Session Description Protocol

Author: *James Wright, MSc*

"This paper is a technical overview of the Session Description Protocol and is designed for IT professionals, managers, and architects who want a quick low-level guide to the workings and details of the protocol."

Konnetic
SIP & IMS .NET development

Session Description Protocol

With the advent of protocols used to negotiate and define a communication session's parameters (e.g. Session Initiation Protocol), there was a need to explain the purpose and enrolment process. Session Description Protocol (SDP), defined in RFC4566, achieves that by providing a format for session characterisation and media definition. Using this protocol participants are expected to agree on the descriptive values, timings, their respective capabilities and desired media formats. This exchange is referred to as the Offer/Answer Model and is formalised in RFC3264. SDP can be used with a number of transport protocols, such as Session Announcement Protocol (SAP), Session Initiation Protocol (SIP), Hypertext Transfer Protocol (HTTP), and others.

Content

Several important pieces of information are mandatory within an SDP message:

- Session name.

- Time(s) the session is active.

- The media comprising the session.

- The owner/originator of the session.

- How to receive the media (addresses, ports etc).

Other optional information may be provided:

- Bandwidth to be used by the conference.

- The purpose of the session.

- Contact information for the person responsible for the session.

- Time zone information.

- Session attributes extending SDP.

Anatomy of a Message

Field Format

The encoding of the protocol is primarily UTF8 (descriptive fields can have other encoding as specified with a 'charset' attribute – defined later). Each piece of information is conveyed in a field. Each field is separated from the next by a carriage return/line feed sequence (CRLF). The form of each field is:

<type>=<value>[CRLF]

Where the *<type>* is a case-insensitive and unique single character field name. And *<value>* is structured text whose

format depends upon *<type>*. They are separated by an unpadded '=' (equal) sign.

Message Structure

Within an SDP message there are three main sections, detailing the Session, Timing, and Media descriptions. Each message may contain more than one Timing and Media description. Each field must appear in the order shown:

Table 1. *Session Description*

Field	Type	Opt/Mnd	Description
Protocol Version	v	M	The current protocol version. Always "0" using RFC4566.
Origin	o	M	The session originator's name and session identifiers.
Session Name	s	M	The textural session name.
Session Information	i	O	Textural information about the session.
Uri	u	O	A pointer to supplemental session information.
Email Address	e	O	Email contact information for the person responsible.
Phone Address	p	O	Phone contact information for the person responsible.
Connection Data	c	C	The connection type and address.
Bandwidth	b	O	Proposed bandwidth limits.
			[Timing Descriptions Go Here]
Time Zones	z	O	Accounts for daylight saving information.
Encryption Keys	k	O	A simple mechanism for exchanging keys. Rarely used.
Attributes	a	O	One or more attributes that extend the protocol.
			[Media Descriptions Go Here]

Note: M - mandatory; O- optional; C- Conditional (Connection Data must appear in either the Session or Media descriptions).

Table 2. *Timing Description*

Field	Type	Opt/Mnd	Description
Timing	t	M	Start and end times.
Repeat Times	r	O	Specified the duration and intervals for any session repeats.

Times are represented as Network Protocol Time (RFC1305): the number of seconds since 1900; intervals can be represented with NTP times or in typed time: a value and time units (days ('d'), hours ('h'), minutes ('m') and seconds ('s')) sequence.

Thus an hour meeting from 10am on 1st August 2010, with a single repeat time a week later at the same time can be represented as:

> *t=3487140000 3487143600*

> *r=604800 3600 0*

Or using typed time:

> *t=3487140000 3487143600*

> *r=7d 3600 0*

Table 3. *Media Description*

Field	Type	Opt/Mnd	Description
Media Descriptions	m	M	Media definitions including media type (e.g. 'audio'), transport details and formats.
Session Information	i	O	[Same as above]
Connection Data	c	C	[Same as above]
Bandwidth	b	O	[Same as above]
Encryption Keys	k	O	[Same as above]
Attributes	a	O	[Same as above]

Attributes

SDP uses attributes to extend the core protocol. Attributes can appear within the Session or Media sections and are scoped accordingly as "session-level" or "media-level".

Attributes take two forms:

- A property form: "a=<flag>" conveys a property of the session.

- A value form: "a=<attribute>:<value>" provides a named parameter.

Table 4. *Sample Attributes*

Attribute	Form	Description
Category	cat:<category>	A dot-separated hierarchical category used for filtering sessions.
Keywords	keywds:<keywords>	Assists in the identification of sessions.
RTP Payload Type	rtpmap:<payload type> <encoding name>/<clock rate>	Maps an RTP payload to the encoding, format and clock rate.
Receive Only	recvonly	Tools should start in receive-only mode.
Send/Receive	sendrecv	Tools can start in send and receive mode.
Type	type:<conference type>	Types include "broadcast", "meeting" and "moderated".

KONNETIC
SIP & IMS .NET development

Attribute	Form	Description
Charset	charset:<character set>	The character set used in the session name and information fields.
Language	lang:<language tag>	The default language for the session.

Example

Below is an example session description for a seminar presentation on "SDP Implementation" available in audio and video over RTP from address 124.191.8.1 using ports 49170 and 51372 respectively. It is an hour long seminar which starts at 10am on 1st August 2010 and the contact is John Doe at jdoe@mel99.melbourne.com.

Table 5. *SDP Message Example*

Line	Description
v=0	The version of the protocol is "0" RFC4566.
o=jdoe 2890844526 89 IN IP4 214.191.7.5	jdoe is the originator. The session ID is 2890844526 and the version is 89. The session was created on machine 214.191.7.5.
s=SDP Implementation	A short textural session name
i=A Seminar on the session description protocol	Further information about the session.
u=http://www.konnetic.com/documents/sdpseminar.html	A link to extra information about the session.
e=jdoe@mel99.melbourne.com (John Doe)	The email for jdoe, including full name.
c=IN IP4 124.191.8.1/127	The IP connection to the session host is 124.191.8.1 using a time-to-live of 127.
t=3487140000 3487143600	The number of seconds since 1900 for the timings. These represent an hour meeting at 10am on the 1st August 2010.
a=recvonly	Applications should begin the session in receive-only mode.
m=audio 49170 RTP/AVP 0	Session audio is on port 49170 using RTP Profile for Audio and Video Conferences with minimal Control running over UDP. The final zero is extra parameter information for RTP/AVP.
m=video 51372 RTP/AVP 31	Session video is on port 51372 using RTP Profile for Audio and Video Conferences with minimal Control running over UDP. The final 31 is extra parameter information for RTP/AVP.
a=orient:portrait	Any presentations will be in portrait.

Note: The two media descriptions (the lines beginning with m), define an audio and a video profile. These profiles are described in the Real Time Protocol (RTP) specification, RFC3550, and the companion document, RTP Profile for Audio and Video Conferences with Minimal Control, RFC3551.

KONNETIC
SIP & IMS C# & VB development

Further Reading

M. Handley et al., "SDP: Session Description Protocol" RFC 4566 2006.

J. Rosenberg et al., "SIP: Session Initiation Protocol" RFC 3261 2002.

H. Schulzrinne. (2010) Henning Schulzrinne. [Online]. http://www.cs.columbia.edu/~hgs/

A. B. Johnston, SIP: Understanding the Session Initiation Protocol, 3th ed. Boston, MA, USA: Artech House, 2007.

H. Schulzrinne et al., "RTP: A Transport Protocol for Real-Time Applications" RFC 3550 2003.

Konnetic Website. http://www.konnetic.com

Other Papers In the Series

IMS - Technologies and Architecture

http://www.konnetic.com/documents/KonneticIMSTechnologiesAndArchitecture.pdf

IMS - Services and Revenue

http://www.konnetic.com/documents/KonneticIMSServicesAndRevenue.pdf

SIP - An Introduction

http://www.konnetic.com/documents/KonneticSIPIntroduction.pdf

SIP - Technical Overview

http://www.konnetic.com/documents/KonneticSIPTechnicalOverview.pdf

James Wright MSc is the founder of Konnetic, a specialist in providing SIP and IMS based software to the .NET (C#, VB.NET, F# and more) community.

KONNETIC
SIP & IMS .NET development