TTC STANDARD

JJ-90.21

Technical Specification of a Framework for Provider's SIP Networks

（English Edition）

Version 1

June 2, 2005

THE TELECOMMUNICATION TECHNOLOGY COMMITTEE

Introduction

This document provides the TTC original Standard formulated by the TTC Signaling working group.

The working group translated JJ-90.21 Japanese Version 1 (June 2, 2005) into English, and issued JJ-90.21 English Version on August 25, 2005.

In case of dispute, the original to be referred is the Japanese Edition of the text.

August 25, 2005

TTC Signaling Working Group

Contents

<References>

1. Relationship with international recommendations and standards

Nothing to note.

2. History of Revised Version

Revision	Date	Details of revision
Edition 1.0	June 2, 2005	Initial publication (Revised TS-1003 version1)

3. Reference documents

3.1. Normative References

[1] “SIP: Session Initiation Protocol”, TTC standard JF-IETF-RFC3261, version1, The Telecommunication Technologies Committee), June 2005.

[2] “Reliability of Provisional Responses in the Session Initiation Protocol (SIP)”, TTC standard JF-IETF-RFC3262, version1, The Telecommunication Technologies Committee, June 2005.

[3] “An Offer/Answer Model with the Session Description Protocol (SDP)”, TTC standard JF-IETF-RFC3264, version1, The Telecommunication Technologies Committee, June 2005.

[4] “SDP: Session Description Protocol”, TTC standard JF-IETF-RFC2327, version1, The Telecommunication Technologies Committee, June 2005.

[5] “A Privacy Mechanism for the Session Initiation Protocol (SIP)”, TTC standard JF-IETF-RFC3323, version1, The Telecommunication Technologies Committee, June 2005.

[6] Zimmerer, E., Peterson, J., Vemuri, A., Ong, L., Audet, F., Watson, M. and M. Zonoun, “MIME media types for ISUP and QSIG objects”, RFC 3204, Internet Engineering Task Force (IETF), December 2001.

[7] “The tel URI for Telephone Numbers)”, TTC standard JF-IETF-RFC3966, version1, The Telecommunication Technologies Committee, June 2005.

[8] Postel, J., “User Datagram Protocol”, RFC 768/STD 6, Internet Engineering Task Force (IETF), August 1980.

Postel, J., “Internet Protocol”, RFC 791/STD 7, Internet Engineering Task Force (IETF), September 1981.

[9] Freed, N. and Borenstein, N., “Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types”, RFC 2046, Internet Engineering Task Force (IETF), November 1996.

[10] “Inter-Carrier Interface based on ISUP”, TTC standard JJ-90.10, sixth edition, The Telecommunication Technologies Committee, April 2003.

[11] “Technical Specification on SIP to TTC ISUP Interworking”, TTC standard JF-IETF-RFC3398, version1, The Telecommunication Technologies Committee, June 2005.

[12] International Telecommunications Union, “The International Public Telecommunications Numbering Plan”, [13] ITU-T Recommendation E.164, ITU-T, 1997.

3.2. Informative References

[14] Postel, J., “Transmission Control Protocol”, RFC 793/STD 7, Internet Engineering Task Force (IETF), September 1981.

[15] Dierks, T. and C. Allen, “The TLS Protocol Version 1.0”, RFC 2246, Internet Engineering Task Force (IETF), January 1999.

[16] Stewart, R., Xiw, Q., Aharp, C., Schwarzbauer. H., Taylor, T., Rytina, I., Kalla, M., Zhang, L. and V. Paxson, “Stream Control Transmission Protocol”, RFC 2960, Internet Engineering Task Force (IETF), October 2000.

[17] Camarillo, G. and H. Schulzrinne, “Early Media and Ringback Tone Generation in the Session Initiation Protocol (SIP)”, RFC 3960, Internet Engineering Task Force (IETF), December 2004.
[18] Rosenberg, J., Peterson, J., Schulzrinne, H. and G. Camarillo, “Best Current Practices for Third Party Call Control (3pcc) in the Session Initiation Protocol (SIP)”, RFC 3725/BCP 85, Internet Engineering Task Force (IETF), Internet Engineering Task Force, April 2004.
[19] Johnston, A., Sparks, R., Cunnigham, C., Donovan, S. and K. Summers, “Session Initiation Protocol Service Examples”, draft-ietf-sipping-service-examples-08, Internet Engineering Task Force (IETF), Work in Progress, March 2005.
[20] Mahy, R., Biggs, B. and R. Dean, “The Session Initiation Protocol (SIP) ‘Replaces’ Header”, RFC 3891, Internet Engineering Task Force (IETF), September 2004.
[21] Mahy, R. and D. Petrie, “The Session Initiation Protocol (SIP) ‘Join’ Header”, RFC 3911, Internet Engineering Task Force (IETF), October 2004.
[22] Peterson, J., “Session Initiation Protocol (SIP) Authenticated Identity Body (AIB) Format”, RFC 3893, Internet Engineering Task Force (IETF), September 2004.
[23] Sparks, R., “Internet Media Type message/sipfrag”, RFC 3420, Internet Engineering Task Force (IETF), November 2002.
[24] Rosenberg, J., Schulzrinne, H. and R. Mahy, “An INVITE Initiated Dialog Event Package for the Session Initiation Protocol (SIP)”, draft-ietf-sipping-dialog-package-05, Internet Engineering Task Force (IETF), Work In Progress, November 2004.
[25] Willis, D. and B. Hoeneisen, “Session Initiation Protocol (SIP) Extension Header Field for Registering Non-Adjacent Contacts”, RFC 3327, Internet Engineering Task Force (IETF), December 2002.
[26] Willis, D. and B. Hoeneisen, “Session Initiation Protocol (SIP) Extension Header Field for Service Route Discovery During Registration”, RFC 3608, Internet Engineering Task Force (IETF), October 2003.
[27] International Telecommunications Union, “Visual telephone systems and equipment for local area networks which provide a non-guaranteed quality of service,” ITU-T Recommendation H.323, 2003.
[28] Andreasen, F. and B. Foster, “Media Gateway Control Protocol (MGCP) Version 1.0”, RFC 3435, Internet Engineering Task Force (IETF), January 2003.
[29] “Technical report on the Session Initiation Protocol (SIP)”, TTC technical report TR-1007, The Telecommunication Technologies Committee), March 2003.

4. Industrial property rights

Information regarding submittal of TTC’s “The Policy for the Handling of Industrial Property Rights” is available on TTC’s home page.

5. Contact

Signalling Working Group

1. Overview

1.1. Scope of this specification

This specification applies to networks (providers' SIP networks) where there is a possibility of calls being connected via a Inter-Network Interface between networks provided by providers that establish interconnections based on JJ-90.10 [10], and which use SIP (JF-IETF-RFC3261 [1]) to connect to other provider's networks and user terminals. Calls which this specification is applied to include not only calls that are actually connected via interconnection interfaces based on JJ-90.10, but also calls between SIP terminals through providers' SIP networks, etc.

The rules in this specification for media are only concerned with voice calls between networks conforming to JJ-90.10 [10],[1] and the rules for the media sent between terminals are beyond the scope of this specification. Also, the rules for SIP call control signal are to be applied regardless of the combination of the interworking networks finally used to establish the call.[2]

This specification imposes no limits on the capabilities of providers' SIP networks, and places no restrictions on the use of extensions as long as they are agreed upon between providers and conform to the SIP related specifications. Detailed specifications relating to specific interfaces and regulations relating to services provided in addition to basic call establishment as a provider's SIP network are provided in separate documents based on this specification.

1.2. Purpose and provisions of this specification

- This specification provides frameworks such as architectures and models for prescribing how services and interfaces associated with providers' SIP networks should be specified.
- The call control signal conditions covered by this specification are the signal processing conditions that are applied in common in relation to SIP according to JF-IETF-RFC3261 [1] and extensions thereof in cases where a provider's SIP network connects with another provider's SIP network.
- The conditions to be satisfied by providers' SIP networks in this specification relate to security and congestion control for the protection of interconnected networks, especially existing networks. These include the presence of functions for protecting the network from the effects of congestion by calls originated through the provider's SIP network. Similarly, regulations are provided for requirements that should be considered to achieve improved interconnection performance including future expansion of providers' SIP networks.
- The conditions to be satisfied by media such as voice signals in this specification relate to the media capabilities supported by the MGC/MG (Media Gateway Controller/Media Gateway) situated in providers' SIP networks to guarantee the connection of voice calls with the ISUP network. No particular restrictions are imposed on the media conditions of sessions between SIP UAs (user agents) established via a provider's SIP network.

1.3. Content of this specification

This specification defines the requirements to be satisfied by providers' SIP networks and the rules for the connection interfaces in order to establish connections between networks within the scope stated in section 1.1. The structure of this specification is as follows.

[1] The media scope may be expanded in response to subsequent studies or the application of additional interfaces.
[2] No restrictions are imposed on how SIP operations should be handled differently according to the provider's policy or to the interworked networks or protocols.

- Main body: Definition of terminology and the connection model. Requirements to be satisfied by providers' SIP networks, etc. Media capabilities to be supported by nodes situated in a provider's SIP network in order to guarantee the connection of voice calls with the ISUP network.
- Appendices: Contain the following reference information relating to the main body:
  - Notes on the transparency of SIP messages in providers' SIP networks (Appendix i)
  - SIP UA media capabilities (Appendix ii)
  - General properties relating to SIP media capability (Appendix iii)
  - Notes in the case that a SIP UA used by users who are managed by a provider's SIP network acquires its IP address dynamically (Appendix iv)
  - Guidelines on dealing with spoofed SIP URIs in From headers and ensuring uniqueness (Appendix v)

1.4. Terminology

The main terms used in the main body and appendices of this specification are defined here.

< Provider's SIP network >

A network that consists of SIP nodes under a certain level of control of a certain provider, and which establishes sessions with external networks and terminals via SIP nodes that transmit SIP messages and constitute a boundary. Provider's SIP network may be directly or indirectly interconnected with the networks provided by Carriers regulated by JJ-90.10 [10], either via the connection interface of its own network or via the other provider's network. This standard relates to provider's SIP network.

<SIP node>

A network entity that receives and transmits SIP messages. Refers to a node having the functions of an SIP UA in JF-IETF-RFC3261[1] (including an SIP terminal, B2BUA or MGC), or a node having the functions of an SIP proxy server (either stateful or stateless). Physically identical SIP nodes may operate logically as SIP UAs for some calls, and as SIP proxy servers for others.

<Session>

A media (e.g., voice) stream established by exchanging an SDP (Session Description Protocol) [4] by SIP messages via a connection interface.

<Call>

A relationship and state of end points and a network managed by the exchange of SIP messages via a connection interface starting with an Initial INVITE request.

<Initial INVITE request>

An INVITE request transmitted to set up a call and its associated session, recognized at the server side by the fact that it includes a To header with no To-tag parameters.

<Incoming call>

This term applies to a connection interface that uses a SIP; A call in the case that an Initial INVITE request is transmitted through this connection interface from other provider's network towards this provider's SIP network.

<Outgoing call>

This term applies to a connection interface that uses a SIP; A call in the case that an Initial INVITE request is transmitted through this connection interface from this provider's SIP network towards other provider's network.

<Session management SIP message>

A general term for SIP messages (requests and corresponding responses) exchanged in a dialog established by initial INVITE request and corresponding 1xx or 2xx responses except 100 (Trying) response. Includes re-INVITE messages

(INVITE messages with a To-tag parameter in the To header), PRACK messages, UPDATE messages and BYE messages.

<Adjacent SIP node>

A SIP node that exists on another provider's SIP network and transmits and receives SIP messages directly to and from this provider's SIP network via interface A (Fig. 1).

<Interworking SIP node>

A SIP node that exists on this provider's SIP network and transmits and receives SIP messages directly to and from another provider's SIP network via interface A (Fig. 1).

<Boundary>

A signaling node or a group of signaling nodes on local network side that is situated at the boundary between this provider's SIP network (local network) and other provider's network (including terminals).

<MGC>

Media Gateway Controller. In this specification, an SIP UA that is a signal node that exists in a provider's SIP network and interworks SIP and ISUP.

<MG>

Media Gateway. In this specification, a node that exists on a provider's SIP network and establishes a voice path between a circuit of GSTN and an IP voice media stream under the control of an MGC. When reference is made in to an MGC/MG in the main body of this document, the MGC and MG may be physically separate entities or the same entity.

<Anonymous URI>

A URI that is used when wishing to make the URI information anonymous. A specific format is the format <sip:anonymous@anonymous.invalid> as recommended in JF-IETF-RFC3323 [5].

<User(s) managed by a provider's SIP network>

The user(s) that a provider's SIP network must have the responsibility to identify at the boundary of the provider's SIP network when he/she originates a call.

<Connection interface>

A logical connection point related to a call control signal that exists between the provider's SIP network and other provider's network or user. In the main body of this document, these are used by labeling them according to the protocols used as the call control signals and categories (user connection interface or network connection interface) (see Table 1).

<User connection interface>

The category of connection interface between the provider's SIP network and the user managed by the provider's SIP network. Interface B in the provider's SIP network interconnection model (Fig. 1) is included in this category. As for interface A and interface C in the provider's SIP network interconnection model, the responsibility for identifying the call originator requested from the network beyond the connection interface, so they are not included in this category. The content of other connection interface categories prescribed at a later date might result in them being included in this category, but if the call originator is identified in another protocol network or in a more remote network, it is not included in this category.

<Network connection interface>

The category of connection interfaces other than user connection interfaces.

2. Interconnection model

2.1. Architecture model

Fig. 1 shows the interconnection architecture model referred to in this standard.

A provider's SIP network as covered by this standard is assumed to have the ability to be connected with an interconnection network conforming to JJ-90.10 [10] via an interface C belonging to local network or an interface C belonging to a provider's SIP network (other provider's network) connected to local network.

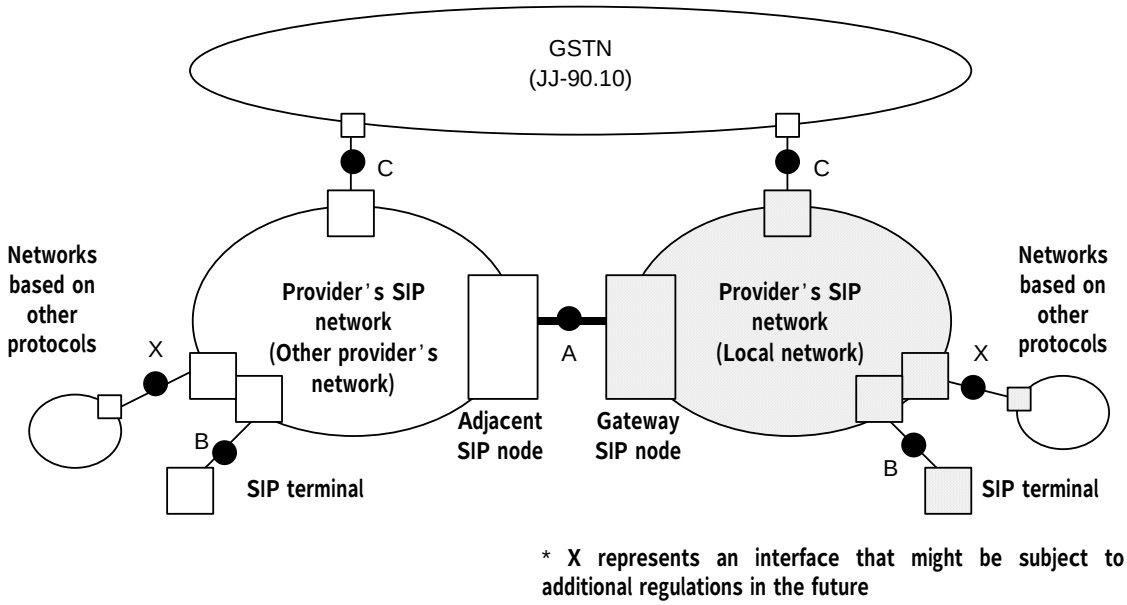

Fig. 1/JJ-90.21: Provider's SIPnetwork interconnection model

Table 1/JJ-90.21: Connection interface regulations in the interconnection model

Interface	Protocol	Boundary	Category
A	SIP	SIP proxy etc.	Network
B	SIP	SIP outbound proxy etc.	User
C	ISUP	MGC	Network

With regard to connection interface categories[3] other than those mentioned in Table 1, it is assumed that the typesof categories will be added when necessary, and that they will be covered either by this standard or by other reference materials.

2.2. Call model

Table 2 shows the call model patterns covered by this standard.[4]

[3] Specific examples include ITU-T H.323 and MGCP (Media Gateway Control Protocol) (RFC 3435 [28]).
[4] This does not include the case where SIP signals are closed inside a single provider's SIP network without passing through interface A, and the case of the interface C–interface A–interface C call pattern.

Note that a call which is originated in a provider's network, passes throught multiple SIP provider's networks to the local network via incoming interface A and also a call which is terminated in a provider's network, passes through multiple provider's SIP networks from the local network via outgoing interface A are within the scope of this specification.

Also, when new interface X are defined, the positioning of interface X in the call model is assumed to conform to interface B, unless stated otherwise.

Table 2/JJ-90.21: Call models in the interconnection model

	Call originating interface of another provider's SIP network	Incoming interface of the provider's SIP network	Outgoing interface of the provider's SIP network	Call terminating interface of another provider's SIP network
1.	B (or X)	A	C	–
2.	B (or X)	A	B (or X)	–
3.	B (or X)	A	A	C
4.	B (or X)	A	A	B (or X)
5.	C	A	B (or X)	–
6.	C	A	A	B (or X)
7.	–	B (or X)	A	C
8.	–	B (or X)	A	B (or X)
9.	–	C	A	B (or X)

3. Requirements relating to the provider's SIP network

3.1. Requirements relating to messages

3.1.1. SIP message transparency

When SIP messages are exchanged through a provider's SIP network between one interface A and another, or between an interface A and an interface B and it is possible to establish a call and a session, the degree of guarantee of information transparency of the SIP messages (see Appendix i) or the existense of such a guarantee are the essential requirements in relation to interconnectivity, including the various functions in the SIP and the future extensions. Accordingly, in the provider's SIP network, adequate information related to SIP message information transparency should be given to the interconnected peers when necessary based on each item in Appendix i.

Depending on the services and connection configurations provided by the provider's SIP network, it should be kept in mind that inforamtion transparency may not necessarily be uniquely determined even at the same interface.

3.1.2. Support for SIP extensions

To guarantee the future operation of negotiations on the use of extensions in calls through a provider's SIP network, with regard to cases where there is a node that performs different processing from the operation of a SIP proxy server conforming to JF-IETF-RFC3261 [1] in the provider's network, the following points must be kept in mind with regard to the provider's SIP network when it is not possible to reliably guarantee that the processing operations of this node

will not be affected by the realization of extensions specified by option-tag[5] including unrecognized ones (referred to as an unsupported option-tag below).[6]

Note that if all the SIP nodes in the provider's SIP network conform to the processing operations of an SIP proxy server conforming to JF-IETF-RFC3261 [1], then the following processing need not be performed.

- When the Require header included in a received SIP request includes an unsupported option-tag, the provider's SIP network SHOULD respond to this SIP request with a 420 (Bad Extension) response that includes the unsupported option-tag in the Unsupported header. The same applies to the Proxy-Require header.
- When a Supported header included in a received SIP request and 2xx response includes an unsupported option-tag, the Supported header included in the message forwarded in the provider's SIP network SHOULD NOT include the unsupported option-tag.
- When a Require header included in a received 421 (Extension Required) response[7] includes an unsupported option-tag, a 421 (Extension Required) response is not sent directly when transmitting an error response to the previous level in the provider's SIP network, but a response such as 400 (Bad Request) that does not include a Require header SHOULD be transmitted.[8]

3.1.3. Processing unrecognized headers/parameters

In the provider's SIP network, to ensure future extensibility, when forwarding an SIP message to the next hop in situations where unrecognized headers or unrecognized header parameters are received, processing SHOULD preferably continue as if these headers or header parameters did not exist.[9] Even when operating as an SIP UA, processing SHOULD preferably be continued by ignoring these headers or header parameters.

3.1.4. The message size of SIP requests

In cases where, for example, UDP is the only form of transport that can be used to transmit an SIP request to the next hop, there is a risk of network congestion or node congestion due to the effects of multiple fragmentation in the IP layer. To avoid these problems, in cases where fragmentation is expected to occur beyond a certain upper limit, the transfer of SIP requests to the next hop in the provider's SIP network MAY be rejected. In this case, it is RECOMMENDED to respond with a 513 (Message Too Large) response. When sending back an error response from the provider's SIP network with the message size given as the reason for rejection, it SHOULD be possible to be detected by the provider's SIP network of this situation.[10]

At the boundary of interface A, it MUST be possible to process messages of at least 1,300 bytes as stated in section 18.1.1 of JF-IETF-RFC3261 [1].

---

[5] A list of option-tags for SIP extensions is currently available in the option-tag part of the web page at http://www.iana.org/assignments/sip-parameters.

[6] When new extensions are prescribed, if it can be confirmed that the processing actions performed at the SIP node have no effect on these extensions, then it should be made possible to not perform the processing prescribed in this section on the same new option-tag.

[7] In RFC 3261, it is not recommended that the UAS requests support for a specific extension by sending back a 421 (Extension Required) response.

[8] This is to avoid repeated iterations of adding the option-tag specified by the UAC and retransmitting, whereupon the SIP node sends back a 420 (Bad Extension) response and resends it without the specified option-tag, whereupon a 421 (Extension Required) response is received.

[9] If it operates as an SIP proxy server, it will normally perform transparent forwarding. However, it may be deleted as a policy of the provider's SIP network in cases such as when it is required in a service provided by the network, or when it is clearly known in advance that these headers or header parameters clearly affect the interconnectivity. In such cases, the processing operations must be clarified according to the contents of section 3.1.1 and section 3.1.2.

[10] It is envisaged that conditions could be confirmed for exceeding the envisaged upper limit on the message size, and if necessary measures could be taken such as increasing the upper limit on message size.

It is strongly RECOMMENDED that an SIP UA is capable of receiving fragmented SIP packets.

3.1.5. Guaranteed delivery of 1xx responses

When a SIP is used to connect a call and a 1xx response is lost in transit route for whatever reason, it may become impossible to convey information to the calling user, such as a ring tone indicating that the callee is being alerted, or announcements generated by the network. Accordingly, measures SHOULD be taken to ensure that 1xx responses are reliably transferred in calls made at a provider's SIP network that is capable of connecting to a GSTN, including cases where interconnections are established with other provider's networks. In the SIP protocol, this can be accomplished with the 100rel option provided in JF-IETF-RFC3262 [2], but it MAY also be implemented by other means besides the 100rel option such as guaranteeing reliable transport such as TCP in all routes where messages are carried.

3.2. Identifying the originating user

When the provider's SIP network transmit the Initial INVITE request originated by a users managed by the provider's SIP network through the user connection interface, it must be possible to identify who originates the call. With regard to incoming calls through a network connection interface, it must be possible to reliably identify the user that originated the call either in the network connected by this connection interface or in a more remote network.

3.3. Media requirements

3.3.1. IP-side media conditions at the boundary of interface C

When the provider's SIP network has an interface C, exchanges SIP messages through interface A and establishes a session with a GSTN, the MGC/MG at the interface C boundary is expected to support at least the media capability profile shown in Table 3. The MGC/MG MUST also be able to engage in SIP negotiations conforming to JF-IETF-RFC3261 [1] or JF-IETF-RFC3264 [3].

The MGC/MG MAY have other capabilities besides those shown in the media capability profile of Table 3.

Table 3/JJ-90.21: Minimum guaranteed media capability profile of the interface C boundary

	Main category	Sub-category	Profile
1-1	SDP capability element (send)	Receiving IP address	IPv4, Unicast
1-2		Port number	Any port that is not well-known
1-3		Codec	Supports G.711 μLaw
1-4		Bandwidth	–
1-5		Packetization interval	20 or unspecified
1-6		Direction	`sendrecv` or not given
2-1	SDP capability element (receive)	Receiving IP address	IPv4, Unicast
2-2		Port number	Any port that is not well-known
2-3		Codec	Supports G.711 μLaw
2-4		Bandwidth	–
2-5		Packetization interval	20 or unspecified (20 ms spacing supported)
2-6		Direction	`sendrecv` or unspecified

	Main category	Sub-category	Profile
3-1	SDP capability element special values (send)	Receiving IP address (c= line: 0.0.0.0)	Not transmitted
3-2		Port number (m= line: 0)	Not transmitted
3-3		Bandwidth (b= line: 0)	Not transmitted
4-1	SDP capability element special values (receive)	Receiving IP address (c= line: 0.0.0.0)	MAY be impossible to process when received
4-2		Port number (m= line: 0)	Not expected to be received (multiple media are not established in the scope of interconnections between the main body document and ISUP)
4-3		Bandwidth (b=line: 0)	Not expected to be received
5-1	SDP format	Multi-part MIME body (offer)	Not transmitted
5-2		Multi-part MIME body (answer)	Not transmitted
5-3		No m= line SDP (offer)	Not transmitted
5-4		Multiple m= lines SDP (offer)	Not transmitted
5-5		Multiple PT (answer)	Not transmitted
6-1	SDP format (receive)	Multi-part MIME body (offer)	May be received (in case of multipart/mixed and when handling parameter of the Content-Disposition header corresponding to an unrecognizable Content-Type other than application/sdp is optional)
6-2		Multi-part MIME body (answer)	May be received (in case of multipart/mixed and when handling parameter of the Content-Disposition header corresponding to an unrecognizable Content-Type other than application/sdp is optional)
6-3		No m= line SDP (offer)	MAY not be received (In such case, send back an error response)
6-4		Multiple m= lines SDP (offer)	MAY not be received (however, if there is an m= line that can be supported, it SHOULD preferably be possible to receive it)
6-5		Multiple PT (answer)	Only a single PT MAY be supported.
7-1	Early Media (180/Media/Alert-Info)	Received/received/received	a. (ring tone)/b. (generate received media) (b. MAY be generated only when it can be judged that it is media from a reliable node in the provider's SIP network where expect to genetrate media[11])
7-2		Received/received/ not received	Same as above
7-3		Received/not received/received	a. (ring tone)
7-4		Received/not received/not received	a. (ring tone)
7-5		Not received/received/not received	d. (silence)

[11] See Appendix i of JF-IETF-RFC3398 [11].

	Main category	Sub-category	Profile
8-1	Call originating establishment procedure (offer/answer)	INVITE/2xx	Compatible
8-2		INVITE/1xx (100rel)	Compatible
8-3		2xx/ACK	Incompatible (INVITE request not transmitted without SDP)
8-4		1xx (100rel)/PRACK	Incompatible (INVITE request not transmitted without SDP)
9-1	Call receiving establishment procedure (offer/answer)	INVITE/2xx	Compatible
9-2		INVITE/1xx (100rel)	Compatible
9-3		2xx/ACK	MAY be incompatible
9-4		1xx (100rel)/PRACK	MAY be incompatible
10-1	Multiple dialog processing (existing/new)	Early/Early	–
10-2		Early/Confirm	Priority given to confirm dialog
10-3		Confirm/Confirm	Priority given to earlier confirm dialog
11-1	Session modification request transmission (State/request)	Confirmed/re-INVITE	Not transmitted
11-2		Confirmed/UPDATE	Not transmitted
11-3		Early/UPDATE (UAS)	MAY be transmitted
11-4		Early/UPDATE (UAC)	Not transmitted
11-5		Early/PRACK	Not transmitted
12-1	Session modification request reception (State/request)	Confirmed/re-INVITE	Processed if modified content is acceptable
12-2		Confirmed/UPDATE	Processed if modified content is acceptable (only when the Allow header contains UPDATE)
12-3		Early/UPDATE (UAS)	MAY not be processed
12-4		Early/UPDATE (UAC)	MAY not be processed
12-5		Early/PRACK	MAY not be processed
13-1	Session modification contents (send)	Receiving IP address	Not transmitted
13-2		Receiving port number	Not transmitted
13-3		Payload type modification	Not transmitted
13-4		Payload type deletion	Not transmitted
13-5		Media addition	Not transmitted
13-6		Media deletion	Not transmitted
13-7		Direction	Not transmitted
13-8		Received packet interval	Not transmitted

	Main category	Sub-category	Profile
14-1	Session modification contents (receive)	Receiving IP address	MAY be unchanged (SHOULD be changeable)
14-2		Receiving port number	MAY be unchanged
14-3		Payload type modification	MAY be unchanged
14-4		Payload type deletion	MAY be unchanged
14-5		Media addition	MAY be unchanged
14-6		Media deletion	MAY be unchanged
14-7		Direction	MAY be unchanged
14-8		Received packet interval	MAY be unchanged

3.4. Security requirements

The following requirements must be satisfied in the provider's SIP network.

3.4.1. Message privacy

In a provider's SIP network, third parties must not be able to see or tamper the contents of messages.

3.5. Congestion control requirements

In a provider's SIP network, the following requirements must be satisfied.

3.5.1. Functions for preventing the spread of congestion

In outgoing calls, according to its own judgment or that of the destination provider, it must be possible to automatically or manually restrict the transfer of Initial INVITE requests that fit a certain criteria as causes of congestion in order to prevent the spread of congestion.

3.5.2. Functions for call barred from specific users

To prevent the spread of network congestion due to malformed calls or unnecessarily large numbers of calls being made from a user connection interface, the originating provider's SIP network must be able to take subsequent practical steps to stop the transfer of Initial INVITE requests from the users it manages.

3.5.3. Upper limit on the number of simultaneous connection attempt calls/connected calls from a single user

In order to prevent network congestion from being caused by outgoing calls from (a single) user managed by a provider's SIP network from the user connection interface, it must be possible to set a finite upper limit on the number of simultaneous connection attempt calls/connected calls from a single user.

4. Interface A specifications

4.1. Scope of interface A specifications

The following sections of this chapter discuss the specifications relating to interface A in the interconnection model prescribed by section 2.

The connection interface conditions discussed here are applied not only when the call establishes a session with networks defined by JJ-90.10 [10], but also in other connection patterns (e.g., connections between SIP UAs).

4.2. Connection interface requirements

This section prescribes the basic connection requirements above the network layer relating to interface A.

Protocols[12] with requirements not specified in this section are not excluded from use, and MAY be used based on agreement between providers.

However, security must be considered in layers below the network layer so as not to process illegal messages resulting from processing abnormalities in DoS attacks, falsified caller addresses, and the like.

4.2.1. Network layer interface

Internet Protocol (IP) Version 4 (IPv4) (RFC 791/STD 7) [8] must be supported. This does not prohibit the use of Internet Protocol (IP) Version 6 (IPv6).

4.2.2. Transport layer interface

User Datagram Protocol (UDP) (RFC 768/STD 6 [8]) must be supported. Support for Transmission Control Protocol (TCP) is also RECOMMENDED. The use of TLS to resolve security issues with messages between providers' SIP networks on the basis of agreement between the two companies should not be prevented.

4.2.3. Application interface

Session Initiation Protocol (SIP) v2.0 (JF-IETF-RFC3261 [1]) shall be used.

4.3. SIP message requirements

4.3.1. Essential header configurations

Table 4 shows the essential header settings for Initial INVITE requests.

[12] E.g., Internet Protocol Version6 (IPv6), Transmission Control Protocol (TCP) [14], Transport Layer Security (TLS) [15], Stream Control Transmission Protocol (SCTP) [16]

Table 4/JJ-90.21: Essential header settings for Initial `INVITE` requests

Header	Incoming call (reception)	Outgoing call (transmission)
`To`	Values that conform to the format of JF-IETF-RFC3261 [1] must be permitted. If no `tag` parameter is provided, this header is recognized as an Initial `INVITE`.	Must conform to the format of JF-IETF-RFC3261 [1]. No `tag` parameter must be provided.
`From`	Values that conform to the format of JF-IETF-RFC3261 [1] must be permitted. It MAY be presumed that `tag` parameter is provided.	Must conform to the format of JF-IETF-RFC3261 [1].
`Contact`	Values that conform to the format of JF-IETF-RFC3261 [1] must be permitted.	Must conform to the format of JF-IETF-RFC3261 [1].
`Call-ID`	Values that conform to the format of JF-IETF-RFC3261 [1] must be permitted.	Must conform to the format of JF-IETF-RFC3261 [1].
`CSeq`	Values that conform to the format of JF-IETF-RFC3261 [1] must be permitted.	Must conform to the format of JF-IETF-RFC3261 [1].
`Via`	Values that conform to the format of JF-IETF-RFC3261 [1] must be permitted.	Must conform to the format of JF-IETF-RFC3261 [1].
`Max-Forward`	Values that conform to the format of JF-IETF-RFC3261 [1] must be permitted.	Must conform to the format of JF-IETF-RFC3261 [1].

4.3.2. Message routing header fields

This section shows the requirements for SIP URI format header fields in SIP messages from a gateway SIP node to an adjacent SIP node, which indicate the destination of SIP requests transferred after establishing a session for hop-by-hop message routing from the adjacent SIP node to the gateway SIP node.

<Requirements>

- The hostport part must be the IP address format[13] of the gateway SIP node transmitting the message, except when the maddr parameter is set.
- When the maddr parameter is set, the maddr parameter must satisfy the above condition.
- The port part MAY be set (port numbers other than 5060 are also permitted).
- The transport parameter MAY not be set.
- Other parameters such as the lr parameter[14] MAY be set if required.[15]
- The address specified by the abovementioned hostport part must be reachable from an adjacent SIP node.

<Applied header fields>

- The rec-route field of the Record-Route header at the head of the SIP request
- (When the gateway SIP node is a SIP UA and no Record-Route header is set) the hostport part of the Contact header in the SIP request.

[13] In the case of IPv4, this is expressed in ABNF format as follows: IPv4address = 1*3DIGIT "." 1*3DIGIT "." 1*3DIGIT "." 1*3DIGIT.
[14] However, it should still adapt to strict-routing when the lr parameter is inserted.
[15] The received parameters must be transparently forwarded.

- The rec-route field of Record-Route headers in 1xx or 2xx responses other than 100 (Trying) responses (i.e., Record-Route headers set when a corresponding request is received from an adjacent SIP node and transferred to a lower level).
- (When the gateway SIP node is a SIP UA and no Record-Route header is not set) the hostport part of Contact headers in 1xx or 2xx responses other than 100 (Trying) responses.

4.3.3. Session management SIP message requirements

Session management SIP messages must be transferred within a dialog according to the contents of the Route header.

4.4. Call destination URI specification scheme

The setting of a Request-URI in an Initial INVITE request is prescribed as follows.

4.4.1. user part

When the destinations of outgoing calls and incoming calls are telephone numbers that can be specified in the E.164 number format, it is RECOMMENDED that the user part of the SIP URI of the Request-URI in the Initial INVITE request uses the global-number-digits format of the tel URI prescribed by ABNF in JF-IETF-RFC3966 [7]. The use of a visual-separator is NOT RECOMMENDED. Table 5 shows the format corresponding to the call recipient numbers prescribed by JJ-90.10 [10].

Note that when global-number-digits includes one or more parameters (anything preceded by a semicolon), it MUST be possible to continue processing even when the contents of these parameters cannot be recognized, as long whatever follows each semicolon does not start with “m-”.

Table 5/JJ-90.21: Request-URI user part settings

Format	Conditions	Application
+ [Country code] [National Number]	Any country code except 81, up to 15 digits	International network calls
+81ABCDEFGHJ	A and B must not be 0	Regional fixed-line phone calls, IP phone calls (category A)
+81A0CDEFGHJK	A=2,7,8,9; C must not be 0	Mobile/PHS/wireless pager calls
+8150CDEFGHJK	C must not be 0	IP phone calls (category B)

This specification does not rule out the use of formats other than those shown in Table 5, including non-numerical formats, as long as they are agreed upon between the providers’ SIP networks. Note that the format of the user part MAY be specified separately in other technical specifications of connection interface regulations applied to interface A, including the content of the formats shown in Table 5.

4.4.2. hostport part

The hostport part of an Initial INVITE request in an outgoing call is set to the name of the host or domain to which the adjacent SIP node belongs (including the IP address format[16]).

Accordingly, the hostport part of an Initial INVITE request in an incoming call is expected to be set with the name of the domain or host to which the gateway SIP node belongs (including the IP address format[17]).

[16] In ABNF: IPv4address = 1*3DIGIT “.” 1*3DIGIT “.” 1*3DIGIT “.” 1*3DIGIT.

4.4.3. Option URI parameter part

The option URI parameters are ignored during processing.

4.5. Security requirements

The following requirements must be satisfied in the interface A between providers' SIP networks.

4.5.1. Message privacy

In the connection interface, third parties must not be able to see or tamper the contents of messages.

Messages received at a connection interface must be guaranteed to be messages from another provider's SIP network that can reasonably be expected to be reliable, and messages from illegal callers resulting from activities such as spoofing must not be received or processed.

4.5.2. Ensuring the validity of the From header

The URI in the From header of an Initial INVITE request transmitted through interface A from a provider's SIP network must not be a URI that indicates a different user from the user that generated the Initial INVITE request; in other words, it must not be possible to spoof other users.

With regard to the URI of the From header in an Initial INVITE request received through interface A from another provider's SIP network, if the other provider's SIP network conforms to the above mentioned content of the main body, then it can be regarded as not having been spoofed.

(End of document)

[17] In ABNF: IPv4address = 1*3DIGIT "." 1*3DIGIT "." 1*3DIGIT "." 1*3DIGIT.

Appendix i. Information transparency in the SIP network

i.1. Purpose of this appendix

When two SIP UAs establish a substantial session by exchanging SIP messages across a network comprising SIP proxy servers that conform to the operations prescribed by JF-IETF-RFC3261 [1], the information contained in the messages is basically transferred transparently between the SIP UAs via the intervening SIP proxy servers. Accordingly, the functions prescribed by JF-IETF-RFC3261 [1] and their associated SIP extensions are basically implemented on the premise of information transparency in the network. However, when this is implemented in a real network, there will be cases where the property of information transparency cannot be maintained for various reasons, in which case there are latent constraints.

This appendix classifies the properties of information transparency into different types, considers their effects, and mentions the content to be referred to when rationalizing the conditions required of a provider's SIP network conforming to the main body and the conditions presented to a network to which a provider's SIP network is connected, etc.

The content of this appendix includes representative items related to maintaining the property of information transparency, but this in no way implies that all items are included.

i.2. Overview

In this appendix, the following sort of model is considered. The terminology used here is obtained from the main body of this document.

The SIP UA on the caller sends out an Initial INVITE request, and taking this as an opportunity, the Initial INVITE request is transmitted to the SIP UA on the callee which establishes a substantial call through a set of intervening nodes (Fig. A.i). The SIP UA on the callee recognizes that an SIP dialog has been established by sending back a response including a tag parameter in the To header, and the SIP UA on the caller recognizes that an SIP dialog has been established by receiving a response including a tag parameter in the To header. When the established session is an RTP session on an IP network, IP packets might be exchanged directly between SIP UAs, or they might be terminated in IP layer and relayed one or more times by intervening nodes.

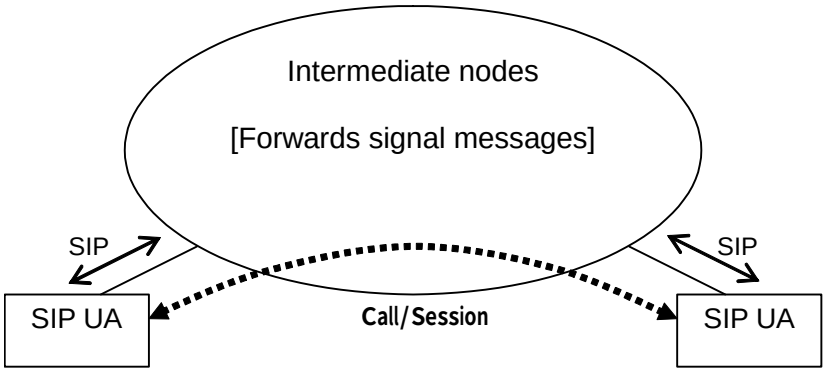

Fig. A.i/JJ-90.21: Message exchanges in an SIP session

In this appendix, section i.3 discusses the main factors associated with information transparency which is guaranteed when passing through intermediate nodes configured only from SIP proxy servers prescribed by JF-IETF-RFC3261

[1], but may not be guaranteed in certain types of node, and section i.4 rationalizes the constraints that arise when information transparency is not maintained.
The term “SIP UA” refers to the SIP UAs at both endpoints of a substantial call in Fig. A.i. The term “intermediate node” refers to any node in Fig. A.i (which may or may not be a SIP node).

i.3. Information transparency

The quoted text in each part of this section indicates the processing contents that are guaranteed between SIP UAs if the network is configured only from SIP proxies prescribed by JF-IETF-RFC3261 [1]. Any content that follows is there to supplement the content of the quoted text.
Also, at the end of each paragraph, an example is shown in which it is envisaged that operations different to the content of the quoted text will occur.

i.3.1. Dialog information transparency

“Information relating to a dialog[18] is sent transparently between SIP UAs, which share the same dialog information.”[19]

<Examples where information transparency is not assumed>

- When there is an intermediate node that is a complete B2BUA in which the value of the Call-ID header is not inherited.
- When an intermediate node amends(using a different tag from the original one) on receiving a response that includes several different To-tags resulting from forks or a sequential search by intermediate nodes.
- When an intermediate node is given a message with no To-tag or From-tag (differs depending on whether or not To-tags are used).

i.3.2. Message information transparency

“An SIP message in a dialog is transferred to the other SIP UA in the dialog regardless of the method used.”

In JF-IETF-RFC3261 [1], messages in a dialog are transferred to the other UA via the intermediate nodes within the limits of operating according to the Record Route function. Also, when Record Route is not performed, messages are exchanged directly without passing through the intermediate nodes. After a message has been transferred, it is still naturally possible that an error response such as 405 (Method Not Allowed) may be sent back by the SIP UA.
Note that even when messages conform to the Record Route function, a 420 (Bad Extension) error response might still be sent back from an intermediate node when the message includes a Proxy-Require header.

<Examples where information transparency is not assumed>

- When the intermediate nodes only support simple INVITE/ACK/CANCEL – BYE methods, and the transfer of other methods is not supported.
- When an intermediate node also terminates the RTP, it may absorb UPDATE requests or re-INVITE requests without transferring them in order to modify the session.

[18] Uniquely determined by the combination of Call-ID, local tag and remote tag.
[19] Strictly speaking, it might be included in the contents of the header information transparency in section i.3.4, but it is specially distinguished as having a special significance.

i.3.3. CSeq number information transparency

"CSeq numbers are transferred transparently between SIP UAs."

When an intermediate node generates a request message by itself without being prompted by the transmission of a request from an SIP UA, the CSeq number of the Initial INVITE request will become unmatched in mid transit, even if it was matching to start with.

<Examples where information transparency is not assumed>

- When a complete B2BUA in which CSeq numbers are not inherited exists as an intermediate node.
- When an intermediate node sends out a BYE request to terminate a session.
- When an intermediate node sends out an UPDATE request or a re-INVITE request for session management.
- When an intermediate node sends out an UPDATE request or a re-INVITE request to modify a session.

i.3.4. Header information transparency

"When the proxy column is neither m or d in Table 2 and Table 3 of JF-IETF-RFC3261 [1], or in the tables corresponding to the same content in the RFCs prescribing the other extension methods and extension headers, headers (including unrecognized headers) are forwarded transparently between SIP UAs."

<Examples where information transparency is not assumed>

- When there is an intermediate node that only transparently forwards specific headers.
- When there is an intermediate node that edits a specific header.
- When there is an intermediate node that does not transparently forward unrecognized headers.

i.3.5. Session information transparency

"The content of an SDP is transparently transferred between SIP UAs."[20]

<Examples where information transparency is not assumed>

- When there is an intermediate node or session relay node that transforms the address information (c= line) or port number (m= line) in order to establish an RTP session via an NAPT or the like.
- When there is an intermediate node that deletes part of the capability information (e.g., the payload type of the m= line) in order to restrict the session information of codecs and the like permitted in the network.

i.3.6. Message body information transparency

"The message body is transparently forwarded between SIP UAs regardless of the method."

<Examples where information transparency is not assumed>

- When there is an intermediate node that deletes the message body or part of multipart/mixed messages in order to restrict the information acceptable for forwarding on the network.
- When there is an intermediate node that deletes the message body in order to truncate the message to a length that can be handled.

[20] Strictly speaking, it might be included in the contents of the message body information transparency in section i.3.6, but it is specially distinguished as having particular significance.

i.3.7. Topological information transparency

"Once headers for the routing of SIP messages (e.g., Via, Route, Record-Route, Path (RFC 3327 [25]) and Service-Route (RFC 3608 [26]) headers[21]) have been added by an SIP proxy server, they are transparently forwarded to the SIP UA."

<Examples where information transparency is not assumed>

- When there is a B2BUA intermediate node that processes routing information completely separately.
- When there is an intermediate node that prevents the external disclosure of information relating to the number or arrangement of nodes in the provider's SIP network.

i.4. Constraints that arise when transparent forwarding is dropped

This section discusses the effects and constraints that result when a network fails to maintain transparent forwarding characteristics in the SIPs cited in each part of section i.3. However, it should be kept in mind that this is not a comprehensive list of all possible effects and constraints.

i.4.1. Dialog information transparency

<Anomalies during processing using dialog information>

The expected operation results are no longer obtained in processes that presume the retention of common dialog information between SIP UAs engaged in communication. As a specific example, it may become impossible to implement sequences such as Attended Transfer, Call Park, Call Pickup, 3-way Conference, 3rd Party Join, or Single Line Extension which are thought to use the Replaces header (RFC 3891 [20]) or Join header (RFC 3911 [21]) which currently being studied by the IETF.[22]

To provide compatibility with these extensions, it is not impossible to rewrite the header values in mid transit, but it is likely that problems will occur in adapting to future unknown extensions or implementational difficulties. In many cases since it is assumed that an option-tag has been set, it might be possible to adapt by deleting from Supported headers that include unknown option-tags during forwarding, but this is liable to damage the basic functional extensibility between SIP UAs.

Also, usage under a variety of different circumstances is imagined in a dialog event package currently being studied (draft-ietf-sipping-dialog-package-05 [24]), and problems are likely to arise when it is used in environments where dialogs are not guaranteed to be identical between UAs.

<Problems when using an AIB>

When using an Authenticated Identity Body (AIB) (RFC 3893 [22]) where ID information is included as a signature in the message body of an SIP message, the value of the Call-ID header used for replay protection is liable to be included in the signature calculation, so when the Call-ID header is modified, it might not be possible to use the AIB information properly.

<Problems in comparing log data>

It is envisaged that it will become harder to refer to the corresponding log when passing through an intermediate node that does not maintain dialog transparency, when comparing the log information between an SIP node and an SIP UA, and when there is no mapping information at an intermediate node.

---

[21] Since the Path and Service-Route headers are only expected to be used in a REGISTER request, they are not used in cases where only signals for establishing a call are considered.

[22] The service sequence example shown here is published in the Internet draft document draft-ietf-sipping-service-examples-07 [19].

i.4.2. Message information transparency

<The effects of message non-transparency on extensions>

When a SIP request is not forwarded in a dialog, it may become impossible to implement services between the SIP UAs that use this message. By making changes to, e.g., the Allow headers at the intermediate node in question, it is possible to prevent errors caused by failure to send out the SIP request in methods where it is not forwarded, but the use of functions is liable to be restricted.

i.4.3. CSeq number information transparency

<Problems when using AIB>

When an Authenticated Identity Body (AIB) (RFC 3893 [22]) containing signed ID information is contained in the message body of an SIP message, the value of the CSeq header for replay protection is liable to be included in the signature calculation, so when the value of the CSeq header changes, it may become impossible to handle the AIB information properly.

<Problems when using a 100rel option or the like>

When a Reliable Provisional Response (100rel) as prescribed by JF-IETF-RFC3262 [2] is used between SIP UAs, the RAck header includes a copied value of the CSeq number, so normal processing cannot be performed between the SIP UAs without giving suitable consideration.

Although specialized correspondences are possible in JF-IETF-RFC3262 [2], there is no guarantee that subsequent SIP extensions will not be able to use the values of CSeq headers in other headers or the like, and it is possible that normal processing may become impossible simply due to support only being provided for SIP UAs where sessions are established for new extensions.

i.4.4. Header information transparency

<The effects of header non-transparency on extensions>

When a header is not transparently forwarded, extensions that use this header are liable to not operate normally.

i.4.5. Session information transparency

<The effects on session performance exchange>

When information relating to SDP performance exchange is not transparently forwarded, it is possible that sessions might be established with lower performance than the session performance of the SIP UAs, or that a session may be impossible to establish regardless of the performance that the UAs are intrinsically capable of establishing.

i.4.6. Message body information transparency

<The effects of message body non-transparency on extensions>

When a message body is not transparently forwarded, it may not be possible to exchange information that uses this message body.

i.4.7. Topological information transparency

<Problems when using AIB>

When using an Authenticated Identity Body (AIB) (RFC 3893 [22]) including signed ID information in the message body of an SIP message, since the value of the Contact header is included in the signature calculation, it may not be possible to handle the AIB information properly when the value of the Contact header changes.

Appendix ii. The media capabilities of SIP UAs

ii.1. Overview

In SIP (JF-IETF-RFC3261 [1]), media sessions are established and managed through the exchange of SDP (Session Description Protocol: JF-IETF-RFC2327 [4]) on SIP messages based on a so-called "offer/answer" model (JF-IETF-RFC3264 [3]). This appendix summarizes the items that need to be considered when engaging in discussions relating to processing capabilities during SDP reception.

This appendix discusses the operations assumed in cases that are based on the RFCs they refer to (JF-IETF-RFC3264 [3] and JF-IETF-RFC2327 [4]).

Here, it is presumed that a "unicast" "RTP" media stream is established by an SDP exchange.

ii.2. SDP capability elements

The main SDP elements are shown in Table A.ii-1. It should be pointed out that this is not an exhaustive list of all the elements.

Also, of the elements listed in Table A.ii-2, mention is made of cases where there are values that have a special significance, such as zero values.

Table A.ii-1/JJ-90.21: SDP capability elements

	Content	SDP element	Additional notes
(1)	Receiving IP address	c= line	Processing must be possible both in the Session part and in the Media part.
(2)	Port number	m= line	
(3)	Codec	m= line and a=rtpmap	Must also be compatible with static payload types where there is no a=rtpmap.
(4)	Bandwidth	b= line	
(5)	Packetization interval	a=ptime	The transmitting side should conform to the value of ptime, but the receiving side should still be capable of reception at other Packetization intervals.
(6)	Direction	a= line	inactive/sendrecv/sendonly/recvonly

Table A.ii-2/JJ-90.21: Special values of SDP capability elements

	Content	SDP element	Special value	Significance	Additional notes
(1)	Receiving IP address	c= line	0.0.0.0	Media hold	An obsolete provision of RFC 2543. Not recommended for use in JF-IETF-RFC3264 [3]. (Specification with the direction attribute is recommended)
(2)	Port number	m= line	0	Media rejection/deletion	
(3)	Bandwidth	b= line	0	Rejection of media reception	RTCP reception also prevented.

As an expression of the capabilities of equipment installations, the SDP capability elements shown in Table A.ii-1 and Table A.ii-2 must determine whether or not transmission is possible, and if so, which processing operations are to be performed. In these tables, the terms “transmission” and “reception” refer to SDP transmission and reception that occurs independently of SDP offer/answer messages. However, when there is a notable difference from offer/answer messages, the details of this difference should be mentioned.

ii.3. SDP format

ii.3.1. Multi-part MIME body (offer or answer)

When a multipart/mixed MIME body is received and either each part can be processed by the SIP UA or the handling parameter in a Content-Disposition header is set to optional, then it should be possible for the included SDP to process it normally.

When there are no capabilities that can be processed, a 415 (Unsupported Media Type) response must be sent back with an Accept header that lists only the supported types (application/sdp, etc.).

ii.3.2. SDP (offer) with no m= line

When an SDP (Initial) offer is received with no m= line, it must be possible to respond with an SDP answer that has no m= line. This is sometimes used in the initial INVITE request in Third Party Call Control (RFC 3725 [18]). In this case it is expected that an m= line will be added by a normal re-INVITE request or UPDATE request.

ii.3.3. SDP (offer) with multiple m= lines

When an SDP is received that includes multiple m= lines, it must be possible to send back an answer in which the port numbers are set to zero except in the m= lines that are compatible.

ii.3.4. Receiving multiple payload types (answer)

When an offer is transmitted including multiple payload type values as compatible payload types in the m= line, an answer may be received including multiple payload type values in the m= line. This signifies that it is possible to switch freely between multiple payload types in a single session, so when switching is not possible, the SDP offer must be repeated by issuing a re-INVITE request or UPDATE request including only the payload type values that are actually desired to be used.

Table A.ii-3/JJ-90.21: SDP format

	SDP	Offer/answer	Additional notes
(1)	Multi-part MIME body	Offer	When a 415 response has been received, it should be possible to retry according to the content of this response and one’s own policy.
(2)		Answer	When a multi-part MIME cannot be recognized or includes a Content Type that is not set to `handling=optional`, a 415 response must be given including a suitable `Accept` header.
(3)	SDP with no `m=` line	Offer	May be used in 3pcc (RFC 3725 [18])
(4)	SDP with multiple `m=` lines	Offer	May be sent out from terminals with video capabilities, etc.
(5)	Reception of multiple payload types	Answer	If multiple payload types are not transmitted, then they are not received.

ii.4. Early media and local ring tones

General rules for early media and local ring tones are mentioned in JF-IETF-RFC3960 [17], which describes the processing that should be performed depending on whether or not a 180 (Ringing) response or an Early media (RTP packet) is received.

When a 180 (Ringing) response includes an Alert-Info header, information derived from the value of the Alert-Info header may be used as a local ring tone depending on its content. The following pattern is thus envisaged for the processing operations performed by an SIP UA with the ability to send out local ring tones:

(a) Use a local ring tone.

(b) Play back the received media.

(c) Access and generate the resource indicated by the Alert-Info header.

(d) Do not generate audio.

Table A.ii-4 shows the respective patterns and processing options for the reception of a 180 (Ringing) response, early media, and an Alert-Info header.

Table A.ii-4/JJ-90.21: Early media and local ring tones

	180	Media	Alert-Info	Processing details (selection)	Additional notes
(1)	Received	Received	Received	(a)/(b)/(c)	Reference [17] provides an example of a policy for (b), but in PSTN GW and the like, it might be a policy such as not generating media before the reception of a 2xx response.
(2)	Received	Received	Not received	(a)/(b)	
(3)	Received	Not received	Received	(a)/(c)	
(4)	Received	Not received	Not received	(a)	
(5)	Not received	Received	Not received	(b)/(d)	Same as (1)/(2)

ii.5. Session establishment

In establishing a session at the Initial INVITE transaction, there are a number of patterns in the procedure whereby offers and answers are exchanged.

ii.5.1. When initiating a call (when transmitting an Initial INVITE request)

Table A.ii-5 shows the procedure for establishing a session with an Initial INVITE transaction where the present situation can be assumed during call initiation. Offers from the called party will not occur as long as the calling party does not transmit an Initial INVITE request containing no SDP.

Table A.ii-5/JJ-90.21: Procedure for establishing a session when initiating a call

	Type	Offer	Answer	Additional notes
(1)	Calling party offer	INVITE	2xx	This is the most standard form
(2)		INVITE	1xx (100rel)	When 100rel is supported
(3)	Called party offer	2xx	ACK	Must be handled when there is no offer in the Initial INVITE
(4)		1xx (100rel)	PRACK	Must be handled when there is no offer in the Initial INVITE and when 100rel is supported

ii.5.2. When receiving a call (when receiving an Initial INVITE request)

Table A.ii-6 shows the procedure for establishing a session with an Initial INVITE transaction where the present situation can be assumed during call reception. Situations in which an INVITE request containing no SDP offer might be received are envisaged to include Third Party Call Control (RFC 3725 [18]).

Table A.ii-6/JJ-90.21: Procedure for establishing a session when receiving a call

	Type	Offer	Answer	Additional notes
(1)	Called party offer	INVITE	2xx	This is the most standard form
(2)		INVITE	1xx (100rel)	Must be handled during 100rel support (there might be no answer SDP in a 2xx response)
(3)	Calling party offer	2xx	ACK	Essential when there is no offer in the Initial INVITE (or for error disconnection)
(4)		1xx (100rel)	PRACK	Essential when transmitting a Reliable 1xx response if the Initial INVITE has no offer

ii.6. Processing multiple dialogs

When an SIP UA has sent out an Initial INVITE request, it is possible that multiple dialogs might be established by receiving responses including To-tags that are different from those received so far, in addition to the existing dialog. it is even possible that multiple existing dialogs may already have been established. Since multiple dialogs each have their own corresponding media, they must be suitably processed based on a policy.

Table A.ii-7/JJ-90.21: Processing multiple dialogs

	Existing dialog	New dialog	Required processing
(1)	Early dialog	Early dialog	It is possible to have a policy on which dialog to give priority to in terms of user interface processing based on criteria such as whether or not there is an SDP, what it contains, etc. However, when using `100rel`, since it is also possible that it might not contain an answer to a `2xx` response, it is preferable to either save all the session information or explicitly terminate the Early dialog by transmitting a BYE request. When there are no particular criteria on which to base this decision, it is recommended that priority is given to the newer dialog. (When there is no response, other actions such as forwarding may be considered.)
(2)	Early dialog	Confirmed dialog	The session is changed in the content of the confirmed dialog. The Early dialog may be explicitly terminated by transmitting a BYE request, or its contents may be discarded after 64×T1.
(3)	Confirmed dialog	Confirmed dialog	It is possible to have a policy on which dialog to give priority to based on criteria such the SDP (or whether to keep both dialogs running simultaneously). When either dialog is selected, the other dialog should preferably be explicitly terminated with a BYE request. (Simply failing to send back an ACK request will cause the `2xx` response to be re-sent.)

ii.7. Session modification

Session modification is requested by transmitting an SIP request that includes a modified SDP. When the content of the modification is accepted at the receiving end of the SIP request, it replies with a 2xx response including the SDP answer. If the content of the modification is not accepted at the receiving end of the SIP request, it must reply with a 488 (Not Acceptable Here) response, and the session and dialog saved at the time the modification was requested must be kept intact.

ii.7.1. Transmitting modification requests

Table A.ii-8 shows the categories envisaged when requesting a session modification.

In practical implementations, when an error response is received to an offer that includes a session modification after a dialog has been established, care should be taken to avoid terminating the session unnecessarily.

Table A.ii-8/JJ-90.21: Session modification request transmission categories

	State	Request	Content of processing
(1)	Confirmed	re-INVITE	Modification of a confirmed dialog.
(2)		UPDATE	Modification of a confirmed dialog. This is restricted to cases where the message from the callee includes an Allow header containing an UPDATE.
(3)	Early	UPDATE (UAS)	When modifying a session from the UAS side of an INVITE transaction with respect to an Early dialog.
(4)		UPDATE (UAC)	When modifying a session from the UAS side of an INVITE transaction with respect to an Early dialog.
(5)		PRACK	When modifying a session from the UAS side of an INVITE transaction with respect to an Early dialog. * When there is no answer in a 2xx response to a PRACK request, it is taken to mean that the session modification has failed (the offer was not recognized), and it may be necessary to adopt an implementation where processing is continued.

ii.7.2. Receiving modification requests

Table A.ii-9 shows the categories when receiving a session modification request, the conditions for receiving these requests, and the processing performed when modification is impossible.

Table A.ii-9/JJ-90.21: Session modification request reception categories

	State	Request	Reception conditions	Processing performed when modification not possible
(1)	Confirmed	re-INVITE	Normally liable to be received	488 response
(2)		UPDATE	Not received unless the Allow header contains UPDATE	488 response
(3)	Early	UPDATE (UAS)	Not received unless the Allow header of the INVITE request contains UPDATE	488 response
(4)		UPDATE (UAC)	Not received unless the Allow header of a reliable 1xx response contains UPDATE	488 response
(5)		PRACK	Not received unless in 100rel mode	488 response

ii.7.3. Content of modification

Table A.ii-10 shows the main session modification contents.

Table A.ii-10/JJ-90.21: Session modification content

	Modified element	Content of modification	Additional notes (purposes, etc.)
(1)	Receiving IP address	Modify the IP address in the `c=` line	May be needed when a terminal is moving, for example. In this case the `Contact` header may also change at the same time.
(2)	Receiving port number	Modify the port number in the `m=` line	May be needed in conjunction with changes of IP address, etc.
(3)	Payload type modification	Modify the payload type in the `m=` line (and the content of `a=rtpmap`)	May be needed when a codec is changed, etc.
(4)	Payload type deletion	Delete an unused payload type from the `m=` line	When an answer with multiple payload types is sent back, there may be cases where they are reduced to one in cases where it is not possible to change the payload type dynamically in RTP during a call.
(5)	Media addition	Add an `m=` line	May be needed for the addition of video communication or other application streams.
(6)	Media deletion	Set the port number to 0 in the `m=` line	
(7)	Direction	Change that value of a to `inactive/sendonly/recvonly/sendrecv`	May be needed when putting calls on hold, etc.
(8)	Received packet interval	Modify `a=ptime`	

Appendix iii. SIP media capability profiles

iii.1. About SIP media capability profiles

This appendix prescribes profiles for declaring the media-processing capabilities of SIP UAs that handle media based on the contents of Appendix ii. This is not a unique scheme for declaring media capabilities, but it is intended to be used as a tool for comparing and confirming common capabilities.

iii.2. SIP media capability profiles

Table A.iii-1 shows the format for declaring media profiles.

Table A.iii-1/JJ-90.21: SIP media capability profiles

	Major category	Minor category	Profile specification format	Notes
1-1	SDP capability element (send)	Receiving IP address	IPv4/IPv6, Unicast/Multicast, specifiable address range, etc.	Section ii.2 Table A.ii-1
1-2		Port number	Range of values that can be specified	
1-3		Codec	Supported codecs	
1-4		Bandwidth	Range of values that can be specified. When not given, declared as “not given”.	
1-5		Packetization interval	Same as above	
1-6		Direction	Same as above	
2-1	SDP capability element (receive)	Receiving IP address	IPv4/IPv6, Unicast/Multicast, specifiable address range, etc.	
2-2		Port number	Range of values that can be specified	
2-3		Codec	Supported codecs	
2-4		Bandwidth	Range of values that can be specified. When not given, declared as “not given”.	
2-5		Packetization interval	Same as above	
2-6		Direction	Same as above	
3-1	SDP capability element special value (send)	Receiving IP address (c= line: `0.0.0.0`)	Presence/absence of transmission, and transmission contract conditions	Section ii.2 Table A.ii-2
3-2		Port number (`m=` line: 0)	Same as above	
3-3		Bandwidth (b= line: `0`)	Same as above	
4-1	SDP capability element special value (receive)	Receiving IP address (c= line: `0.0.0.0`)	Content and conditions of processing during reception	
4-2		Port number (`m=` line: 0)	Same as above	
4-3		Bandwidth (b= line: `0`)	Same as above	

	Major category	Minor category	Profile specification format	Notes
5-1	SDP format (send)	Multi-part MIME body (offer)	Presence/absence of transmission. Transmission contract and conditions when transmitting.	Section ii.3 Table A.ii-3
5-2		Multi-part MIME body (answer)	Same as above	
5-3		SDP with no `m=` line (offer)	Same as above	
5-4		SDP with multiple `m=` lines (offer)	Same as above	
5-5		Multiple payload types (answer)	Same as above	
6-1	SDP format (receive)	Multi-part MIME body (offer)	Content and conditions of processing during reception	
6-2		Multi-part MIME body (answer)	Same as above	
6-3		SDP with no `m=` line (offer)	Same as above	
6-4		SDP with multiple `m=` lines (offer)	Same as above	
6-5		Multiple payload types (answer)	Same as above	
7-1	Early Media (180/Media/Alert-Info)	Received / received / received	Basically, the selection of processing contents from a, b or c. In other cases, the processing contents are explicitly declared.	Section ii.4 Table A.ii-4
7-2		Received / received / not received	Same as above, except that the basic choice is between a and b.	
7-3		Received / not received / received	Same as above, except that the basic choice is between a and c.	
7-4		Received / not received / not received	Same as above, except that the basic choice is a.	
7-5		Not received / received / not received	Same as above, except that the basic choice is between b and d.	
8-1	Procedure for establishing an outgoing call (offer / answer)	`INVITE/2xx`	Presence/absence of compatibility. When conditions exist, they are stated explicitly.	Section ii.5.1 Table A.ii-5
8-2		`INVITE/1xx (100rel)`	Same as above	
8-3		`2xx/ACK`	Same as above (presence/absence of an `INVITE` transmission that does not include an SDP)	
8-4		`1xx (100rel)/PRACK`	Same as above (presence/absence of an `INVITE` transmission that does not include an SDP)	

	Major category	Minor category	Profile specification format	Notes
9-1	Procedure for establishing an incoming call (offer / answer)	INVITE/2xx	Presence/absence of compatibility. When conditions exist, they are stated explicitly.	Section ii.5.2 Table A.ii-6
9-2		INVITE/1xx (100rel)	Same as above	
9-3		2xx/ACK	Same as above (The processing performed on receiving an INVITE containing no SDP is explicitly stated)	
9-4		1xx (100rel)/PRACK	Same as above (The processing performed on receiving an INVITE containing no SDP is explicitly stated)	
10-1	Multiple dialog processing (existing / new)	Early/Early	Content of processing	Section ii.6 Table A.ii-7
10-2		Early/Confirm	Same as above	
10-3		Confirm/Confirm	Same as above	
11-1	Session modification request transmission (State / request)	Confirmed/re-INVITE	Presence/absence of transmission. Transmission contract and conditions when transmitting.	Section ii.7.1 Table A.ii-8
11-2		Confirmed/UPDATE	Same as above	
11-3		Early/UPDATE (UAS)	Same as above	
11-4		Early/UPDATE (UAC)	Same as above	
11-5		Early/PRACK	Same as above	
12-1	Session modification request reception (State/request)	Confirmed/re-INVITE	Ability/inability to receive, and conditions thereof. When reception is rejected, the processing contents, including whether or not the session is to be continued.	Section ii.7.2 Table A.ii-9
12-2		Confirmed/UPDATE	Same as above	
12-3		Early/UPDATE (UAS)	Same as above	
12-4		Early/UPDATE (UAC)	Same as above	
12-5		Early/PRACK	Same as above	
13-1	Session modification contents (send)	Receiving IP address	Presence/absence of a modification request transmission. When transmitting, the opportunity and conditions thereof.	Section ii.7.3 Table A.ii-10
13-2		Receiving port number	Same as above	
13-3		Change payload type	Same as above	
13-4		Delete payload type	Same as above	
13-5		Add media	Same as above	
13-6		Delete media	Same as above	
13-7		Direction	Same as above	
13-8		Received packet interval	Same as above	

	Major category	Minor category	Profile specification format	Notes
14-1	Session modification contents (receive)	Receiving IP address	Ability/inability to modify (state)	Section ii.7.3 Table A.ii-10
14-2		Receiving port number	Ability/inability to modify (state)	
14-3		Change payload type	Ability/inability to modify (state)	
14-4		Delete payload type	Ability/inability to modify (state)	
14-5		Add media	Ability/inability to modify (state)	
14-6		Delete media	Ability/inability to modify (state)	
14-7		Direction	Ability/inability to modify (state)	
14-8		Received packet interval	Ability/inability to modify (state)	

Appendix iv. Notes on SIP terminals that use dynamic IP addresses

iv.1. Problems that occur when using dynamic IP addresses

In SIP terminals, the use of mechanisms that dynamically register the binding of an AoR and Contact address (URI) to a registrar server using a REGISTER message as prescribed by JF-IETF-RFC3261 [1] is widely employed.

In such cases, since the binding at the registrar server is generally a soft state, a call made to this AoR after the actual dynamic IP address of the terminal that performed registration has been released and allocated to another terminal will be processed according to this (old) binding. In this case the following problems can occur:

<Leakage of communication activity/contents of incoming messages>

Since this AoR-addressed message is forwarded to a third party, the fact that a call has been placed to the holder of this AoR is liable to leak out to the third party.

It is also possible that the contents of the SIP message might be leaked to a third party, which might lead to the disclosure of secret information.

<Reception of unexpected messages>

When a dynamically allocated IP address is left behind in a registrar server bound to another AoR, messages addressed to the other AoR may be received unexpectedly.

iv.2. Recommended behavior of terminals when using dynamic IP addresses

If it is possible to purge dynamic IP addresses and provide integrated management of the contents of binding registered in the registrar server on the network side, then it might be possible to eliminate the problems described in section iv.1 by processing on the network side alone, but in general there is no need for integrated management in the network. Under this condition, an effective way of eliminating these problems is for the SIP UA that allocates dynamic IP addresses used by users managed by the provider's SIP network (simply referred to as the SIP UA below) to perform the following behaviors:

<Limitation of binding lifetime>

When the SIP UA uses an IP address obtained dynamically, the binding lifetime registered by the REGISTER request is not set beyond the purge time limit of the dynamic IP address. (Either the IP address purge period is extended, or the binding lifetime is set within the purge period.)

<Explicit cancellation of binding>

When an SIP UA finishes waiting for and receiving incoming calls (e.g., when an application is terminated), it cancels the binding by using a REGISTER request set with the Contact header it registered at the registrar server, and sets the value of the expires parameter of this Contact header to zero or sets the Expires header to 0. After the application or equipment has been restarted, for example, when it is possible to assume a binding that it had previously registered, it should similarly explicitly cancel the binding it had previously registered.

The use of a REGISTER request with the Contact header set to "*" is liable to also cancel the registration of other SIP UAs of the same user (the same AoR), so care must be taken in cases where it is possible for a single AoR to have multiple bindings.

<Confirmation of Request-URI in incoming messages>

SIP UAs do not judge whether or not to process SIP messages solely based on the receiving port number of the received packets, but also check the value of the Request-URI in the SIP request and limit the processing to cases where the value matches what it was expecting (e.g., the URI it had recorded in the Contact header of a REGISTER message). By performing this sort of processing, the SIP UA can avoid calls resulting from unexpected incoming calls caused by bindings left behind in the registrar server beyond the purge period of the dynamic address.

Also, when the SIP UA is operating on the Internet, ignoring messages that contain an unexpected Request-URI is an effective way of avoiding attacks from software that discovers for the IP address of an arbitrary IP-based SIP application.

Appendix v. The SIP URI of From headers

v.1. Purpose of this appendix

In section 4.5.2 of the main body of this document, the requirement that spoofing of other users must not be performed was mentioned as a way of ensuring the validity of the From header in interface A. This appendix presents guidelines for guaranteeing that, for example, different users do not have the same SIP URI between multiple providers' SIP networks.

v.2. Anonymous URI

Since an anonymous URI (sip:anonymous@anonymous.invalid) is guaranteed not to be set for a specific user in all the providers' SIP networks, it is recommended that an anonymous URI is used when no particular user is being specified.
By using a URI that expresses anonymity within each provider's SIP network (e.g., sip:anonymous@ttc.or.jp), it is possible to manage the network without specifying a particular user, but it should be kept in mind that the anonymity of the provider is likely to be lost.

v.3. SIP URIs

v.3.1. host part

When the host part takes the hostname format, it uses the correct domain name or server name with a root consisting of a TLD or ccTLD managed by ICANN which has the authority used by the provider that manages the provider's SIP network. In this case, the provider that manages the provider's SIP network includes the domain name or server name entrusted for operation from another provider.
When the host part is formatted as an IPv4 address, it takes the fixed IP address belonging to the provider that manages the provider's SIP network, and must not use a dynamic IP address allocated to a terminal or the like.

v.3.2. user part

When the provider that manages a provider's SIP network allocates a SIP URI to a user managed by this provider's SIP network, the user part should be specified so as to be unique with respect to the specified hostname part. No limits are imposed on the format of the user part, except that it should be unique and that it should conform to the ABNF (Augmented Backus-Naur Form) prescribed by JF-IETF-RFC3261 [1].
When generating a From header for a signal message received from other provider's network and not from a user managed by the provider's SIP network, its value must be chosen and set so that the user part of the hostname part in the SIP URI set by the boundary must be set so that the same value is never generated when the caller are actually different. For example, when a connection to other provider's network is made using ISUP (interface C), the above condition can be satisfied by selecting the text string of the E.164 number.