Internet Engineering Task Force
INTERNET-DRAFT
draft-ietf-mmusic-scip-00.ps

MMUSIC WG
Schulzrinne
GMD
February 22, 1996
Expires: 8/1/96

Simple Conference Invitation Protocol

Status of this Memo

Abstract

The conference invitation protocol (SCIP) is an application-level protocol for inviting users to multimedia conferences. Network users are identified by their universal communication identifier, usually their electronic mail address. SCIP offers personal mobility by supporting forwarding and redirection. It can reuse the general email infrastructure, including DNS MX records, mailing lists and aliases. The protocol combines aspects of HTTP and SMTP and can re-use their security mechanism. The protocol is extensible in methods and parameters and is designed to allow interoperation with ITU-T T.124 (Generic Conference Control). Extension to VCR-control are possible as well. The protocol supports both loose and tight conference styles.

1 Introduction

1.1 Purpose

The conference invitation protocol allows users to invite other users as well as automatic applications to point-to-point or multicast conferences. It provides extensions

1.2 Requirements

This document uses the same words as RFC 1123 for defining the significance of each particular requirement. These words are:

must: This word or the adjective "required" means that the item is an absolute requirement of the specification.

should: This word or the adjective "recommended" means that there may exist valid reasons in particular circumstances to ignore this item, but the full implications should be understood and the case carefully weighed before choosing a different course.

may: This word or the adjective "optional" means that this item is truly optional. One implementation may choose to include the item because a particular application requires it or because it enhances the product, for example, another implementation may omit the same item.

An implementation is not compliant if it fails to satisfy one or more of the *must* requirements for the protocols it implements. An implementation that satisfies all the *must* and all the *should* requirements is said to be "unconditionally compliant"; one that satisfies all the *must* requirements but not all the *should* requirements for its protocols is said to be "conditionally compliant".

1.3 Terminology

This specification uses a number of terms to refer to the roles played by participants in SCIP communications. The definitions of client, server and proxy are similar to those used by HTTP.

Calling party: The party initiating a conference invitation. Note that the calling party does not have to be the same as the one creating a conference.

Called party: The person or service that the calling party is trying to invite to a conference.

Conference: A logical grouping of several sessions. A conference is identified by a globally unique conference identifier.

Conference member: The union of all session members.

Client: An application program that establishes connections for the purpose of sending requests. Clients may or may not interact directly with a human user.

Session member: A member of a session, either an application used by a human or a support tool of some kind (e.g., a video recorder).

Server: An application program that accepts connections in order to service requests by sending back responses. A server interacts with the called user agent to determine whether to accept a call.

Session: A single media, identified by a common media identifier. In a multicast setting, each session has a single multicast address.

Proxy: An intermediary program which acts as both a server and a client for the purpose of making requests on behalf of other clients. Requests are serviced internally or by passing them, with possible translation, on to other servers. A proxy must interpret, and, if necessary, rewrite a request message before forwarding it.

[calling user agent:] The client application which initiates a request.

Any given program may be capable of acting both as a client and a server. A typical multimedia conference controller would act as a client to initiate calls or invite others to conferences and as a server to accept invitations. However, since a server should be reachable even if the conference controller is not running, server and conference controller may well be separate. The protocol between server and conference controller is a local matter, but SCIP itself may be used for implementation efficiency. In that case, the conference controller may well initiate a connection to the server after being started by the server. The issues are somewhat similar to the separation of MUA and MTA on a local host, with the added difficulty that synchronous communication is needed. (TBD: move this section?)

1.4 Overall Operation

The protocol can be used to either initiate a two-party multimedia call, similar to a phone call, to an invidual or to invite an individual to a multicast conference. Except for using unicast or multicast network addresses in the media description, there is no difference in protocol operation or end system behavior. Conferences may take place immediately or in the future. The protocol can convey information about conferences that repeat a determinate or indeterminate number of times. The protocol can also "invite" a recorder to a conference and thus serve as a control mechanism for accessing media-on-demand services.

The SCIP protocol is based on HTTP and employs many of its concepts, data types, and protocol operations. The protocol is designed so that it could share a single server with HTTP, although that is usually not desirable.

The called party is identified by its electronic mail (RFC 822) address. It is also possible that the UCI differs from the (primary) electronic mail address, but this is not recommended. The domain named in the UCI should also accepts SMTP connections, but may simply forward them to a regular electronic mail exchange.

The caller contacts the server, located according to Section 1.4.1, with a call request. The request contains information about the originator, subject and urgency of the call and, typically for conferences, the anticipated duration. For conferences, out-of-band contact information such as email addresses, phone numbers or URIs may also be offered. The call request indicates the desired media, together with their encoding and network parameters such as the unicast or multicast address, protocol type and port number.

The callee may either accept the call, forward the call or reject the call. When accepting a call, the called party returns a subset of the media listed in the Accept field of the request, namely those media it is prepared to receive. The called party must not generate any media types not listed in the Accept field. If several parties are being invited in parallel, a second round of negotiation may be needed. For large-scale conferences, a separate, multicast-based negotiation protocol may be preferable, but has not been specified. When rejecting a call, the server may offer a reason or a time to call back at (using the Retry-After field).

If the called server runs on a mail exchange host, the called user will likely not be reachable on that host. Rather, that server will use a local mechanism to locate the user within the local environment. This local mechanism is beyond the scope of this specification; examples include multicast queries, user registration services or use of active badges. The server may also map a user name to a temporary IP address to address the common situation of users connected to the Internet by a modem. A server can operate in *redirect* or *proxy* mode. In redirect mode, the server returns a status code and header fields indicating a possible current location of the called party. In proxy mode, the server maintains the incoming SCIP connection while it establishes new client SCIP connection to the intended called party. It then simply forwards the response of the called to the caller. A caller should cache the current location of the called party so that it can short-circuit the lookup process for future calls. There is no indication of lifetime (say, via an Expires header), since user behavior is unpredictable.

As in HTTP, the called server closes the connection. There is a keep-alive mechanism that allows the client to request that the server maintain the connection. This can be used for tight conference control and T.120 interoperation. However, it is also possible to request conference parameter changes even when the initial connection was torn down after call acceptance by establishing a new connection.

A forwarding or name resolution may yield more than one name, for example, when expanding a mailbox using SMTP EXPN. The user agent should offer a choice of "reach first" or "reach all". In the "reach first" case, the caller tries each UCI in turn, only the first one accepts the call. In the "reach all" case, the client tries to invite as many as possible. It may do this either in parallel or sequentially. (It may be useful to describe UCIs, similar to the HTTP/1.1 URI field, for example, to indicate, for example, several departments within an organization or the language-abilities of different possible parties.) Note that while a response may indicate several UCIs, a single SCIP request can only act on a single UCI.

In a multiprocess operating system, a single server per host will typically be running continuously as a priviledged process. For incoming calls, the server can either determine the call disposition automatically, based on some user-specified rules, or signal to the user an incoming call, e.g. through an acoustic signal or a pop-up window. If the called party accepts the call, the server then starts the necessary conferencing applications and passes the parameters conveyed by the call request to them.

1.4.1 Resolving Addresses

A client implementing the SCIP protocol should follow the following steps in locating a server belonging to the callee address. For brevity, the action "check if valid server" implies attempting to connect to the address at the service TCP port. If the connection attempt succeeds, the sequence is aborted.

- Strip the domain part from the addr-spec.
- If a location has been cached for this UCI, check if called party is present there.
- If the domain has a DNS A record, check if valid server.
- If the domain has a DNS SRV resource record [1], check if valid server.
- If the domain has a DNS MX record, check (in order of MX preference) if one of the records points to a valid server.

Note that this procedure makes it possible to have a SCIP-only server (i.e., one not acting as an MTA) by simply adding it as an MX record, usually with lower priority than a true MTA. Mail delivery will simply skip the SCIP-only server, just as SCIP will skip any non-SCIP mail exchange hosts.

If the procedure above does not yield a SCIP server or if the user name is not recognized by a SCIP server, a client should attempt to contact the mail transfer agent for the same domain using SMTP and expand the name using the SMTP EXPN and VRFY commands. If EXPN yields more than one name, the client should treat this as a group call and contact all list members in the manner described above. The client should contact list members in parallel. A client may limit the number of parallel connection attempts; a user agent acting a client may request confirmation if the number of addresses exceeds a given threshold. SMTP expansion can be used to offer the service of life-long UCIs, without actually handling calls.

If all attempts to contact a SCIP server fail, a user agent may attempt to send a MIME message to the address, with content type `message/cip`.

2 Notational Conventions and Generic Grammar

2.1 Augmented BNF

See RFC HTTP 1.1, Section 2.1.

2.2 Basic Rules

See RFC HTTP 1.1, Section 2.2. The attribute-value bag is not used.

```
phone-number  =  E123 / phrase ”<” E123 ”>” / E123 comment
E123          =  ”+” country-code (SPACE / ”-”)
                 1*(DIGIT / ”A” / ”B” / ”C” / ”D” /
                 ”#” / ”*” / ”.” / ”-”)
country-code  =  1*3 DIGIT
time          =  rfc1123-date / hex-time
hex-time      =  *HEX ; time in seconds since 1 Jan 1900
```

3 Protocol Parameters

3.1 Product Tokens

See HTTP/1.1, Section 3.8.

3.2 Universal Communication Identifiers

SCIP defines universal communication identifiers (UCIs) as a generalization of mailboxes according to RFC 822. UCIs can be used for electronic mail or interactive communications. UCIs have the format of RFC 822 addr-specs, possibly with some extensions.

TBD: Possibly affix port number (with :port) to allow user-space implementations and sharing of HTTP servers?

A special form of UCI is an E.164 international telephone number, written as a numeric string preceded by a plus-sign, followed by the country code. A phone number may contain the digits 0 through 9, the letters A through D, the star (ASCII 0x2A) and the pound sign (ASCII 0x23). A E.164 UCI may employ the ASCII period ”.” (ASCII 0x2E) or dash ”-” (ASCII 0x2D) for grouping. These are ignored when processing. An application may either directly dial this telephone number through a local computer-telephony interface, establish a phone-call through a locally-configured LAN-telephony gateway or offer the user the number for manual dialing through an appropriate interface. It is the responsibility of the computer-telephony interface or gateway to translate the number to a number that is valid at the origination point of the telephone call, e.g., by removing the country code and prefixing the remainder with any long-distance access code, or dialing the appropriate international access code. (Note: Implementation of the full AT modem dial commands such as pauses or choosing between tone and pulse dialing is not useful, since dialing conventions will differ from location to location.) E.140 telephone numbers can be easily distinguished from mailboxes by the presence of the leading plus sign and the absence of the at-sign. (TBD: Can a mailbox start with a +?)

3.3 UCIs as Uniform Resource Identifiers

Since UCIs are meant as universal identifiers for both synchronous and asynchronous communications, SCIP reuses the `mailto` URI scheme (see Section XX, RFC 1738). It is suggested that the `mailto` URI be extended to encompass telephone numbers as well. The user interface of browsers implementing SCIP should offer the user the choice of either sending electronic mail or trying to establish real-time communication.

Since the number of parameters and the total length of a typical conference description is large, it is recommended to use a http or ftp or similar scheme to retrieve an object of type message/scip, defined in Appendix A. It is also possible to use the “data” URI scheme [2] to convey information of type `message/cip`.

4 SCIP Message

SCIP headers may be folded as described in RFC 822.

5 Request

A request from a client to a server includes, within the first line of that message, the method to be applied to the UCI, the identifier of the called party, and the protocol version in use. (Note: there is no need for a HTTP/0.9 backward compatible request.)

Request-Line	=	method SP UCI SP "SCIP/1.0" CRLF

5.1 Method

6 Response

A server returns a response message to a request.

Response	=	Status-Line *(General Header / Response header) CRLF
Status-Line	=	SCIP-Version SP Status-code SP Reason-phrase CRLF

An example:

```
SCIP/1.0 302 Moved Temporarily
Location: secretary@westwing.whitehouse.gov
Location: security@eastwing.whitehouse.gov
```

7 Method Definitions

For T.124 compatibility, additional methods will be defined in the future.

7.1 CALL

Call the user identified by the called-UCI.

7.2 CHANGE

Change parameters of the conference.

7.3 CLOSE

Close the conference.

8 Status Code Definitions

8.1 Informational (1xx)

Currently, no 1xx type status codes are defined for SCIP. The HTTP codes 100 (Continue) and 101 (Switching Protocols) are not applicable. Progress indication such as "ringing" may be useful.

8.2 Successful (2xx)

The HTTP status codes 201 (Created), 202 (Accepted), 203 (Non-Authoritative Information), 204 (No Content), 205 (Reset Content), 206 (Partial Content) are not applicable and must not be sent in response to a SCIP method.

8.2.1 200 OK

The request has succeeded. The information returned with the response depends on the method used in the request:

CALL The call has been accepted by the called party.

8.3 Redirection 3xx

8.3.1 301 Moved Permanently

The user identfied by the UCI has moved permanently and any future calls should be to the new UCI returned in the Location field. If a user may be at several locations, several Location fields may be returned, with the client then trying to contact the new locations in turn or in parallel. (HTTP only allows one Location field.)

8.3.2 302 Moved Temporarily

The user identified by the UCI has moved

8.4 Caller Error 4xx

The 4xx class of status codes is intended for cases in which the client (caller) seems to have erred. Status codes 400 (Bad Request), 401 (Unauthorized), 402 (Payment Required), 403 (Forbidden), 404 (Not Found), 405 (Method Not Allowed), 406 (None Acceptable), 408 (Request Timeout), 410 (Gone) have the same interpretation as for HTTP, with URI replaced by UCI.

8.5 Callee Error 5xx

The 5xx class of status codes indicates that the server is incapable of completing the request. The codes 500 (Internal Server Error), 501 (Not Implemented), 502 (Bad Gateway), 503 (Service Unavailable), 504 (Gateway timeout) are to be interpreted in the same way as in HTTP 1.1, except that URI is to be replaced by UCI. Busy conditions, i.e., where the called party indicates she does not wish to receive calls, or a busy signal from a telephony gateway, are indicated by status code 503. If known, a Retry-After field can give an indication when a new call may succeed.

9 Header Field Definitions

9.1 Accept

```
Accept        =  ”Accept” ”:” #(
                 media-range
                 [”;” ”ttl” ”=” ttl-value]
                 [”;” ”addr” ”=” net-address]
                 [”;” ”cbw” ”=” bandwidth]
                 [”;” ”bw” ”=” bandwidth]
                 [”;” ”key” ”=” encryption-key]
                 [”;” ”id” ”=” media-id]
                 [”;” ”i” ”=” information]
                 [”;” ”tp” ”=” (”rtp” / other-transport-protocol)]
                 [”;” ”pt” ”=” payload-type]
                 [”;” ”dir” ”=” ”sendonly” / ”recvonly” /
                 ”hduplex” / ”fduplex”]
                 )
media-range   =  type ”/” (subtype / ”*”)
type          =  (”audio” / ”video” / ”application”)
subtype       =  (audio-enc [”.” sr ”.” ch]) / video-enc / application)
net-address   =  [(host / multicast-address)] [”:” port]
sr            =  <audio sampling rate>
ch            =  <audio channel count>
```

For audio and video, the media subtype designates the encoding using the RTP profile designations, e.g., H261, PCMU, L16, etc.. Parameters propagate, that is, they apply to all following media, even across several Accept lines. (This follows from the HTTP convention that several fields with the same name are equivalent to a single field with comma-separated items.) TBD: There are currently no preferences encoded, as this makes responses difficult to handle. (Who decides if the preference of the called and calling parties differ?) However, adding a “q” parameter like HTTP might be useful if the behavior can be defined.

The response contains the subset of media that the called party can support, without parameters. (This is the reason that sampling rate, channel count and other identifying encoding parameters are part of the subtype rather than parameters.)

bw: Per-sender bandwidth, in kb/s.

cbw: Conference bandwidth, in kb/s. This is the total bandwidth for all senders of this media instance.

id: Media instance identifier. The identifier is a random base-64 string with at least 8 characters that can be used to identify media change requests. Different values of this field separate several media streams. A media stream can consist of a number of media types which are either only used sequentially or, if used in parallel and with different network associations, carry exactly the same information. (TBD: Should the latter be allowed? It is useful for having both low-bandwidth and high-bandwidth versions of the same material.)

ttl: The multicast time-to-live value.

key: Encryption key for this media type, in base-64 encoding.

pt: RTP dynamic payload type for the encoding.

tp: Transport protocol.

dir: Direction of transmission, as viewed from the called part. Half-duplex (hduplex) and full-duplex (fduplex) permit sending and receiving. Half-duplex indicates that there is a mechanism to ensure that only one party speaks at any given time and is mainly useful for two-party conversations.

Example request for a conference containing audio and video:

```
CALL foo@bar.com SCIP/1.0
Accept: audio/pcmu.16000.1;ttl=128;addr=224.2.0.1;pt=95;id=Axuay,
        audio/gsm.8000.1
Accept: video/h261;ttl=128;addr=224.2.0.2;id=Zkd1k,
        video/jpeg;bw=128;recvonly
```

The sample response shown below indicates that the called party only supports PCMU audio and JPEG video:

```
200 OK
Accept: audio/pcmu.16000.1
Accept: video/jpeg
```

TBD: Alternative: Identify possible encodings in response numerically or by identifier to simplify matching.

9.2 Accept-Language

TBD.

9.3 Authorization

TBD.

9.4 Call-Id

Each call must be identified by a globally unique call identifier. If a user invites somebody to the same conference, it must use the same Call-Id it was invited with. (TBD: For T.120 interoperation, this field is the conference identifier, which is not guaranteed to be globally unique.)

Call-Id = "Call-Id" ":" "<" local-id "@" addr-spec ">"

9.5 Forwarded

The Forwarded response header is to be used by servers to indicate the intermediate steps between the calling user agent and the server on call requests. A Forwarded response is not inserted if redirection occurs. A proxy server adds a Forwarded field in the response to the client when it contacts another server for call completion. It is analogous to the "Received" field of RFC 822 and the Forwarded field of HTTP. It is intended to be used for tracing transport problems and avoiding request loops.

Forwarded	=	"Forwarded" ":" "for" FQDN
FQDN	=	<Fully qualified domain name>

Multiple Forwarded header fields are allowed and should represent each proxy server that has forwarded the call request. It is strongly recommended that proxies used as a portal through a network firewall do not, by default, send out information about internal hosts within the firewall region. This information should only be propagated if explicitly enabled. If not enabled, the for token and FQDN should not be included in the field value, and any Forwarded headers already present in the message (those added behind the firewall) should be removed.

9.6 From

From = "From" ":" mailbox

An example is

```
From: Herbert Hoover <president@whitehouse.gov>
```

9.7 Keep-Alive

Used in request to have server keep open the connection after the initial call.

9.8 Key

This field can be used to convey an encryption key for the media sessions that are marked as encrypted. Naturally, the key is safe from eavesdroppers only if the SCIP connection is itself encrypted.

9.9 Location

Location = "Location" ":" absoluteUCI

An example is:

```
Location: secretary@westwing.whitehouse.gov
```

9.10 Phone

The Phone field provides contact information for the person initiating the conference. This person may differ from the one issuing the conference invitation or creating the conference announcement. For a multicast conference, either the Phone or Email field must be specified. More than one Phone or Email field is allowed. The field is intended to be used as the RTCP SDES fields by the same name, e.g., to summon help if problems arise or to contact a human operator should a conference cause network problems.

Phone = "Phone" ":" phone-number

An example is:

```
Phone: conference operator <+1.415.555.1212>
Phone: +49.30.25499.182 (Conference Chair)
```

9.11 Priority

Priority = "Priority" ":" ("urgent" / "high"
/ "normal" / "low" / user-defined-priority)

The Priority field gives a general indication of the urgency of the message. This allows the user agent of a called party to forward or refuse calls without assistance from the user. Abuse is dealt with by social oppobrium.

9.12 Reach

Reach = "Reach" ":" ("first" / "all")

9.13 Repeat

The Repeat field indicates for conferences, that the session repeats at some interval.

9.14 Retry-After

The Retry-After field indicates the earliest time another call attempt is likely to be successful. It has the same syntax as the HTTP field by the same name.

9.15 Subject

The Subject provides a summary of the conference topic or indicate the nature of the call.

9.16 Time

Time	=	"Time" ":" start-time ["," stop-time]
start-time	=	rfc1123-date ; time conference starts
stop-time	=	rfc1123-date ; time conference ends

9.17 URI

The URI header field points to additional information about the conference and/or the caller. There should be no more than one URI field.

URI = "URI" ":" URI

An example is:

```
URI: http://www.w3.org/pub/Talks/
```

9.18 User-Agent

(See HTTP 1.1, 10.43) The User-Agent field contains information about the user agent originating the request. This is for statistical purposes, the tracing of protocol violations, and automated recognition of user agents for the sake of tailoring responses to particular user agent limitations. Although it is not required, user agents should include this field with the requests. The field can contain multiple product tokens (see Section 3.1) and comments identifying the agent and any subproducts which form a significant part of the user agent. By convention, the product tokens are listed in order of their signficance for identifying the application.

User-Agent = "User-Agent" ":" 1*(product / comment)

Example:

```
User-Agent: isc/1.2 libscip/0.9
```

10 Response

11 Security Considerations

TBD.

12 Acknowledgements

The document structure and parts of the texts were lifted from the HTTP/1.1 specification. The compact representation is similar to that used by SDP.

13 Author Address

Henning Schulzrinne
GMD Fokus
Hardenbergplatz 2
D-10623 Berlin
Germany
electronic mail: schulzrinne@fokus.gmd.de

A Internet Media Type message/scip

In addition to defining the SCIP/1.0 protocol, this document serves as the specification for the Internet media type "message/scip". The following is to be registered with IANA according to RFC 1590:

```
Media type name:          message
Media subtype name:       scip
Required parameters:      none
```

```
Optional parameters:     version, msgtype
  version: The SCIP version number of the enclosed message (e.g.,
           "1.0").  If not present, the version can be determined from
           the first line of the body.
  msgtype: The message type - "request", "response" or "directory".
           If "directory", only the header fields of the request are
           in the body.  If not present, the type can be determined
           from the first line of the body.
Encoding considerations: only "7bit", "8bit", or "binary" are
                         permitted.
Security considerations: none.
```

B Compact Representation

In some environments, bandwidth is at a premium and a more compact representation is desired. Examples of such environments include encoding conference information in `data` URIs [2] or carrying them in limited-bandwidth multicast directories. The compact representation described here should only be used in those circumstances, as it is harder for humans to debug and is less compatible with HTTP and SMTP conventions. It is anticipated that a single parser can process both formats without difficulty.

The following abbreviations may be used for field names:

Accept	M
Email	e
Key	k
Phone	p
Repeat	r
Time	t
Subject	i
URI	u

Fields not listed above are generally not used in non-interactive (e.g., directory or WWW) applications. In a compact representation, a single LF should be used as a line terminator.

C Tolerant Applications

The line terminator for SCIP-header fields is the sequence CRLF, as for HTTP. However, it is recommended that applications, when parsing such headers, recognize a single LF as a line terminator and ignore the leading CR.

D T.124 and H.245 Interoperation

T.124 and H.245 interoperation is through gateways.

SCIP field	T.124 parameter
Subject	Conference Description
	Conference ID
	Conference Name
	Conference Name Modifier

E Open Issues

- Character set: ISO 8859-1 or UTF-7? The former is easily available on most systems, covers a large fraction of the non-Asian current Internet population and is HTTP compatible. UTF-7 is probably more future-safe.
- Terminology: Session for group of streams? (but: also applications); conference? (but also "invitation")
- Name of protocol: Could be SIP (session invitation protocol), but invitations are to conferences, which consist of (audio, video, ...) sessions. Also, may be used for more than just invitation (configuration). CICP,
- Methods for VCR-style control.
- Operation with telephony gateways.
- Charging mechanisms for media-on-demand - reuse HTTP extensions?

References

[1] A. Gulbrandsen and P. Vixie, "A DNS RR for specifying the location of services," Internet Draft, Internet Engineering Task Force, Jan. 1996. Work in progress.

[2] L. Masinter, "Data: URL scheme," Internet Draft, Internet Engineering Task Force, Feb. 1996. Work in progress.