <122> [Section 2.2.2.5.11](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.
- <123> [Section 2.2.2.5.11](#): Supported by the Lync Server 2013 Persistent Chat Room Management Web Application only.
- <124> [Section 2.2.2.5.11](#): Supported by the Lync Server 2013 Persistent Chat Room Management Web Application only.
- <125> [Section 2.2.2.5.12](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.
- <126> [Section 2.2.2.5.12](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2, Lync Server 2010, Lync 2010: This behavior is not supported.

[<127> Section 3.1.1.1](#): Office Communicator 2007, Office Communicator 2007 R2: This behavior is not supported.

[<128> Section 3.1.1.1](#): Office Communicator 2007, Office Communicator 2007 R2: This behavior is not supported.

[<129> Section 3.1.1.1](#): Office Communicator 2007, Office Communicator 2007 R2: This behavior is not supported.

[<130> Section 3.1.1.5.1](#): Office Communicator 2007, Office Communicator 2007 R2: This behavior is not supported.

[<131> Section 3.1.1.5.1](#): Office Communicator 2007, Office Communicator 2007 R2: This behavior is not supported.

[<132> Section 3.1.1.5.1](#): Office Communicator 2007, Office Communicator 2007 R2: This behavior is not supported.

[<133> Section 3.1.1.5.1](#): Office Communicator 2007, Office Communicator 2007 R2: This behavior is not supported.

[<134> Section 3.1.1.5.1](#): Office Communicator 2007, Office Communicator 2007 R2: This behavior is not supported.

[<135> Section 3.1.1.5.1](#): Office Communicator 2007: This behavior is not supported.

[<136> Section 3.1.1.5.1](#): Office Communicator 2007: This behavior is supported.

[<137> Section 3.1.1.5.1](#): Office Communicator 2007: This behavior is not supported.

[<138> Section 3.1.1.5.1](#): Office Communicator 2007: This behavior is not supported.

[<139> Section 3.1.2.1](#): Office Communications Server 2007, Office Communications Server 2007 R2: This behavior is not supported.

[<140> Section 3.1.2.1](#): Office Communications Server 2007, Office Communications Server 2007 R2: This behavior is not supported.

[<141> Section 3.1.2.1](#): Office Communications Server 2007, Office Communications Server 2007 R2: This behavior is not supported.

[<142> Section 3.1.2.1](#): Office Communications Server 2007, Office Communications Server 2007 R2: This behavior is not supported.

[<143> Section 3.1.2.5.1](#): Office Communications Server 2007, Office Communications Server 2007 R2: This behavior is not supported.

[<144> Section 3.1.2.5.1](#): Office Communications Server 2007, Office Communications Server 2007 R2: This behavior is not supported.

[<145> Section 3.1.2.5.1](#): Office Communications Server 2007, Office Communications Server 2007 R2: This behavior is not supported.

[<146> Section 3.1.2.5.1](#): Office Communications Server 2007, Office Communications Server 2007 R2: This behavior is not supported.

[<147> Section 3.1.2.5.1](#): Office Communications Server 2007, Office Communications Server 2007 R2: This behavior is not supported.

[<148> Section 3.1.2.5.1](#): Office Communications Server 2007, Office Communications Server 2007 R2: This behavior is not supported.

[<149> Section 3.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<150> Section 3.2](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<151> Section 3.2.1](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<152> Section 3.2.1.5](#): Office Communicator 2007, Office Communicator 2007 R2: This behavior is not supported.

[<153> Section 3.4.1.5.2](#): Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is only supported in this version.

[<154> Section 3.4.2.5](#): Office Communications Server 2007: This behavior is not supported.

[<155> Section 3.4.2.5](#): Office Communications Server 2007: This behavior is not supported.

[<156> Section 3.4.2.5.5](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<157> Section 3.4.2.5.6](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<158> Section 3.4.2.5.7](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

[<159> Section 3.4.2.5.8](#): Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is only supported in this version.

[<160> Section 3.4.2.5.9](#): Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is only supported in these versions.

[<161> Section 3.5](#): Office Communications Server 2007, Office Communicator 2007, Office Communications Server 2007 R2, Office Communicator 2007 R2: This behavior is not supported.

7 Change Tracking

This section identifies changes that were made to this document since the last release. Changes are classified as New, Major, Minor, Editorial, or No change.

The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.
- The removal of a document from the documentation set.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the technical content of the document is identical to the last released version.

Major and minor changes can be described further using the following change types:

- New content added.
- Content updated.
- Content removed.
- New product behavior note added.
- Product behavior note updated.
- Product behavior note removed.
- New protocol syntax added.
- Protocol syntax updated.
- Protocol syntax removed.
- New content added due to protocol revision.
- Content updated due to protocol revision.
- Content removed due to protocol revision.
- New protocol syntax added due to protocol revision.
- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.
- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated**.

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.
- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact dochelp@microsoft.com.

Section	Tracking number (if applicable) and description	Major change (Y or N)	Change type
6 Appendix A: Product Behavior	Updated list of supported products.	Y	Content updated due to protocol revision.

8 Index

A

Abstract data model

automatic client update

[client](#) 83

[server](#) 87

basic registration

[client](#) 73

[server](#) 76

provisioning

[client](#) 81

[server](#) 82

remove binding from registrar

[client](#) 79

[server](#) 80

survivable mode

[server](#) 91

[Applicability](#) 13

Automatic client update

client

[abstract data model](#) 83

[higher-layer triggered events](#) 83

[initialization](#) 83

[local events](#) 87

[messaging processing](#) 83

[overview](#) 83

[sequencing rules](#) 83

[timer events](#) 87

[timers](#) 83

example

200 OK response

[allow](#) 112

[allow with dynamic URL](#) 114

[allow with static URL](#) 115

403 Forbidden response

[block with dynamic URL](#) ([section 4.3.5](#) 113,
[section 4.3.7](#) 114, [section 4.3.8](#) 115)

[block with prompt](#) 113

[block with static URL](#) 113

[process request](#) 112

server

[abstract data model](#) 87

[higher-layer triggered events](#) 87

[initialization](#) 87

[local events](#) 91

[messaging processing](#) 87

[sequencing rules](#) 87

[timer events](#) 91

[timers](#) 87

B

Basic registration

[client](#) 73

[abstract data model](#) 73

[higher-layer triggered events](#) 74

[initialization](#) 74

[local events](#) 76

[message processing](#) 74

[sequencing rules](#) 74

[timer events](#) 76

[timers](#) 74

[example](#) 93

[overview](#) 73

server

[abstract data model](#) 76

[higher-layer triggered events](#) 77

[initialization](#) 77

[local events](#) 79

[message processing](#) 77

[sequencing rules](#) 77

[timer events](#) 78

[timers](#) 76

unregistration

[example](#) 94

Basic unregistration

[example](#) 94

C

[Capability negotiation](#) 13

[Change tracking](#) 128

Client

automatic client update

[abstract data model](#) 83

[higher-layer triggered events](#) 83

[initialization](#) 83

[local events](#) 87

[message processing](#) 83

[overview](#) 83

[sequencing rules](#) 83

[timer events](#) 87

[timers](#) 83

[basic registration](#) 73

[abstract data model](#) 73

[higher-layer triggered events](#) 74

[initialization](#) 74

[local events](#) 76

[message processing](#) 74

[sequencing rules](#) 74

[timer events](#) 76

[timers](#) 74

Provisioning

[abstract data model](#) 81

[higher-layer triggered events](#) 81

[initialization](#) 81

[local events](#) 82

[message processing](#) 81

[overview](#) 81

[sequencing rules](#) 81

[timer events](#) 81

[timers](#) 81

remove binding from registrar

[abstract data model](#) 79

[higher-layer triggered events](#) 80

[initialization](#) 79

[message processing](#) 80

[overview](#) 79

[sequencing rules](#) 80

[timers](#) 79

D

Data model - abstract
automatic client update
[client](#) 83
[server](#) 87
basic registration
[server](#) 76
provisioning
[client](#) 81
[server](#) 82
remove binding from registrar
[client](#) 79
[server](#) 80
survivable mode
[server](#) 91
Data model - abstract
basic registration
[client](#) 73
Deregister
[example](#) 95
[messages](#) 17

E

Examples
automatic client update
200 OK response
[allow](#) 112
[allow with dynamic URL](#) 114
[allow with static URL](#) 115
403 Forbidden response
block with dynamic URL ([section 4.3.5](#) 113,
[section 4.3.7](#) 114, [section 4.3.8](#) 115)
[block with prompt](#) 113
[block with static URL](#) 113
[process request](#) 112
[basic registration](#) 93
[basic unregistration](#) 94
[deregistration](#) 95
[in-band provisioning](#) 96
[client to server request](#) 96
[delegated request](#) 111
[server to client response](#) 97
[registrar change notify](#) 95
[survivable mode notify](#) 95
Extensions to REGISTER Requests and Responses
message
[deregister NOTIFY request](#) 17
[extensions to Contact header](#) 17
[extensions to Server header](#) 17
[header field extensions](#) 15
[ms-keep-alive header field](#) 15
[Ms-Subnet header field syntax](#) 16
[P-Preferred-Registrar header field](#) 17
[Presence-State header field](#) 15
[SIP REGISTER request](#) 14
[SIP REGISTER response](#) 14
[survivable mode NOTIFY request](#) 18
[text/registration-event message](#) 18

F

[Fields - vendor-extensible](#) 13

G

[Glossary](#) 8

H

Higher-layer triggered events
automatic client update
[client](#) 83
[server](#) 87
basic registration
[client](#) 74
[server](#) 77
provisioning
[client](#) 81
[server](#) 82
remove binding from registrar
[client](#) 80
[server](#) 80
survivable mode
[server](#) 92

I

[Implementer - security considerations](#) 117
In-band provisioning
[example](#) 96
[client to server request](#) 96
[delegated request](#) 111
[server to client response](#) 97
In-band Provisioning Messages message ([section 2.2.2](#) 19, [section 2.2.2.3](#) 20)
[request](#) 19
[data model](#) 20
[response](#) 20
[data model](#) 22
[Index of security parameters](#) 117
[Informative references](#) 13
Initialization
automatic client update
[client](#) 83
[server](#) 87
basic registration
[client](#) 74
[server](#) 77
provisioning
[client](#) 81
[server](#) 82
remove binding from registrar
[client](#) 79
[server](#) 80
survivable mode
[server](#) 92
[Introduction](#) 8

L

Local events
automatic client update
[client](#) 87
[server](#) 91
basic registration
[client](#) 76
[server](#) 79
provisioning
[client](#) 82

- [server](#) 82
- remove binding from registrar
 - [server](#) 81

M

Message processing

- automatic client update
 - [client](#) 83
 - [server](#) 87
- basic registration
 - [client](#) 74
 - [server](#) 77
- client
 - [initialization](#) 81
 - [local events](#) 82
 - [timer events](#) 81
- provisioning
 - [client](#) 81
 - [server](#) 82
- remove binding from registrar
 - [client](#) 80
 - [server](#) 80
- server
 - [initialization](#) 82
 - [local events](#) 82
 - [timer events](#) 82

Messages

- Extensions to REGISTER Requests and Responses
 - [deregister NOTIFY request](#) 17
 - [extensions to Contact header](#) 17
 - [extensions to Server header](#) 17
 - [header field extensions](#) 15
 - [ms-keep-alive header field](#) 15
 - [Ms-Subnet header field syntax](#) 16
 - [P-Preferred-Registrar header field](#) 17
 - [Presence-State header field](#) 15
 - [SIP REGISTER request](#) 14
 - [SIP REGISTER response](#) 14
 - [survivable mode NOTIFY request](#) 18
 - [text/registration-event message](#) 18
- [In-band Provisioning Messages](#) 19
 - application/vnd-microsoft-roaming-provisioning-v2+xml document
 - [data model](#) 20
 - [request](#) 19
 - [data model](#) 20
 - [response](#) 20
 - [data model](#) 22
 - [syntax](#) 14
 - [transport](#) 14

N

[Normative references](#) 12

O

[Overview \(synopsis\)](#) 13

P

- [Parameters - security index](#) 117
- [Preconditions](#) 13
- [Prerequisites](#) 13

- [Product behavior](#) 118
- Provisioning
 - client
 - [abstract data model](#) 81
 - [higher-layer triggered events](#) 81
 - [initialization](#) 81
 - [overview](#) 81
 - [timers](#) 81
 - [example](#) 96
 - client
 - [delegated request](#) 111
 - [request](#) 96
 - server
 - [response](#) 97
 - [messages](#) 19
 - server
 - [abstract data model](#) 82
 - [higher-layer triggered events](#) 82
 - [initialization](#) 82
 - [timers](#) 82

R

- References
 - [informative](#) 13
 - [normative](#) 12
- Registrar
 - [example](#) 95
 - [messages](#) 17
 - removing a binding
 - [overview](#) 79
- Registration
 - example
 - [basic registration](#) 93
 - [basic unregistration](#) 94
 - [deregistration](#) 95
 - [notify for registrar change](#) 95
 - [survivable mode notify](#) 95
 - [Relationship to other protocols](#) 13
- Remove binding from registrar
 - client
 - [abstract data model](#) 79
 - [higher-layer triggered events](#) 80
 - [initialization](#) 79
 - [message processing](#) 80
 - [overview](#) 79
 - [sequencing rules](#) 80
 - [timers](#) 79
 - [overview](#) 79
 - server
 - [abstract data model](#) 80
 - [higher-layer triggered events](#) 80
 - [initialization](#) 80
 - [local events](#) 81
 - [message processing](#) 80
 - [sequencing rules](#) 80
 - [timer events](#) 81
 - [timers](#) 80

S

- Security
 - [implementer considerations](#) 117
 - [parameter index](#) 117
- Sequencing rules

- automatic client update
 - [client](#) 83
 - [server](#) 87
- basic registration
 - [client](#) 74
 - [server](#) 77
- client
 - [initialization](#) 81
- provisioning
 - [client](#) 81
 - [server](#) 82
- remove binding from registrar
 - [client](#) 80
 - [server](#) 80
- server
 - [initialization](#) 82
- Server
 - automatic client update
 - [abstract data model](#) 87
 - [higher-layer triggered events](#) 87
 - [initialization](#) 87
 - [local events](#) 91
 - [message processing](#) 87
 - [sequencing rules](#) 87
 - [timer events](#) 91
 - [timers](#) 87
 - basic registration
 - [abstract data model](#) 76
 - [higher-layer triggered events](#) 77
 - [initialization](#) 77
 - [local events](#) 79
 - [message processing](#) 77
 - [sequencing rules](#) 77
 - [timer events](#) 78
 - [timers](#) 76
 - provisioning
 - [abstract data model](#) 82
 - [higher-layer triggered events](#) 82
 - [initialization](#) 82
 - [local events](#) 82
 - [message processing](#) 82
 - [sequencing rules](#) 82
 - [timer events](#) 82
 - [timers](#) 82
 - remove binding from registrar
 - [abstract data model](#) 80
 - [higher-layer triggered events](#) 80
 - [initialization](#) 80
 - [local events](#) 81
 - [message processing](#) 80
 - [sequencing rules](#) 80
 - [timer events](#) 81
 - [timers](#) 80
 - survivable mode
 - [abstract data model](#) 91
 - [higher-layer triggered events](#) 92
 - [initialization](#) 92
 - [timers](#) 92
- [Standards assignments](#) 13
- Survivable mode
 - [example](#) 95
 - [messages](#) 18
 - [overview](#) 91
- server
 - [abstract data model](#) 91
 - [higher-layer triggered events](#) 92
 - [initialization](#) 92
 - [timers](#) 92
- [Syntax](#) 14

T

- Timer events
 - automatic client update
 - [client](#) 87
 - [server](#) 91
 - basic registration
 - [client](#) 76
 - [server](#) 78
 - provisioning
 - [client](#) 81
 - [server](#) 82
 - remove binding from registrar
 - [server](#) 81
- Timers
 - automatic client update
 - [client](#) 83
 - [server](#) 87
 - basic registration
 - [client](#) 74
 - [server](#) 76
 - provisioning
 - [client](#) 81
 - [server](#) 82
 - remove binding from registrar
 - [client](#) 79
 - [server](#) 80
 - survivable mode
 - [server](#) 92
- [Tracking changes](#) 128
- [Transport](#) 14
- Triggered events
 - automatic client update
 - [client](#) 83
 - [server](#) 87
 - basic registration
 - [client](#) 74
 - [server](#) 77
 - provisioning
 - [client](#) 81
 - [server](#) 82
 - remove binding from registrar
 - [client](#) 80
 - [server](#) 80
 - survivable mode
 - [server](#) 92

V

- [Vendor-extensible fields](#) 13
- [Versioning](#) 13