Simple Universal Call/Conference Establishment Sequence - SUCCESS

P. Cordell, BT Labs

Overview

- Purpose
- Protocol requirements
- Brief protocol description
- Capability negotiation
- Message encoding
- Conclusion

Purpose

- Exchange ideas
- Share our requirements

- Work in progress

A Single Protocol

- Easy to listen to
- No protocol interworking issues
- Uniform extension mechanism
- Single initial point of contact
  - May jump into SCCP, T.120 or H.323 later

Extensible Call Model

- Loose control
- Tight control
  - Membership kept in call control plane
  - Close down phase
  - Progress information for user and diagnostics

Extensible Features

- Supplementary services
  - Transfer, Hold etc.
- Authentication
- Floor control
- Java Applets

- Extension through standards bodies
- Extension by proprietary (de-facto?) features

Universal Initiation

- Initiate real-time conversational streams
- Initiate live ‘broadcast’ feeds
- Initiate pre-stored server feeds
  - Including answering machine functions etc.

Environment

- Interworking with POTS and H.320
- Third party call control
  - Call centres
  - Managed conferences

The Protocol

- Hello - Announce presence in conference
  - Includes reply and reply-ack field
- Progress - Indicate progress entering call
- Bye - Indicate leaving conference
- ByeBye - Acknowledge endpoint leaving
- Feature - Extensible feature frame work

- draft-cordell-success-00.txt.Z

Feature Message

- Functionality supported by ‘plug-ins’
- Features register with call control layer
- Call control exchanges messages
- Check for registered feature on message reception

Capability Negotiation

- Important to allow new, better codecs
- Recommend common base set
  - G.723 - Good for slow links
  - QCIF H.261 - Compatible with existing standards
- Negotiation prior to conference

Negotiation

- Capabilities in Hello message
- Message indicates streams being sent
- Allows multiple sets of media so multiple streams of similar type can be negotiated
- Can determine common receive capabilities over time
- Receivers may have to switch decoders based on RTP header

Aligning with SIP

INVITE 128.16.64.19/65729 SIP/2.0

From:mjh@isi.edu

To:*

To:schooler@cs.caltech.edu Reply

To:pete@bt.com

Respond-to:IN IP4 UDP 239.128.16.254 1 16

Content-type:meta/sdp

Content-Length:187

Message Encoding

- Message description independent of line format
- Optional future extensions requires hierarchy in messages
- Common format allows
  - Easy understanding
  - Easy definition of new standards
  - Code re-use

Encoding Suggestion

- ASCII line format
- ASN.1 subset message description

INTEGER	OCTET	IA5String
BMPString	STRING	SIZE
SEQUENCE	CHOICE	BOOLEAN
NULL	OPTIONAL	SET OF
SEQUENCE OF		

- Line format: x = y

Encoding Examples

- ASN.1: size INTEGER(0..255),
  coded as: size = 12
- ASN.1:

```
link SEQUENCE {
        address   IA5String(SIZE(0..255)),
        port   INTEGER(0..65535) }
coded as:       link = ( address = “bt.com”
                                port = 1200 )
```

Use Abbreviations

- If: link == l
  address == a
  port == p
- Message becomes:
  l = (a = “bt.com” p = 1200)

Conclusion

- Single extensible protocol required
- Interworking with existing networks required
- Network services required
- Third-party control required